Report from Dagstuhl Seminar 18371

# Knowledge Graphs: New Directions for Knowledge Representation on the Semantic Web

**Edited by**

**Piero Andrea Bonatti[1], Michael Cochez[2], Stefan Decker[3], Axel Polleres[4], and Valentina Presutti[5]**

1 **University of Naples, IT,** pieroandrea.bonatti@unina.it
2 **Fraunhofer FIT, DE,** michael,cochez@fit.fraunhofer.de
3 **RWTH Aachen, DE,** decker@informatik.rwth-aachen.de
4 **Wirtschaftsuniversität Wien, AT,** axel.polleres@wu.ac.at
5 **STLab, ISTC-CNR - Rome, IT,** valentina.presutti@istc.cnr.it

## Abstract

The increasingly pervasive nature of the Web, expanding to devices and things in everyday life, along with new trends in Artificial Intelligence call for new paradigms and a new look on Knowledge Representation and Processing at scale for the Semantic Web. The emerging, but still to be concretely shaped concept of “Knowledge Graphs” provides an excellent unifying metaphor for this current status of Semantic Web research. More than two decades of Semantic Web research provides a solid basis and a promising technology and standards stack to interlink data, ontologies and knowledge on the Web. However, neither are applications for Knowledge Graphs as such limited to Linked Open Data, nor are instantiations of Knowledge Graphs in enterprises — while often inspired by — limited to the core Semantic Web stack. This report documents the program and the outcomes of Dagstuhl Seminar 18371 "Knowledge Graphs: New Directions for Knowledge Representation on the Semantic Web", where a group of experts from academia and industry discussed fundamental questions around these topics for a week in early September 2018, including the following: what are knowledge graphs? Which applications do we see to emerge? Which open research questions still need be addressed and which technology gaps still need to be closed?

**Seminar** September 9–14, 2018 – http://www.dagstuhl.de/18371
**Keywords and phrases** knowledge graphs, knowledge representation, linked data, ontologies, semantic web
**Digital Object Identifier** 10.4230/DagRep.8.9.1

Knowledge Graphs: New Directions for Knowledge Representation on the Semantic Web, *Dagstuhl Reports*, Vol. 8, Issue 09, pp. 1–92
Editors: Piero Andrea Bonatti, Michael Cochez, Stefan Decker, Axel Polleres, and Valentina Presutti
Dagstuhl Reports
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

## 1 Table of Contents

**Introduction**

**Overview of Short Talks**

**Grand Challenges**

**Knowledge Graphs and the Web**

*Sarven Capadisli, Michael Cochez, Claudio Gutierrez, Andreas Harth, and Antoine Zimmermann*

## 2 Introduction

In 2001 Berners-Lee et al. stated that “The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation.”

The time since the publication of the paper and creation of the foundations for the Semantic Web can be roughly divided in three phases: The first phase focused on bringing Knowledge Representation to Web Standards, e.g., with the development of OWL. The second phase focused on data management, linked data and potential applications. In the third, more recent phase, with the emergence of real world applications and the Web emerging into devices and things, emphasis is put again on the notion of Knowledge, while maintaining the large graph aspect: Knowledge Graphs have numerous applications like semantic search based on entities and relations, disambiguation of natural language, deep reasoning (e.g. IBM Watson), machine reading (e.g. text summarization), entity consolidation for Big Data, and text analytics. Others are exploring the application of Knowledge Graphs in industrial and scientific applications.

The shared characteristic by all these applications can be expressed as a challenge: the capability of combining diverse (e.g. symbolic and staatistical) reasoning methods and knowledge representations while guaranteeing the required scalability, according to the reasoning task at hand. Methods include: Temporal knowledge and reasoning, Integrity constraints, Reasoning about contextual information and provenance, Probabilistic and fuzzy reasoning, Analogical reasoning, Reasoning with Prototypes and Defeasible Reasoning, Cognitive Frames, Ontology Design Patterns (ODP), and Neural Networks and other machine learning models.

With this Dagstuhl Seminar, we intend to bring together researchers that have faced and addressed the challenge of combining diverse reasoning methods and knowledge representations in different domains and for different tasks with Knowledge Graphs and Linked Data experts with the purpose of drawing a sound research roadmap towards defining scalable Knowledge Representation and Reasoning principles within a unifying Knowledge Graph framework. Driving questions include:

- What are fundamental Knowledge Representation and Reasoning methods for Knowledge Graphs?
- How should the various Knowledge Representation, logical symbolic reasoning, as well as statistical inference methods be combined and how should they interact?
- What are the roles of ontologies for Knowledge Graphs?
- How can existing data be ingested into a Knowledge Graph?

In order to answer these questions, the present seminar was aiming at cross-fertilization between research on different Knowledge Representation mechanisms, and also to help to identify the requirements for Knowledge Representation research originating from the deployment of Knowledge graphs and the discovery of new research problems motivated by applications. We foresee, from the results summarized in the present report, the establishment of a new research direction, which focuses on how to combine the results from knowledge representation research in several subfields for joint use for Knowledge Graphs and Data on the Web.

## The Seminar

The idea of this seminar emerged when the organisers got together discussing about writing a grant proposal. They all shared, although from different perspectives, the conviction that research on Semantic Web (and its scientific community) reached a critical point: it urged a paradigm shift. After almost two decades of research, the Semantic Web community established a strong identity and achieved important results. Nevertheless, the technologies resulting from its effort on the one hand have proven the potential of the Semantic web vision, but on the other hand became an impediment; a limiting constraint towards the next major breakthrough. In particular, Semantic Web knowledge representation models are insufficient to face many important challenges such as supporting artificial intelligence systems in showing advanced reasoning capabilities and socially-sound behavior at scale. The organisers soon realised that a project proposal was not the ideal tool for addressing this problem, which instead needed a confrontation of the Semantic Web scientific community with other relevant actors, in the field. From this discussion, the ”knowledge graph” concept emerged as a key unifying ingredient for this new form of knowledge representation – embracing both the Semantic Web, but also other adjacent communities – and it was agreed that a Dagstuhl seminar on ”Knowledge Graphs: New Directions for Knowledge Representation on the Semantic Web” was a perfect means for the purpose.

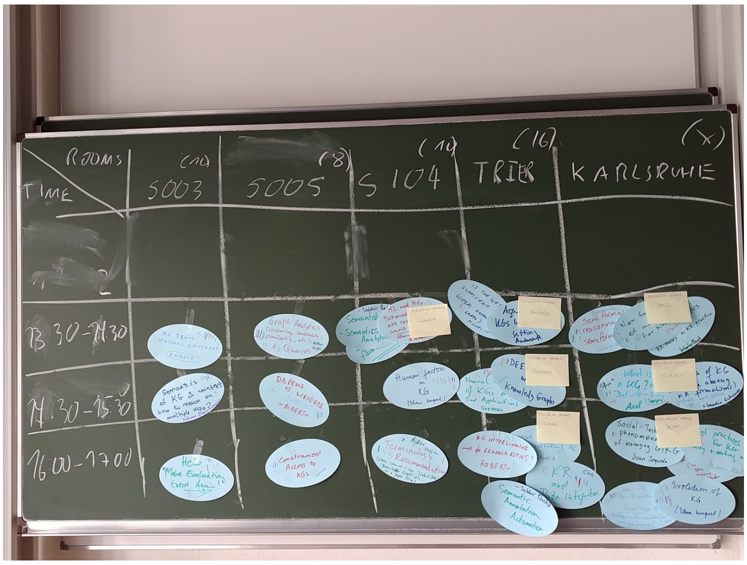
The list of invitees to the seminar included scientists from both academia and industry working on knowledge graphs, linked data, knowledge representation, machine learning, automated reasoning, natural language processing, data management, and other relevant areas. Forty people have participated in the seminar, which was very productive. The active discussions during plenary and break out sessions confirmed the complex nature of the proposed challenge. This report is a fair representative of the variety and complexity of the addressed topics.

The method used for organising the seminar deserves further elaboration. The seminar had a five-day agenda. Half of the morning on the first day was devoted to ten short talks (5 minutes each) given by a selection of attendees. The speakers were identified by the organisers as representatives of complimentary topics based on the result of a Survey conducted before the seminar: more than half of the invitees filled a questionnaire that gave them the opportunity to briefly express their perspectives on the topic and to point out relevant challenges that they would put in their future research agenda with the highest priority.

The aim of these short speeches was to ignite the confrontation by sharing the emerging views on the main challenges from this survey. After the speeches we organised the further discussion in an “Open Space” session that served to collaboratively build the agenda for the rest of the day (and that influenced the agenda of the next days). The open space method consists of giving everyone the opportunity to propose one or more break out topics. To propose a topic, a proposer had to explain in few words what it was about, then write it down on a post-it that was attached on a blackboard (see Figure 1). At the end of the session, attendees were invited to sign up for the topics of their interest (by marking the corresponding post-it).

The more popular ones (up to fifteen and having at least three sign ups) were selected to compose the agenda. Each break out session used a one-hour slot during the afternoon. The second day continued with most of the break out sessions with the aim of continuing the discussion started the first day and work towards consolidating a report (finalised on the fourth day). Reports would reflect view and vision emerging from the break out group. On the same day attendees had the opportunity to self-propose to give additional short

**Figure 1** Blackboard with post-its from the open session.



speeches, addressing missing relevant topics. We used part of the second day's morning for these speeches. We explicitly asked attendees to avoid speeches on "my research" and to only address relevant challenges that were overlooked so far.

On the third day we started with a plenary discussion and the seminar group agreed on splitting into four groups to discuss "Grand challenges" separately, then share the results before going back to the break out sessions. The aim was to share a common high level vision reference before consolidating the more specific discussions that were ongoing in the break out sessions. On the fourth day, the seminar group split again in break out sessions including a "Grand challenges" one. Each session was assigned to at least two coordinators, who committed to consolidate in a draft report the results from the previous meetings. It was decided to merge a few topics, when appropriate.

Break out sessions had varied level of technical abstraction depending on the nature of the topic, and its level of maturity within the state of the art. To give some examples: the break out session about "Grand challenges" mainly discussed a vision for a future research agenda and maintained a high level of abstraction, while the session on "Human and Social Factors in Knowledge Graphs" provided more concrete insights as it could build on both academic and industrial research results, projects and practical experiences. The session on "Applications of Knowledge Graphs" focused on technical details and issues on two relevant sample applications.

Piero Andrea Bonatti
Michael Cochez
Stefan Decker
Axel Polleres
Valentina Presutti

## 2.1 Overview of the Report

This report is organised in two main parts: Section 3 includes a list of abstracts providing an overview of the short speeches that we had the first two days. All the other sections are consolidated reports of the emerging vision, research challenges, possible research agenda, and proposed approaches, from break out sessions. When applicable, the reports give an overview of specific relevant research work.

## 2.2 Seminar Participants

**Figure 2** Group picture.

Wouter Beek (Vrije Universiteit Amsterdam, NL), Christian Bizer (Universität Mannheim, DE), Eva Blomqvist (Linköping University, SE), Piero Andrea Bonatti (University of Naples, IT), Dan Brickley (Google Research - Mountain View, US), Sarven Capadisli (TIB - Hannover, DE), Michael Cochez (Fraunhofer FIT - Sankt Augustin, DE), Claudia d'Amato (University of Bari, IT), Gerard de Melo (Rutgers University - Piscataway, US), Stefan Decker (RWTH Aachen, DE), Michel Dumontier (Maastricht University, NL), Paul Groth (University of Amsterdam, NL & Elsevier Labs, NL), Claudio Gutierrez (IMFD, DCC, University of Chile - Santiago de Chile, CL), Andreas Harth (Fraunhofer IIS - Nürnberg, DE), Aidan Hogan (IMFD, DCC, University of Chile - Santiago de Chile, CL), Sabrina Kirrane (Wirtschaftsuniversität Wien, AT), Markus Krötzsch (TU Dresden, DE), Barend Mons (Leiden University Medical Center, NL), Roberto Navigli (Sapienza University of Rome, IT), Sebastian Neumaier (Wirtschaftsuniversität Wien, AT), Axel-Cyrille Ngonga-Ngomo (Universität Paderborn, DE), Andrea Giovanni Nuzzolese (CNR - Rome, IT), Heiko Paulheim (Universität Mannheim, DE), Lydia Pintscher (Wikimedia Deutschland, DE), Axel Polleres (Wirtschaftsuniversität Wien, AT), Valentina Presutti (STLab, ISTC-CNR, IT), Sabbir Rashid (Rensselaer Polytechnic Institute - Troy, US), Sebastian Rudolph (TU Dresden, DE), Marta Sabou (TU Wien, AT), Juan F. Sequeda (Capsenta Inc. - Austin, US), Cogan Matthew Shimizu (Wright State University - Dayton, US), Elena Simperl (University of Southampton, GB), Dezhao Song (Thomson Reuters - Eagan, US), Steffen Staab (Universität Koblenz-Landau, DE), Volker Tresp (Siemens AG - München, DE), Marieke van Erp (KNAW Humanities Cluster - Amsterdam, NL), Frank van Harmelen (Vrije Universiteit Amsterdam, NL), Maria-Esther Vidal (TIB - Hannover, DE), Michael Witbrock (IBM Research - Yorktown Heights, US), Sonja Zillner (Siemens AG - München, DE), and Antoine Zimmermann (École des Mines de Saint-Étienne, FR)

# 3 Overview of Short Talks

## 3.1 Evolution and dynamics

*Eva Blomqvist (Linköping University, SE)*

Nobody would today consider the Web as a static thing. Similarly, knowledge in a company is never static, it is constantly changing. So why is it that so many approaches developed in the past decades make the assumption that knowledge on the web, or elsewhere, is not going to change? At an early stage this can be a way to manage the complexity of a problem, simply to get started, but we cannot afford to use that excuse any more – if we want to be taken seriously by developers and users out in industry. New methods, technologies and standards that we produce, or propose, need to start from this assumption, i.e., that everything is dynamic, and build on that, rather than ignoring it and then potentially trying to cover it by add-ons at the end. Knowledge graphs in a highly dynamic environment necessarily needs to themselves be highly dynamic and constantly evolving, and we are the ones that have to provide the technology to support that evolution!

## 3.2 Enabling Accessible Scholarly Knowledge Graphs

*Sarven Capadisli (TIB - Hannover, DE)*

Scholarly knowledge includes a range of research artefacts that needs to be described. These include research articles, peer reviews, research data, social interactions like review requests and notifications in general, as well as different kinds of annotations on research objects. The current state of access and use of scholarly knowledge is insufficient for society at large. By enabling accessible scholarly knowledge graphs as well as applications which make use of it, we hope to enable universal access to previous research. By improving the availability through linked research, we can facilitate discovery and building on existing research. A fundamental first step is to investigate and develop effective ways to represent fine-grained information that is accessible, human and machine-interpretable, and interconnected. Other challenges look into ways in which academic journals can decouple the registration and certification functions of scholarly communication. Also we can investigate the feasibility of universal access to knowledge through decentralisation, freedom of expression, privacy respecting, and social.

### 3.3 Logic and learning – Can we provide Explanations in the current Knowledge Lake?

*Claudia d'Amato (University of Bari, IT)*

The goal of the talk is to raise the attention on the following research questions: a) is it important to provide explanations when providing information by exploiting a Knowledge Graph (KG)? b) Would it be possible to design integrated numeric and symbol-based machine learning methods, to be used for tackling the link prediction problem, that are scalable and able to provide interpretable models? c) Are interpretable models a sufficient form of explanation or do we need to provide an actual line of reasoning, illustrating the decision making process? d) Is it possible to develop a unified framework integrating different reasoning paradigms?

A KG is often defined as an object for describing entities and their interrelations, by means of a graph. Knowledge graphs are usually assumed to be large and arbitrary entities may be interrelated, thus covering various topical domains [1]. The importance of assessing relations among entities has driven research on developing effective methods for solving the link prediction problem. This is often regarded as a classification problem that can be solved by the use of machine learning classification methods. In the last few years, vector space embedding solutions have been largely adopted [2, 3]. They allow to create propositional feature vector representations starting from more complex representations, such as graphs, thus allowing to apply numeric approaches resulting scalable besides of effective. The main problem of numeric approaches is that they do not allow to provide somehow an explanation of the predicted results, that is, similarly to the goal of "Explainable AI" research field (which aims to produce "glass box" models that are explainable to a human, without greatly sacrificing performances), an explanation of the reason why a certain entity is predicted as being related (with respect to a given relation) to another one. The exploitation of symbol-based learning methods would be useful at this regards since they are known to generate interpretable models that allow to explain how conclusions are drawn [4, 5]. Nevertheless, symbol-based learning methods are also known to be less scalable than numeric methods. As such integrated numeric and symbol-based approaches need to be developed in order to come up with interpretable models whilst still staying scalable. Such an integrated solution would be an initial step ahead towards providing explanation. Indeed interpretable models actually describe how solutions are obtained but not why they are obtained. Providing an actual explanation means to formulate and supply a line of reasoning, illustrating the decision making process of a model whilst using human understandable features of the input data. The ability of performing reasoning is important not only for providing explanations. KGs are often considered as the output of an information acquisition and integration process, where information may come from several and different sources. As such, problems such as noise and conflicting information may arise. Additionally, some acquired information could be valid only in some contexts or with respect to a certain period of time. As such the ability to apply different reasoning paradigms such deductive reasoning, paraconsistent reasoning, inductive reasoning, normative reasoning, analogical reasoning could be necessary. Large research efforts have been devoted to study each reasoning paradigm, however, when considering large KGs coming from the integration of multiple sources of information, multiple reasoning paradigms could be needed at the same time. As such a unified framework integrating different reasoning paradigms needs to be formalized.

## References

1 H. Paulheim. Knowledge graph refinement: a survey of approaches and evaluation methods. *Semantic Web Journal*, 8(3):489–508, IOS Press, 2017.

2 P. Minervini and C. d'Amato and N. Fanizzi. Efficient energy-based embedding models for link prediction in knowledge graphs. *Journal of Intelligent Information Systestems*, 47(1):91–109, 2016.

3 M. Cochez and P. Ristoski and S. P. Ponzetto and H. Paulheim. Global RDF Vector Space Embeddings. In C. d'Amato et. al (eds.). *The Semantic Web - ISWC 2017 - 16th International Semantic Web Conference (2017), Proceedings, Part I* volume 10587 of *LNCS*, pages 190–207. Springer, 2017.

4 G. F. Luger. Arti1cial Intelligence: Structures and Strategies for Complex Problem Solving. Addison Wesley, 5 edition, 2005.

5 L. De Raedt. Logical and Relational Learning: From ILP to MRDM (Cognitive Technologies) Springer-Verlag, 2008.

6 L. Galárraga, C. Teflioudi, F. Suchanek, and K. Hose. AMIE: Association rule mining under incomplete evidence in ontological knowledge bases. In *Proceedings of the 20th International World Wide Web Conference (WWW 2013)*. ACM, 2013.

7 F. A. Lisi. AL-QuIn: An onto-relational learning system for semantic web mining. *International Journal of Semantic Web and Information Systems*, 2011.

8 J. Józefowska, A. Lawrynowicz, and T. Lukaszewski. The role of semantics in mining frequent patterns from knowledge bases in description logics with rules. *Theory and Practice of Logic Programming*, 10(3):251–289, 2010.

9 J. Völker and M. Niepert. Statistical schema induction. In G. Antoniou et al., editors, *The Semantic Web: Research and Applications - 8th Extended Semantic Web Conference, (ESWC 2011), Proc., Part I*, volume 6643 of *LNCS*, pages 124–138. Springer, 2011.

10 A. K. Joshi and P. Hitzler and G. Dong Logical Linked Data Compression In *The Semantic Web: Research and Applications - 10th Extended Semantic Web Conference, (ESWC 2013), Proceedings*, volume 7882 of *LNCS*, pages 170–184. Springer, 2013.

11 L. Dehaspeand and H. Toironen. Discovery of frequent datalog patterns. *Data Mining and Knowledge Discovery*, 3(1):7–36, 1999.

12 B. Goethals and J. Van den Bussche. Relational association rules: Getting warmer. In *Proceedings of the International Workshop on Pattern Detection and Discovery*, volume 2447 of *LNCS*, pages 125–139. Springer, 2002.

13 C. d'Amato and V. Bryl and L. Serafini. Semantic Knowledge Discovery and Data-Driven Logical Reasoning from Heterogeneous Data Sources. In F. Bobillo et al.(Eds.), ISWC International Workshops, URSW 2011-2013, Revised Selected Papers, vol. 8816 Springer, LNCS/LNAI (2014).

14 R. Agrawal, T. Imielinski, and A. N. Swami. Mining association rules between sets of items in large databases. In P. Buneman and S. Jajodia, editors, *Proc. of the 1993 ACM SIGMOD Int. Conf. on Management of Data*, pages 207–216. ACM Press, 1993.

15 C. d'Amato and V. Bryl and L. Serafini. Data-Driven Logical Reasoning. In Proc. of the 8th Int. Workshop on Uncertainty Reasoning for the Semantic Web (URSW'12), vol. 900, CEUR (2012).

## 3.4 Knowledge graph creation and management

*Michel Dumontier (Maastricht University, NL)*

While there are many manifestations of knowledge graphs (KGs), there are few guidelines as to how to create them or make them widely available in a reliable manner owing to ambiguity in their definition. In the most basic sense, KGs represent some form of knowledge that is amenable to processing by graph or reasoning algorithms, in which entities are related to their attributes and to other entities, along with provenance of where that knowledge was obtained. KGs are created through a myriad of approaches - be it manual, automatic, or semi-automatic - using a variety of data sources such as textual documents, microdata embedded in web pages, large and small databases, and crowdsourced statements. They are subject to a wide variety of data processing activities such as mapping entities to concepts, extracting relations from text, transforming facts to specific formats for indexing, integrating vastly different data sources, and finding errors through quality assessment. All KG creation methods have their advantages and disadvantages, and can often create vastly different KGs that can have important implications in downstream applications such as answering questions, offering recommendations, and making new predictions. Clearly, there remain great challenges towards organizing the emerging KG community in making their KGs FAIR - Findable, Accessible, Interoperable, and Reusable - to the benefit of both humans and machines.

## 3.5 New Symbol Groundings for Knowledge Graphs

*Paul Groth (University of Amsterdam, NL & Elsevier Labs, NL)*

Today’s knowledge graphs primarily use social systems in combination with logics to define the meaning of their symbols. For example, a knowledge graph might use rdfs:subClassOf to define a subclass hierarchy - the meaning of which is found by dereferencing to a document on the web and then interpreting the natural language and mathematical definitions found at the location. I suggest that we should instead think about grounding the symbols of a knowledge graph (the entities and relations) in other mediums. For example, one can think about grounding these symbols in a sub-symbolic space (e.g. vector embeddings). Likewise, it is possible to ground symbols in physical reality through sensors or in shared simulations. Adopting these other forms of groundings would allow for more expressive knowledge graphs. There are a number of sets of related work that provide good routes into these alternative mechanisms. The work of Douwe Kiela is highly relevant as discussed in his thesis “Deep embodiment: grounding semantics in perceptual modalities”. The work of Cynthia Matuszek on grounded language acquisition using robotics is also highly relevant. Additionally, resources such as visualgenome.org enable the connection of symbols and images. Overall, combining these lines of work with Knowledge Graphs can provide a rich set of new research avenues around integration, reasoning, use and exchange.

## 3.6 Cultural issues in multilingual knowledge graph

*Roberto Navigli (Sapienza University of Rome, IT)*

When dealing with multilingual lexicalized knowledge graphs, such as Wikidata or BabelNet, a number of issues arise, including the impossibility to lexicalize a certain concept in a given language (e.g. ikigai from Japanese; gezellig from Dutch), the different perception of the same concept in different cultures (e.g. copyright in the UK vs. Germany) or the granularity of sense distinctions. All these issues need to be addressed in upcoming research of multlingual KGs.

## 3.7 Quality and Evaluation of Knowledge Graphs (beyond DBpedia)

*Heiko Paulheim (Universität Mannheim, DE)*

Various metrics for the quality evaluation of knowledge graphs have been proposed. Zaveri et al. [1] propose a set of 17 metrics, focusing mostly on technical and legal dimensions of the data and Linked Data recommendations. They cluster the metrics into availability, licensing, interlinking, security, and performance. Färber et al. [2] come up with a broader collection, encompassing accuracy, trustworthiness, consistency, relevancy, completeness, timeliness, ease of understanding, interoperability, accessibility, licensing, and interlinking. Looking not only at those papers, but also at the open reviews reveals that defining objective metrics for KG quality is a challenging endeavour. Despite the mere analysis of the quality of existing knowledge graphs, various methods for improving the quality of those knowledge graphs have been proposed as well, which we have reviewed in [3]. In that article, we do not only review more than 40 approaches of KG completion and error detection, but also shed more light on the evaluation.

Some of the key findings of the survey include:

1. Although KG completion and error detection seem related, there are rarely any approaches that tackle both tasks simultaneously.
2. Likewise, although quite a few approaches deal with error detection, error correction is hardly addressed.
3. DBpedia is the most used KG for evaluation. Many papers only report on DBpedia, hence making it difficult to derive general applicability of the proposed approaches.
4. There is a large variety of evaluation setups, ranging from split validation to cross validation with various splits and foldings, and a large number of different metrics used aside precision, recall, and F1-score. Due to those differences in the setup, it is hard to compare results between different papers directly.
5. Scalability evaluations are still rare; almost half of the papers do not mention scalability at all.

Following up on those observations, there are some research question that we identfy worthwhile diving into:

1. Which quality improvements does the community deem the most necessary (e.g., completeness, correctness, linkage, ...)

2. How can we come up with standardized evaluation setups for KGs and KG completion/-correction methods?
3. How can we best preserve the efforts made towards KG improvements?

**References**

1 A. Zaveri, A. Rula, A. Maurino, R. Pietrobon, J. Lehmann, and S. Auer: Quality Assessment for Linked Open Data: A Survey. SWJ 7(1), 2016

2 Michael Färber, Frederic Bartscherer, Carsten Menne, and Achim Rettinger: Linked data quality of DBpedia, Freebase, OpenCyc, Wikidata, and YAGO SWJ 9(1), 2018

3 H. Paulheim: Knowledge Graph Refinement – A Survey of Approaches and Evaluation Methods. SWJ 8(3), 2017

## 3.8 Humans in the Loop, Human readable KG

*Marta Sabou (TU Wien, AT)*

We consider two interaction interfaces between humans and knowledge graphs. On the one hand, during the process of knowledge acquisition and verification, humans act as sources for different types of knowledge, including, factual, expert, or social knowledge. Mechanisms for acquiring such knowledge are diverse and range from manual approaches, to semi-automatic and human-in-the loop systems where algorithmic and human computation are intertwined (e.g., through active learning). On the other hand, knowledge graphs enable various information seeking tasks, such as question answering, search (semantic, exploratory, serendipitous) and conversational systems (e.g., chatbots). Technical challenges in these interaction settings arise from dealing with large KGs and from the need to adapt generic methods to domain/enterprise specific scenarios. Opportunities arise in terms of being able to collect viewpoints, opinions etc from humans that enable the creation of more realistic applications. Additionally, knowledge graphs enable a range of new applications (such as sense making) which should be built by relying on cognitive science theories to maximize the effectiveness of the information transfer to humans. Besides technical challenges, ethical issues should be considered when involving humans in KG creation processes (e.g., through crowdsourcing) as well as for ensuring correct, unbiased and diversity aware output of applications built on top of KGs.

## 3.9 ML with KGs – research and use cases around KGs at Siemens

*Volker Tresp (Siemens AG - München, DE)*

Labeled graphs can describe states and events at a cognitive abstraction level, representing facts as subject-predicate-object triples. A prominent and very successful example is the Google Knowledge Graph, representing on the order of 100B facts. Labeled graphs can be represented as adjacency tensors which can serve as inputs for prediction and decision making, and from which tensor models can be derived to generalize to unseen facts. These ideas can be used, together with deep recurrent networks, for clinical decision support by predicting orders and outcomes. Following Goethe's proverb, "you only see what you know", we presented how background knowledge can dramatically improve information extraction from images by deep convolutional networks and how tensor train models can be used for the efficient classification of videos. We discussed potential links to the memory and perceptual systems of the human brain. We concluded that tensor models, in connection with deep learning, can be the basis for many technical solutions requiring memory and perception, and might be a basis for modern AI.

## 3.10 Privacy and constrained access

*Sabrina Kirrane (WU Wien, AT)*

Irrespective of whether the goal is to provide open access to a knowledge graph or to constrain access to the graph or a subset of the knowledge held therein, policies have an important role to play. For instance, if a data publisher does not specify a license the default is all rights reserved. a company may wish to restrict access to their Enterprise knowledge graph and likewise individuals may exercise there rights to specify how there data should be used and by whom. There are already several existing ontologies and policy languages that could be leveraged ranging from general policy languages, to specific policy languages, including some standardisation efforts, however the expressivity, correctness and completeness with respect to specific use case requirements is still and open research challenge. Although it may be possible to employ existing encryption and anonymisation techniques to knowledge graphs, the utility of the knowledge will most certainly be compromised. Constraints are a fact of life. Therefore we need to figure out how to deal with them!

## 3.11 Value Proposition of Knowledge Graphs

*Sonja Zillner, (Siemens AG, DE)*

We often talk about the value of knowledge graphs. But what is their main value proposition and what is their USP? In industrial settings, knowledge graphs are an important asset for realizing industrial Artificial Intelligence (AI) applications. Through the combination of both, knowledge graphs that capture relevant domain knowhow and AI algorithms that reason and learn to solve problems or answer questions, augmented intelligence applications supporting users to focus on ambitious and creative instead of repetitive tasks can be developed. Examples range from the improved visualization of radiological findings to advanced diagnostics systems for power plants to flexible manufacturing for Industry 4.0 applications. But is there also a clear correlation between the type of a knowledge graph's value proposition and its addressed technical requirements?

## 3.12 Social-Technical Phenomena of (Enterprise) Knowledge Graph Management

*Juan F. Sequeda, (Capsenta, USA)*

An early vision in Computer Science has been to create intelligent systems combining Knowledge and Data capable of reasoning on large amounts of data. Today, this vision is starting to be fulfilled through Knowledge Graphs.

Even though we are starting to see adoption of Knowledge Graphs by the large enterprises, we are also observing barriers for adoption by small and medium enterprises. It is important to understand why and see if there are new scientific problems and opportunities.

We argue that these barriers are not just technical/engineering but also social. For example, we lack tools for non-experts, methodologies to design conceptual models together with mappings, understanding who are the different stakeholders and the roles they can/should play. Therefore it is important to study and understand the socio-technical phenomena of managing (creation, maintenance, evolution, etc) Knowledge Graphs.

## 3.13 Concise account of the notion of Knowledge Graph

*Claudio Gutierrez (University of Chile - Santiago de Chile, CL)*

### Brief origins of the notion.

If one would try to find a footprint of the prehistory of the notion of knowledge graph (KG), it would be the idea of representing knowledge in a diagrammatic form, in people like Aristotle ($\sim$ 350 BC), Sylvester (1878 [11]) , Peirce (1878 [6]), Frege (1879 [2]), etc.

The origins of the modern idea can be traced back to Ritchens (1956 [8]), Qullian (1963 [7]) and Milgram (1967 [12]). From a formal point of view, it was very influential the introduction of the notion of frames (M. Minsky (1974 [4]) *A Framework for representing knowledge*); the formalization of semantic networks (W. A. Woods (1978 [13]), *What's in a Link: Foundations for Semantic Networks*); and the notion of conceptual graphs (J. Sowa (1979 [9]), *Semantics of Conceptual Graphs*).

A systematic study involving KG is the Ph.D. Thesis of R. R. Bakker, *Knowledge Graphs: representation and structuring of scientific knowledge* in 1987 [1]. Many of these ideas were published in 1992 in a paper authored by P. James (a name representing many researchers) and titled *Knowledge Graphs* [3].

Twenty years later, in 2012, Google popularized the notion worldwide with the patent *Knowledge graph based search system*, a system described as follows:

> "[...]a novel, useful system that develops and maintains one or more individual and/or group contexts in a systematic fashion and uses the one or more contexts to develop a Personalized Medicine Service [...] The innovative system of the present invention supports the development and integration of any combination of data, information and knowledge from systems that analyze, monitor, support and/or are associated with entities in three distinct areas:[...]"

### What is really a Knowledge Graph?

John Sowa wrote in the entry *Semantic networks* in the Encyclopedia of Cognitive Science (1987 [10]):

> "Woods (1975) and McDermott (1976) observed, the semantic networks themselves have no well-defined semantics. Standard predicate calculus does have a precisely defined, model theoretic semantics; it is adequate for describing mathematical theories with a closed set of axioms. But the real world is messy, incompletely explored, and full of unexpected surprises."

P. James, mentioned before, defined a Knowledge Graph as follows:

> "A knowledge graph is a kind of semantic network. [...] One of the essential differences between knowledge graphs and semantic networks is the explicit choice of only a few types of relations."

Later Lei Zhang in a Ph.D. thesis titled *Knowledge Graph Theory and Structural Parsing* (2002 [14]) defined:

> "Knowledge graph theory is a kind of new viewpoint, which is used to describe human language [...] knowledge graphs have advantages, which are stronger ability to express, to depict deeper semantic layers, to use a minimum relation set and to imitate the cognition course of mankind etc. Its appearance gave a new way to the research of computer understanding of human language."

The Google Patent (2012) referred above conceptualized KG as systems:

> "The system of the present invention systematically develops the one or more complete contexts for distribution in a Personalized Medicine Service. These contexts are in turn used to support the comprehensive analysis of subject performance, develop one or more shared contexts to support collaboration, simulate subject performance and/or turn data into knowledge."

More recently, M. Nickel, K. Murphy, V. Tresp and E. Gabrilovich in their article *A Review of Relational Machine Learning for Knowledge Graph* (2016 [5]) defined KG as follows:

> "[...] a graph structured knowledge bases (KBs) that store factual information in form of relationship between entities."

In summary, we learned that a knowledge graph is a kind of semantic network, an artifact whose scope, characteristic, features, even uses, remain open and are in the process of being defined. The brief history presented above suggests that, to design the future of the field, it would be valuable to climb on the shoulders of three giant notions: *Frames, Semantic Networks* and *Conceptual Graphs.*

#### References

1 R. R. Bakker. *Knowledge Graphs: Representation and Structuring of Scientific Knowledge.* Ph.D. Thesis, University of Twente, 1987.

2 G. Frege. *Begriffsschrift.* Halle, 1879.

3 P. James. Knowledge graphs. *Linguistic Instruments in Knowledge Engineering. Elsevier Publ.*, 1992.

4 M. Minsky. A framework for representing knowledge. *MIT-AI Memo 306, Santa Monica*, 1974.

5 M. Nickel, K. Murphy, V. Tresp, and E. Gabrilovich. A review of relational machine learning for knowledge graphs. *Proc. of the IEEE*, 2016.

6 C. S. Peirce. How to make our ideas clear. *Popular Science Monthly 12*, 1878.

7 R. Quillian. A notation for representing conceptual information: An application to semantics and mechanical English paraphrasing. Systems Development Corp. *Santa Monica*, 1963.

8 R. H. Ritchens. General program for mechanical translation between any two languages via an algebraic interlingua. *Report on Research: Cambridge Language Research Unit. Mechanical Translation 3 (2)*, 1956.

9 J. Sowa. Semantics of conceptual graphs. *Proc. 17th. ACL*, 1979.

10 J. Sowa. Semantic networks. *In: Encyclopedia of Cognitive Science*, 1987.

11 J. J. Sylvester. Chemistry and algebra. *Nature 17: 284*, 1878.

12 Jeffrey Travers and Stanley Milgram. An experimental study of the small world problem. *Sociometry, Vol. 32, No. 4*, 1967.

13 W. A. Woods. What's in a link: Foundations for semantic networks. *Representation and Understanding. Studies in Cognitive Science, 35-82.*, 1978.

14 L. Zhang. *Knowledge Graph Theory and Structural Parsing.* Ph.D. Thesis, University of Twente, 2002.

# 4 Grand Challenges

*Paul Groth (University of Amsterdam, NL & Elsevier Labs, NL), Frank van Harmelen (Free University Amsterdam, NL), Axel-Cyrille Ngonga-Ngomo (Universität Paderborn, DE), Valentina Presutti (CNR - Rome, IT), Juan F. Sequeda (Capsenta Inc. - Austin, US), and Michel Dumontier (Maastricht University, NL)*

The emergence of large scale knowledge graphs (KGs) has opened up the possibility for a wide-range of exciting research directions. This chapter attempts to capture some of these large scale challenges. The participants tried to formulate challenges that pushed the boundaries of current thinking. We clustered 14 specific challenges into four groups:

1. Representing Knowledge i.e. "How do you say that?"
2. Access and interoperability at scale i.e "How do you get that?"
3. Applications i.e. "How do you do that?"
4. Machine ⇔ Humanity Knowledge Sharing i.e ."Computamus ergo sumus"

These clusters range from deep technical challenges that are being tackled today (e.g. the connection between subsymbolic and symbolic representations) to future visions where knowledge graphs act as the communication mechanism between humanity and AI. We see these challenges not as the definitive list but as inspiration for the broader community to think about where the foundations of knowledge graphs can take us.

During the discussion, one frame that helped us was to think about the notion of knowledge and data at scale. We begin by introducing that context. We then list the challenges themselves.

## 4.1 Context: the structure of knowledge & data at scale

There are credible models of data at scale (from Data Management). There are credible models of knowledge (from Knowledge Representation and Knowledge Engineering), but there are very few credible models of *knowledge and data at scale.* (where "scale" is with respect to each of the V's: volume, variety, veracity, velocity, etc).

Our object of study is therefore the structure of *knowledge and data at scale.*

As in any scientific field, we distinguish the following three layers:

- **Models:** a model of the structure of knowledge and data at scale (e.g. knowledge graphs)
- **Manifestations:** instances of such a model (e.g. Wikidata, DBpedia)
- **Applications:** methods and tools that rely on such instances (e.g. "search engine", "recommender system', "data interoperability","knowledge integration system", "knowledge discovery")

In the light of the above, what the community has done is to develop a model for the structure of knowledge & data at scale, namely: Knowledge Graphs.

Multiple models for the structure of knowledge and data at scale have been proposed:

1. Relation algebra ("it's a table!")
2. Knowledge graphs ("it's a graph!")
3. Latent semantics ("it's a vector space!")
4. World Wide Web ("it's a network of documents!")

Thus, current knowledge graphs are just one model for the "structure of knowledge & data at scale", and alternative and complementary models have been studied and will no doubt be proposed in the future. Each of these will explain/describe/implement other aspects of the structure of knowledge & data at scale. Essential to our understanding of the structure of knowledge & data at scale is knowing when to use which of these complementary and sometimes competing models.

Any model has to be testable, in order to distinguish it from other models and to decide when which model is most appropriate. A model for **knowledge and data at scale** can be tested against:

- Theoretical properties (Kuhn's properties of scientific theories: accuracy, internal consistency, scope, simplicity; as well as properties specific for models of knowledge and data at scale such as robustness, graceful degradation and others)
- Performance in a task at scale (the V's): how well does this model support performance of a particular task, measured in various ways: ease of design, ease of implementation, ease of maintenance, does it combine with other models, and others).
- Performance for users: supporting useful abstractions for presentation, explainability, etc.
- Cognitive "convenience"
- Occam's razor (as always)

These and other criteria will allow models to be compared against other models, telling us when to use which model. When a model falls short on some of these criteria, that is a prompt to improve or extend the model. For example, current knowledge graphs fall short on representing time, versioning, probability, fuzziness, context, reification, and handling inconsistency among others. New generations of knowledge graph models should explain/describe/implement these and other aspects of the structure of "knowledge & data at scale".

The challenges clustered below can be thought of as directions in the exploration of knowledge & data at scale.

## 4.2 Representing Knowledge

How do you say that?

**Diversity and flexibility in methods for knowledge representation.** To have at our disposal a plurality of knowledge representations (from logics to embeddings, and others) to effectively capture all forms of knowledge including ambiguous, inconsistent, incomplete, erroneous, biased, diverging, approximate, contested, and context-specific knowledge.

**How do we capture and represent change within Knowledge Graphs?** Knowledge Graphs today are primarily entity centric. The goal here is develop new formalisms, algorithms, techniques to handle the evolution and change of events, languages, and entities.

**Symbolic meets subsymbolic (KG + ML):** Knowledge graphs represent knowledge by means which generalize well but lack the flexibility of more fuzzy models to knowledge representation such as those used by connectionist paradigms. Connectionist approaches on the other hand fail to generalize and lack explainability. The is a need to enable connectionist ML to consume and generate knowledge graphs while allowing for knowledge to represent and infer upon knowledge stored in connectionist models.

## 4.3 Access and interoperability at scale

How do you get that?

**Interoperable knowledge graphs:** A large number of KGs are already available and they capture a large portion of the knowledge of domains such as products, persons, locations, etc. However, these knowledge graphs are available in heterogeneous formats with partly incompatible semantics. The multitude of formats and semantics is likely to persist. How can we create a universe of knowledge graphs with different semantics that can interoperate and represent all knowledge necessary for both humans and AIs. Related questions include scalable partial/incomplete reasoning under these constraints as well the need to abide by practical restrictions while using KGs.

**A Public FAIR Knowledge Graph of Everything.** We increasingly see the creation of knowledge graphs that capture information about the entirety of a class of entities. For example, Amazon is creating a knowledge graph of all products in the world and Google and Apple have both created knowledge graphs of all locations in the world. This grand challenge extends this further by asking if we can create a knowledge graph of "everything" ranging from common sense concepts to location based entities. This knowledge graph should be "open to the public" in a FAIR manner democratizing this mass amount of knowledge.

**Uniform computational access to knowledge-based services.** Can we access all forms of knowledge, whether previously stated or inferred by computational service) through a common interface, thereby reducing the barrier to finding and using knowledge at the time it is needed?

**Rapid task-performant reindexing of knowledge.** Successful execution of particular tasks (finding relevant datasets, predicting new drug uses, etc) may require transformation of knowledge to other representations that are better suited for the task. Can we create an infrastructure to facilitate this repurposing of global knowledge?

## 4.4 Applications

How do you do that?

**Answering sophisticated questions over heterogeneous knowledge graphs.** Can we answer sophisticated, context-sensitive questions over different knowledge graphs with different formalisms, languages, schemas, content, availability, restrictions, access methods?

**Make the translation of knowledge to praxis instantaneous.** Currently, knowledge is transferred to practice through complex chains where humans translate knowledge into software and physical systems. With the advent of cyber-physical systems (e.g. augmented reality, IoT), there is the potential to directly translate knowledge into action. Thus, the vision is that gap between knowing and practicing will approach zero.

**Knowledge graphs as socio-technical systems.** Graphs and their applications are largely created by people. We need to leverage theory, methods and empirical evidence from other disciplines (behavioural economics, CSCW, UX, cognitive psychology etc) to:

Understand the cognitive and social processes by which knowledge (and knowledge shaped as a graph) emerges; Identify patterns and best practices to support these processes; Improve developer experience to allow them to create, curate and reuse KGs effectively; Provide guidelines and best practices to help developers use and appreciate large-scale KGs that are inherently messy, diverse and evolving; Understand what social features (expertise of KG contributors, their motivations, group composition) influence the outcomes (completeness of the knowledge, how it is represented, what is missing, viewpoints and opinions etc.)

## 4.5 Machine ⇔ Humanity Knowledge Sharing

Computamus ergo sumus (We compute, therefore we are).

**Knowledge graphs as an interface between humanity and machines as well as machines and machines.** As machines become generators of knowledge how do we enable communication between those machines and humanity as whole. Likewise, just as humans share their knowledge through institutions such as libraries how do we enable machines and humans to share their knowledge at scale.

**Generating, grounding, translating, and using machine generated languages.** Recent work suggests how machines may create their own languages that are distinct from those that we know and understand. How can we explain or translate statements made in those languages to other (human / non-human) languages?

**Natural Data Understanding:** The plethora of data formats and implicit semantics required universal machinery which can consume arbitrary data (incl. KGs) and generate KGs. All existing efforts in natural language understanding, processing of web tables, etc. can be regarded as a foundation for this effort.

**Self-aware KGs:** What should KGs be? They are currently regarded as the result of some (partly continuous) knowledge extraction processes. One possible path towards knowledge graphs being universal enablers for agent-agent interchange (where agent = humans + machines) would be to regard them as biological entities, which live in a digital space. These universal independent social agents would be able to interface with other agents (humans, machines) to fulfill goals set externally or internally.

# 5 On the Creation of Knowledge Graphs: A Report on Best Practices and their Future

*Sabbir Rashid (Rensselaer Polytechnic Institute - Troy, US) Eva Blomqvist (Linköping University, SE), Cogan Matthew Shimizu (Wright State University - Dayton, US), and Michel Dumontier (Maastricht University, NL),*

Knowledge Graphs have an important role in organizing and making information more broadly available to people and machines. While many knowledge graphs have been developed, the approaches used to build them can differ substantially. The elusiveness of standards or best practices in this area poses a substantial challenge to the knowledge engineer that wants to maximize their discovery and reuse, as dictated by the FAIR (Findable, Accessible, Interoperable, Reusable) principles. In this chapter, we define a set of best practices to constructing FAIR knowledge graphs.

## 5.1 Introduction

A knowledge graph (KG) is a conceptual entity that, as the name implies, is a graph structure that represents knowledge [13]. Knowledge graphs have been used for a variety of tasks, including question answering, relationship prediction, and searching for similar items. A knowledge graph has several components, including resources that convey attributes and entities, relationships between such resources, and additionally annotations to express metadata about the resources. Several requirements for KGs [13] are that they express meaning as structure and use a limited set of relations. All statements and entities in a knowledge graph should be unambiguous, which can be accomplished by “using global identifiers with unambiguous denotation." Furthermore, knowledge graphs must provide justification for statements by including explicit provenance information.

The Findability, Accessibility, Interoperability, and Reusability (FAIR) [17] principles provide guidelines for publishing data and metadata on the web. In order for a knowledge graph to adhere to the FAIR principles, resources in the graph should use unique and persistent identifiers. The knowledge graph should be accessible, freely and openly, even after the data has been retired. The resources should be described using descriptive metadata that is written using vocabularies that adhere to the FAIR principles. Furthermore, provenance about the resource, such as how they was derived, should be rich and detailed.

Interoperability is the ability of data or tools from different sources to be integrated [17]. It is one important aspect of knowledge graphs, as it advocates understanding and reuse amongst various users. In order to achieve this across the multitude of engineers, developers, and researchers, it is important to define a coherent set of best practices for the engineering and creation of knowledge graphs. However, today there exist many best practices and methodologies for creating different kinds of knowledge graphs (e.g. ontologies, linked data, etc.), resulting in knowledge graphs of various quality. Worse yet, when drawing the best practices from these different development communities, we see that they are sometimes incompatible. For example, there is conflict over the best practices between those communities that adhere to strong or weak weight semantics for their knowledge graphs. In addition, we must consider how these best practices may apply to knowledge graphs in general outside of semantic web communities, such as those used by the Natural Language

Processing (NLP) or Machine Learning (ML) communities. Often, at the application or project level, methodologies used for the creation of knowledge graphs may not necessarily follow any particular set of best practices. For example, several methods employed when constructing knowledge graphs may include using models inherent in specific ontologies that suggest a particular knowledge representation pattern. Additionally, for annotation using concepts from ontologies, such as the annotation of dataset variables, the use of public search engines on vocabulary repositories (Bioportal [14], Ontobee [18], linked open vocabularies [16]) are often employed. The result may be the use of a concept that best matches the task at hand. The use of multiple ontologies may have undesirable consequences, such as resulting in inconsistencies between terms. If such concepts do not exist, one may continue to define their own terms and build their own ontology without using commonly agreed upon definitions. As a validation approach, some form of consistency checking is required in order to keep the knowledge graph suitable for inference activities.

Knowledge graph developers must choose between methodologies and best practices for specific domains, or for engineering different kinds of knowledge graphs. Therefore, one challenge is to examine and consolidate existing best practices, and possibly extend them, to encompass all kinds of knowledge graphs, as well as currently unaddressed aspects of knowledge graphs. Thus, we may ultimately provide a clear workflow for an arbitrary development team to create FAIR [17] (Findable, Accessible, Interoperable, and Re-usable) knowledge graphs.

## 5.2 Existing Best Practices

We distinguish between two different categories of best practices for the creation of knowledge graphs: its provision (driving “findable” and “accessible”) and its design (driving “interoperable” and “re-usable”). In addition, we must promote these standards, because, even within the Semantic Web community, an exhaustive set of best practices or standards is nonexistent. It is hard to know which best practices to develop if we do not know which others may exist. It then seems unreasonable to expect a similar set of best practices from the NLP and ML communities with whom we try to bridge the gap. While an exhaustive set is difficult to come by, we have identified a non-exhaustive list of existing best practices related to the design of knowledge graphs, inclusion of high quality metadata and provenance, and methods for converting structured data into knowledge graphs.

### 5.2.1 Knowledge Graph Design

When starting to design an ontology or knowledge graph, all the necessary concepts or possible uses may not be initially known. Agile [9] & eXtreme [15, 1] Design (XD) methodologies allows for modular updates when needed, which is pertinent for the design and sharing of ontologies among collaborative groups. This approach is often referred to as modular ontology modelling or design [8].

Agile design encourage simplicity, in which only essential features are implemented at first, and additional features can be included in the future. When following this methodology, one should explain complex ideas fully and keep straightforward ideas simple. The eXtreme Design methodology was inspired by this Agile approach, as ontologies used should only contain concepts and properties that are essential for the particular task at hand. XD requires end-user or customer involvement, is driven based on a set of design requirements, and is iterative in that it produces an early deliverable and subsequently builds on the end result.

Ontology Design Patterns (ODPs) [5] can provide guidance into specific ways to represent different forms of knowledge[1]. Use of ODPs promote interoperability across disciplines. Ontology Design Patterns include a set of ontology pattern types, a list of existing patterns, and a table of domains and their descriptions for which the patterns can be applied. Ontology pattern types cover structural ODPs, such as logical and architectural ODPs, correspondence ODPs, such as re-engineering and alignment ODPs, and presentation ODPs, such as naming and annotation ODPs. Additionally, content, reasoning, and lexico-syntactic ODPs are included. The list of patterns contain community submitted patterns for content, re-engineering, alignment, logical, architectural and lexico-syntactic ODPs. The specified domains are available on the organization wiki[2] and include various sciences, linguistics, music and media, and various industry related topics, such as management, industrial processes, and manufacturing. Creating ODPs allows for re-usability of artifacts across different use cases. Re-usability and interoperability is further promoted by following best practices for documenting Ontology Design Patterns [10].

### 5.2.2 High Quality Metadata

High quality metadata is crucial to help users find relevant knowledge graphs. Two useful specification are the HCLS dataset specification [6] and the Data on the Web Best Practices [12]. The HCLS dataset specification provides detailed requirements on how to describe datasets in terms of Semantic Web vocabularies in order to promote the search and reuse of datasets. Many of the properties recommended in this specification can be applied to the publication of knowledge graphs, such as the inclusion of publication and version information, descriptions and keywords, and provenance.

The Data on the Web Best Practices W3C recommendation also provides guidelines related to the publication of data that can also be extended to publishing knowledge graphs. This document includes the specifications of providing detailed metadata, data quality information, provenance, persistent identifiers, and documentation. Furthermore, the reuse of existing vocabularies and making the data accessible through an application program interface (API) are also recommended.

As mentioned above, an important aspect of high quality metadata is the inclusion of detailed provenance, where the content comes from and how it was generated or derived. PROV-O [11] and Nanopublications [7] offer guidance in this respect. The PROV-O ontology is a W3C recommendation that provides a set of OWL classes, properties, and restrictions that can be used to include provenance annotations. PROV-O includes high level classes for prov:Entity, prov:Agent, and prov:Activity. Entities are defined as anything physical, digital, conceptual, real or imaginary that has fixed features. Example prov:Entity classes include prov:Collection, prov:Plan and prov:Bundle. Agents are defined as the bearers of responsibility for an activity. Included in the set of prov:Agent classes are prov:Organization, prov:Person and prov:SoftwareAgent. Finally, prov:Activities represent events that occur over a period of time that involve entites.

Nanopublications allow for context to reinforce the value of an assertion, which can be included in the form of provenance statements about assertions or facts. The nanopublication

[1] http://ontologydesignpatterns.org/wiki/

[2] http://ontologydesignpatterns.org/wiki/Community:Domain

model can be implemented using a collection of RDF Named Graphs. Facts are included in an assertion graph. Provenance information about the assertion is included in a provenance graph. Provenance about the nanopublication itself is included in a publication information graph.

Linked Data offers another way of providing semantically rich knowledge graphs. The Best Practices for Publishing Linked Data [2] specifies a set of guidelines as a sequence of steps. These steps include selecting and then modeling a dataset, choosing appropriate URIs, referencing standard vocabularies when possible, converting the data to a Linked Data representation, and providing machine access such that the data is reachable by search engines and similar web processes.

### 5.2.3 Structured Data Transformation

Standardized methods to transform data into knowledge graphs make it easier to maintain and reproduce. Two such methods for data tranformation that we recommend include using R2RML [3] or RML [4]. R2RML is language that allows the user to define custom mappings from a relational database schema to an RDF model. The R2RML document itself is written in RDF, in which mappings for each table can include a template for the desired output URI structure, specified ontology classes to be instantiated to, and relationships between columns. Such relationships can be used to link to other relational databases through, for example, primary keys. Since the input data are stored in relational databases, SQL queries can be used in the R2RML mapping files when constructing the desired RDF. An extension to R2RML is RML, which aims to be more generic by keeping the core model of the R2RML vocabulary, but excluding database specific concepts.

## 5.3 Challenges

We have identified four challenges that must be overcome to promote the use of best practices when constructing knowledge graphs. It is important to promote the best practices identified in this chapter in order to encourage wide spread use. We must also find additional best practices that were missed in this initial search. Overcoming the challenges of consolidating and integrating best practices from different communities will allow for interdisciplinary collaboration. Finally, the set of best practices that are required for different methodologies of knowledge graph creation should be specified. For example, the best practices used for manual creation of knowledge graphs may differ from automated approaches. Corresponding sets have to be discovered and organized accordingly.

## 5.4 Conclusion

In this chapter we considered how to apply best practices to knowledge graphs by identifying a non-exhaustive list of existing best practices. We discussed best practices pertaining to knowledge graph design, including Agile and eXtreme Design methodologies, as well as Ontology Design Patterns. We considered W3C specifications pertaining to including high quality metadata when publishing knowledge graphs, including the HCLS dataset specification, the Data on the Web best practices, and the Best Practices for Publishing Linked Data. Furthermore, we considered existing mapping languages for transforming structured data into knowledge graphs, including R2RML and RML. Finally, we discussed some challenges that need to be overcome. A coherent set of best practices for the engineering and creation of knowledge graphs advocates understanding and reuse amongst engineers, developers, and researchers working with knowledge graphs.

## References

1 E. Blomqvist, K. Hammar, and V. Presutti. Engineering ontologies with patterns-the extreme design methodology., 2016.

2 B. Hyland, G. Atemezing, and B. Villazón-Terrazas. Best practices for publishing linked data. *W3C recommendation*, 2014.

3 S. Das, S. Sundara, and R. Cyganiak. R2RML : RDB to RDF mapping language. *W3C Recommendation http://www.w3.org/TR/r2rml/*, 2011.

4 A. Dimou, M. Vander Sande, P. Colpaert, R. Verborgh, E. Mannens, and R. Van de Walle. RML: A generic language for integrated rdf mappings of heterogeneous data. In *LDOW*, 2014.

5 A. Gangemi. Ontology design patterns for semantic web content. In *International semantic web conference*, pages 262–276. Springer, 2005.

6 A. J. Gray, J. Baran, M. S. Marshall, and M. Dumontier. Dataset descriptions: HCLS community profile. *Interest group note, W3C (May 2015) http://www. w3. org/TR/hcls-dataset*, 2015.

7 P. Groth, A. Gibson, and J. Velterop. The anatomy of a nanopublication. *Information Services & Use*, 30(1-2):51–56, 2010.

8 K. Hammar, P. Hitzler, and A. Krisnadhi. *Advances in Ontology Design and Patterns*, volume 32. IOS Press, 2017.

9 A. Hunt and D. Thomas. The trip-packing dilemma [agile software development]. *IEEE Software*, 20(3):106–107, 2003.

10 N. Karima, K. Hammar, and P. Hitzler. How to document ontology design patterns. *Advances in Ontology Design and Patterns, Studies on the Semantic Web*, 32:15–28, 2017.

11 T. Lebo, S. Sahoo, and D. McGuinness. PROV-O: The PROV ontology. *W3C recommendation*, 2013.

12 B. F. Lóscio, C. Burle, and N. Calegaro. Data on the web best practices. *W3C recommendation*, 2017.

13 J. McCusker, J. Erickson, K. Chastain, S. Rashid, R. Weerawarana, and D. McGuinness. What is a Knowledge Graph? *Semantic Web Journal*, Under Review, 2018.

14 N. F. Noy, N. H. Shah, P. L. Whetzel, B. Dai, M. Dorf, N. Griffith, C. Jonquet, D. L. Rubin, M.-A. Storey, C. G. Chute, and M. A. Musen. BioPortal: ontologies and integrated data resources at the click of a mouse. *Nucleic acids research*, 37(suppl_2):W170–W173, 2009.

15 V. Presutti, E. Daga, A. Gangemi, and E. Blomqvist. eXtreme design with content ontology design patterns. In *Proc. Workshop on Ontology Patterns*, 2009.

16 P.-Y. Vandenbussche, G. A. Atemezing, M. Poveda-Villalón, and B. Vatant. Linked open vocabularies (LOV): a gateway to reusable semantic vocabularies on the web. *Semantic Web*, 8(3):437–452, 2017.

17 M. D. Wilkinson, M. Dumontier, I. J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomberg, J.-W. Boiten, L. B. da Silva Santos, P. E. Bourne, et al. The FAIR guiding principles for scientific data management and stewardship. *Scientific data*, 3, 2016.

18 Z. Xiang, C. Mungall, A. Ruttenberg, and Y. He. Ontobee: A linked data server and browser for ontology terms. In *ICBO*, 2011.

# 6 Knowledge Integration at Scale

*Andreas Harth (Fraunhofer IIS - Nürnberg, DE), Roberto Navigli (Sapienza University of Rome, IT), Andrea Giovanni Nuzzolese (CNR - Rome, IT), Maria-Esther Vidal (TIB - Hannover, DE)*

## 6.1 Introduction

The number and variety of data sets have grown exponentially during the last decades and a similar growth rate is expected in the next years. In order to transform the enormous amount of disparate data into actionable knowledge, fundamental problems, such as knowledge integration, must be solved. Integrating data sets requires the effective identification of entities that, albeit described differently, correspond to the same real-world entity. This integration problem has received considerable attention from various computer science domains such as databases, artificial intelligence, and semantic web. However, there are still key challenges that need to be faced in order to integrate knowledge at scale. Open issues arise because entities can be made available by autonomous sources either at rest or in motion, and represented in various models or (un)structured formats. Moreover, entity meaning may change over time and become inconsistent and incomplete with periodic peaks. During the Dagstuhl seminar “Knowledge Graphs: New Directions for Knowledge Representation on the Semantic Web", members of the “Knowledge Integration at scale” working group have discussed these challenges around entity meaning and identity reasoning. Results of these discussions are reported, as well as existing approaches for knowledge integration, grand challenges, and future research directions.

## 6.2 Knowledge Integration and Existing Approaches

The problem of knowledge integration can be framed as follows. Given a collection of data sets such as unstructured text, media (e.g., images, videos, sounds), knowledge graphs, databases, and knowledge bases, the problem of knowledge integration is to identify if two entities in the collection of data sets match or do not match the same real-world entity. An entity–in a data collection–is a multi-modal item of knowledge like a word, a concept, a sentence, a text, a database record, a media segment, a knowledge graph, or an ontology. Solving the problem of knowledge integration requires first identifying the knowledge items in diverse data sets. Then, interoperability conflicts among these items need to be detected, and finally, these conflicts need to be resolved. Once equivalent entities have been matched, different fusion policies are performed for merging them into a single entity [6]. Considering the wide nature of entities, the state of the art has focused on integration methods that reduce manual work and maximize accuracy and precision.

Data integration has been extensively treated in the context of databases [9]. As a result, a vast amount of integration frameworks [13] have been developed; they implement data integration systems following the local-as-view (LAV), global-as-view paradigms (GAV), and global and local as view (GLAV) [20]. Further, query processing has also played a relevant role in solving data integration on the fly. Graph-based traversal [2, 21], and distributed and federated query processing [3, 5, 27, 31] are representative approaches for enabling the fusion of the properties of equivalent entities on demand, i.e., at query execution time.

To overcome interoperability conflicts generated by the wide variety of existing formats–short notes, videos, images, maps, or publications–several unstructured processing techniques have been proposed. Natural language processing techniques contribute to integrating structured and textual data by providing linguistic annotation methods at different levels [25], e.g., part-of-speech tagging, syntactic parsing, named entity recognition, word sense disambiguation, entity linking, and anaphora resolution. Further, visual analytics techniques facilitate the extraction and annotation of entities from non-textual data sources [1, 15]. Annotations extracted from unstructured data represent the basis for determining relatedness among the annotated entities by the mean of similarity measures, as well as for identifying matches between highly similar entities.

Several approaches have been proposed to integrate structured data. KARMA [18] is a semi-automatic framework that relies on ontologies and mapping rules for transforming data sources such as relational tables or JSON files, into RDF knowledge graphs. LDIF [32], LIMES [26], MINTE [6], Sieve [22], Silk [36], and RapidMiner LOD Extension [29] also tackle the problem of data integration. However, they resort to similarity measures and link discovery methods to match equivalent entities from different RDF graphs. Likewise, Galkin et al. [12] present SJoin, a join operator, for identifying entity matches in heterogeneous RDF knowledge during query processing. With the aim of transforming structured data in tabular or nested formats like CSV, relational, JSON, and XML, into RDF knowledge graphs, diverse mapping languages have been proposed [7, 16, 19, 34]. Exemplary mapping languages and frameworks include RDF Mapping Language (RML) [8], R2RDF [33], and R2RML [28]. Additionally, a vast amount of research has been conducted to propose effective and efficient approaches for ontology alignment [4, 11, 23]. Regardless of the effort of automatizing entity and ontology matching, still a significant amount of manual work is required in all these approaches. This lack of automation prevents applications from scaling up to large and heterogeneous data sets.

The AI community has also actively contributed to the problem of data integration [14]. Specifically, recent machine learning methods provide effective and accurate building blocks for entity matching, entity linking, name resolution, deduplication, and identity resolution. For instance, random forest models have shown significant improvement of entity matching [10, 30]. Further, deep learning and embedding representations are promising methods for matching knowledge items represented in diverse formats [24, 35]. Moreover, logic-based approaches like probabilistic soft logic, have evidenced accurate performance in matching of entities from multiple types [17]. Notwithstanding the overall cost reduction and improved precision observed by the state-of-the-art machine learning approaches, the outcome of these approaches directly depends on the quality of the training data. Given the large variety and volume of existing data sets, the generation of these training data sets represents a fundamental open challenge.

## 6.3 Grand Challenges of Knowledge Integration

The tremendous amount of research contributions for integrating knowledge items accurately corroborates the importance of the problem. Nevertheless, data complexity challenges imposed by current available data sets and modern knowledge-driven applications demand novel computational methods for solving knowledge integration at scale. Particularly, integration of multi-modal entities represented at different levels of abstraction and evolving over time, remains unsolved. Finally, context-based knowledge integration, including cultural specificity of a concept, temporal context (within a given culture), and domain context also demand effective and efficient solutions from the community.

## 6.4 Opportunities of Knowledge Integration in Knowledge Graphs

Knowledge graphs encompass large volume of knowledge items, and enable the description of the meaning of their main properties and relations. Albeit challenging in terms of data and knowledge complexity, knowledge graphs bring enormous opportunities for improving modern methods of knowledge integration. First, machine learning approaches like knowledge graph embeddings, transfer learning, bidirectional information extraction, active learning, and distance supervision, can benefit from the knowledge encoded in knowledge graphs, thus providing more accurate results during knowledge integration. Further, the definition of expressive formalisms for describing integrated knowledge such as probabilistic logic, and symbolic or subsymbolic knowledge representation, represent open challenges. Similarly, the definition of entity matching methods capable of exploiting these novel representations correspond to a propitious research topic. Finally, there is an expeditious need of devising methods for efficiently including *humans in the loop*, and enabling them to effectively define and curate high-quality training data sets.

## 6.5 Conclusions and Future Directions

The problem of knowledge integration in a vast variety of large data sets has been discussed. Existing approaches in areas like databases and semantic web, as well as the application of modern machine learning methods, not only evidence the relevance of the problem, but also the diversity of challenges that demand to be faced. The future of the area promises a wide range of opportunities that vary from representation formalisms, modern machine learning methods, and hybrid knowledge integration techniques. Our ambition is that the presented discussion encourages the community to develop novel methods that enable the overall reduction of knowledge integration while providing highly accurate results.

### References

1 R. Agerri, X. Artola, Z. Beloki, G. Rigau, and A. Soroa. Big Data for Natural Language Processing: A streaming approach. *Knowl.-Based Syst.*, 79:36–42, 2015.

2 R. Angles, M. Arenas, P. Barceló, A. Hogan, J. L. Reutter, and D. Vrgoc. Foundations of Modern Query Languages for Graph Databases. *ACM Comput. Surv.*, 50(5):68:1–68:40, 2017.

3 C. B. Aranda. *Federated Query Processing for the Semantic Web.* PhD thesis, Technical University of Madrid, Spain, 2014.

4 M. Cheatham, I. F. Cruz, J. Euzenat, and C. Pesquita. Special issue on ontology and linked data matching. *Semantic Web*, 8(2):183–184, 2017.

5 C. Chen, B. Golshan, A. Y. Halevy, W. Tan, and A. Doan. BigGorilla: An Open-Source Ecosystem for Data Preparation and Integration. *IEEE Data Eng. Bull.*, 41(2):10–22, 2018.

6 D. Collarana, M. Galkin, I. Traverso-Ribón, M. Vidal, C. Lange, and S. Auer. MINTE: semantically integrating RDF graphs. In *Proceedings of the 7th International Conference on Web Intelligence, Mining and Semantics, WIMS*, 2017.

7 D. V. Deursen, C. Poppe, G. Martens, E. Mannens, and R. V. d. Walle. XML to RDF Conversion: A Generic Approach. In *Proceedings of the 2008 International Conference on*

*Automated Solutions for Cross Media Content and Multi-channel Distribution*, AXMEDIS '08, pages 138–144, 2008.

8 A. Dimou, M. V. Sande, P. Colpaert, R. Verborgh, E. Mannens, and R. V. de Walle. RML: A generic language for integrated RDF mappings of heterogeneous data. In *Proceedings of the Workshop on Linked Data on the Web co-located with the 23rd International World Wide Web Conference (WWW 2014)*, 2014.

9 A. Doan, A. Y. Halevy, and Z. G. Ives. *Principles of Data Integration.* Morgan Kaufmann, 2012.

10 X. L. Dong and T. Rekatsinas. Data Integration and Machine Learning: A Natural Synergy. In *Proceedings of the 2018 International Conference on Management of Data, SIGMOD Conference 2018, Houston, TX, USA, June 10-15, 2018*, pages 1645–1650, 2018.

11 J. Euzenat and P. Shvaiko. *Ontology Matching, Second Edition.* Springer, 2013.

12 M. Galkin, D. Collarana, I. T. Ribón, M. Vidal, and S. Auer. Sjoin: A semantic join operator to integrate heterogeneous RDF graphs. In *Database and Expert Systems Applications - 28th International Conference, DEXA 2017, Lyon, France, August 28-31, 2017, Proceedings, Part I*, pages 206–221, 2017.

13 B. Golshan, A. Y. Halevy, G. A. Mihaila, and W. Tan. Data Integration: After the Teenage Years. In *Proceedings of the 36th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, PODS 2017, Chicago, IL, USA, May 14-19, 2017*, pages 101–106, 2017.

14 A. Y. Halevy, A. Rajaraman, and J. J. Ordille. Data Integration: The Teenage Years. In *Proceedings of the 32nd International Conference on Very Large Data Bases, Seoul, Korea, September 12-15, 2006*, pages 9–16, 2006.

15 C. A. Henning and R. Ewerth. Estimating the information gap between textual and visual representations. *IJMIR*, 7(1):43–56, 2018.

16 P. Heyvaert, A. Dimou, B. D. Meester, T. Seymoens, A. Herregodts, R. Verborgh, D. Schuurman, and E. Mannens. Specification and implementation of mapping rule visualization and editing: Mapvowl and the rmleditor. *J. Web Sem.*, 49:31–50, 2018.

17 A. Kimmig, A. Memory, R. J. Miller, and L. Getoor. A Collective, Probabilistic Approach to Schema Mapping. In *33rd IEEE International Conference on Data Engineering, ICDE 2017, San Diego, CA, USA, April 19-22, 2017*, pages 921–932, 2017.

18 C. A. Knoblock and P. A. Szekely. Exploiting Semantics for Big Data Integration. *AI Magazine*, 36(1):25–38, 2015.

19 M. Lefrançois, A. Zimmermann, and N. Bakerally. A SPARQL extension for generating RDF from heterogeneous formats. In *The Semantic Web - 14th International Conference, ESWC 2017, Portorož, Slovenia, May 28 - June 1, 2017, Proceedings, Part I*, pages 35–50, 2017.

20 M. Lenzerini. Data Integration: A theoretical perspective. In *Proceedings of the Twenty-first ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, June 3-5, Madison, Wisconsin, USA*, pages 233–246, 2002.

21 L. Libkin, J. L. Reutter, A. Soto, and D. Vrgoc. TriAL: A navigational algebra for RDF triplestores. *ACM Trans. Database Syst.*, 43(1):5:1–5:46, 2018.

22 P. N. Mendes, H. Mühleisen, and C. Bizer. Sieve: linked data quality assessment and fusion. In *Proceedings of the 2012 Joint EDBT/ICDT Workshops, Berlin, Germany, March 30, 2012*, pages 116–123, 2012.

23 M. Mohammadi, A. A. Atashin, W. Hofman, and Y. Tan. Comparison of Ontology Alignment Systems Across Single Matching Task Via the McNemar's Test. *TKDD*, 12(4):51:1–51:18, 2018.

24 S. Mudgal, H. Li, T. Rekatsinas, A. Doan, Y. Park, G. Krishnan, R. Deep, E. Arcaute, and V. Raghavendra. Deep Learning for Entity Matching: A Design Space Exploration.

In *Proceedings of the 2018 International Conference on Management of Data, SIGMOD Conference 2018, Houston, TX, USA, June 10-15, 2018*, pages 19–34, 2018.

25 R. Navigli. Natural Language Understanding: Instructions for (Present and Future) Use. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden.*, pages 5697–5702, 2018.

26 A. N. Ngomo and S. Auer. LIMES - A time-efficient approach for large-scale link discovery on the web of data. In *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, pages 2312–2317, 2011.

27 M. T. Özsu and P. Valduriez. Distributed and Parallel Database Systems. In *Computing Handbook, Third Edition: Information Systems and Information Technology*, pages 13: 1–24. 2014.

28 F. Priyatna, Ó. Corcho, and J. F. Sequeda. Formalisation and experiences of R2RML-based SPARQL to SQL query translation using morph. In *23rd International World Wide Web Conference, WWW '14, Seoul, Republic of Korea, April 7-11, 2014*, pages 479–490, 2014.

29 P. Ristoski, C. Bizer, and H. Paulheim. Mining the web of linked data with RapidMiner. *J. Web Sem.*, 35:142–151, 2015.

30 S. Rong, X. Niu, E. W. Xiang, H. Wang, Q. Yang, and Y. Yu. A Machine Learning Approach for Instance Matching Based on Similarity Metrics. In *The Semantic Web - ISWC 2012 - 11th International Semantic Web Conference, Boston, MA, USA, November 11-15, 2012, Proceedings, Part I*, pages 460–475, 2012.

31 S. Sakr, M. Wylot, R. Mutharaju, D. L. Phuoc, and I. Fundulaki. *Linked Data - Storing, Querying, and Reasoning.* Springer, 2018.

32 A. Schultz, A. Matteini, R. Isele, C. Bizer, and C. Becker. LDIF - Linked Data Integration Framework. In *Proceedings of the Second International Workshop on Consuming Linked Data (COLD2011)*, 2011.

33 J. F. Sequeda and D. P. Miranker. Mapping Relational Databases to Linked Data. In *Linked Data Management.*, pages 95–115. 2014.

34 D. Spanos, P. Stavrou, and N. Mitrou. Bringing relational databases into the semantic web: A survey. *Semantic Web*, 3(2):169–209, 2012.

35 Z. Sun, W. Hu, Q. Zhang, and Y. Qu. Bootstrapping Entity Alignment with Knowledge Graph Embedding. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden.*, pages 4396–4402, 2018.

36 J. Volz, C. Bizer, M. Gaedke, and G. Kobilarov. Silk - A link discovery framework for the web of data. In *Proceedings of the WWW2009 Workshop on Linked Data on the Web, LDOW*, 2009.

# 7 Knowledge Dynamics and Evolution – “No Dynamic or Evolving Knowledge Graph Without Provenance”

*Eva Blomqvist (Linköping University, SE), Cogan Matthew Shimizu (Wright State University - Dayton, US), Barend Mons (Leiden University Medical Center, NL), and Heiko Paulheim (Universität Mannheim, DE)*

Knowledge lives. It is not static, nor does it stand alone. It may change or grow–evolve; its provenance may become more or less certain. Belief in it may wax or wane over time. Studying how knowledge graphs may capture the evolutionary nature of knowledge is a critical need that the community must address. In this chapter, we outline some motivating use-cases and the accompanying challenges, as well as starting points in existing literature.

## 7.1 Introduction

Knowledge is not static, but constantly evolving. This is true, regardless of whether we are representing personal knowledge, knowledge within a company, or open, common knowledge on the web. Not only does the knowledge itself change, but also our perception of and beliefs about it, such as its trustworthiness or accuracy. Therefore, if knowledge graphs are to capture at least a significant portion of the world's knowledge, they also need to be able to evolve and capture the changes made to the knowledge it contains. There already exist some approaches, in various related fields, for both describing those changes, as well as for dealing with volatile knowledge. However, quite a few open questions still exist.

Perhaps foremost and fundamental of those is the question, ”What exactly does it mean for a knowledge graph to *evolve*?” We do not, at this time, have a clear definition or description of what knowledge graph evolution means. Then, we must address the following.

1. “Are existing methods for reflecting the evolutionary nature of knowledge sufficient for capturing such knowledge in a knowledge graph?”
2. “What problems are not solved by existing methods?”
3. “What tasks are to be performed manually versus completed by a system?”

Yet, we do know that evolution is a very important aspect to the future of knowledge graphs; this is recognised by almost all large knowledge graph developers and providers, today.

Examples of this were given during the Enterprise-Scale Knowledge Graphs Panel at the 17th International Semantic Web Conference.[3] For instance, Yuqing Gao from Microsoft pointed out their challenge of having a real-time knowledge graph, but with archiving, which is still a research challenge at Microsoft. Jamie Taylor of Google also acknowledged the long term evolution of the Google Knowledge Graph as one of their main challenges. The IBM Watson group is also struggling with similar challenges, although they claim to take a more dynamic approach, not focusing on one global knowledge graph, but a framework for building domain specific knowledge graphs, including knowledge discovery and analysis of change effects. Thus, their main struggles include modelling and analysing changing information and incrementally updating global knowledge on horizontally scaled storage solutions.

[3] http://iswc2018.semanticweb.org/panel-enterprise-scale-knowledge-graphs/

The rest of the chapter is organized as follows. Section 7.2 briefly explores related work that we may use as as starting points for further studying evolving knowledge graphs. Section 7.3 describes some use-cases for evolving knowledge graphs, as identified during this Dagstuhl seminar. Section 7.4 presents the initial challenges facing developers for evolving knowledge graphs. Finally, in Section 7.5, we summarize and conclude.

## 7.2 Starting Points

While we may not have a clear definition for evolution in knowledge graphs, we may still draw inspiration from previous work in related fields. As, in the realm of knowledge graphs, there is no clear distinction between data and information on one side and knowledge[4] on the other (ontologically, or in terms of description logics, we would say, ABox and TBox, respectively), we may draw from a wide variety of fields. For example, this means that both approaches for managing changing (web) data [8, 4] as well as schema and ontology evolution[7, 2, 10, 5, 6] may be relevant for knowledge graph dynamics and evolution.

It is also important to examine the rate of change in the data, as approaches differ across the spectrum. When data changes infrequently, state of the art approaches most often include the production of new, manually curated versions of the data at certain time intervals, and some appropriate version tracking and archiving of the dataset[8, 4], possibly combined with query rewriting and other techniques. At the other end of the spectrum, data may be treated as a stream, where approaches for data stream management and stream reasoning[9, 1, 3], including RDF stream processing, have been applied.

The versioning of datasets and ontologies is a quite well-studied problem. For example, there exist annotation schemas and ontologies for describing datasets and ontologies and track versions, such as through extensions of the PROV-O.[5] In addition, there exist mapping languages for mapping between versions. Query rewriting has been used to transform queries over one version to queries over a new version of the data.

Inherently, due to the open world assumption, languages proposed for the web are also quite well suited for managing incomplete information. However, there are less approaches proposed for how to actually manage the change process, detect change needs, apply changes, and so on. We do know quite well, at a technical level, how to manually update ontologies (e.g. implement changes, check consistency, track provenance of changes). Unfortunately, how to automate parts of this process, or how to trigger and guide the change process is largely unstudied. This is particularly true for schemas/ontologies, while for change management in data we can rely on the history of relational database research and thereby also more approaches have been proposed for graph data.

Additionally, there are approaches proposed for managing inconsistent and fuzzy knowledge, for example when using ontologies. These are maybe not so well used in practice, but are usually well founded theoretically, and may have an important role to play when dealing with large scale real-world knowledge that is rarely precise and consistent.

However, particularly targeted at knowledge graphs there are still not many approaches available, hence, we are again left with either applying approaches originally developed with some related structure in mind, e.g. ontologies and linked data, or we may look at the actual practices for managing large knowledge graphs today and learn from there. While the former

[4] as the term is classically used in knowledge representation
[5] https://www.w3.org/TR/prov-o/

was already discussed above, instead considering the latter an example of a change tracking model, particularly targeted at knowledge graphs, is the Wikidata model for storing edit history.

## 7.3 Use Cases

As an additional starting point, during the Dagstuhl seminar we collected a small (incomplete) set of motivating use cases that may provide enough challenges in order to actually start specifying the possible tasks involved and create a more detailed map of what solutions exist and where the white spots are.

- New **laws** are usually written as modifications of previous laws. This creates a complex network of changes to laws, which together makes up the law of a country. If this is to be modelled in a knowledge graph, the evolution/change history conveys important information about the actual content and meaning of the law.
- **Patient records** contain information about the states of a patient. Something that is believed at one point might be proven false in the next time instant. This needs to be captured, tracked and reasoned with, when analysing patient data.
- A related use case is that of **patient monitoring**, where IoT devices and sensors are used to monitor patients, either at home or in a care facility. Streams of data come from each sensor and need to be interpreted both with respect to the patient record and history, as well as in relation to generic medical knowledge. This in order to create situation awareness, and reason on potential future situation that are likely to occur, e.g. to prevent dangerous situations and alert medical staff. Here several KGs can be envisioned, i.e. both the personal, patient specific, knowledge, but also generic medical knowledge graphs, and all of them have to evolve, although potentially at a very different pace.
- Knowledge discovery, for instance in **drug discovery**, implies to treat concepts individually in a local context, to allow for different viewpoints. In this way changes can happen locally, without affecting the whole knowledge graph. Knowledge graphs can later be composed of these components, filtered for certain views. Creating what could be called a fluent KG, where new knowledge can emerge at every new KG composition.
- In many organisations, such as a **police department**, individuals (such as police analysts) want to have their own concepts represented in the knowledge graph. This may imply to have individual knowledge graphs, or individual views of a knowledge graph, but such additions or changes may also need to be introduced into the overall shared knowledge graph if they reflect evolution or emergenc of new concepts, rather than just individual views. Such changes need to be tracked, and one needs to determine what view (or version) to use in a specific case, what knowledge from individual views (or contexts) to propagate and what should stay private.
- **Crisis detection** of large scale events (i.e. natural disaster, battle spaces, crime) is another use case. Inputs will be frequent and likely conflicting. Representative, underlying knowledge graphs will need to handle this in order to reason on what has actually occurred and what is currently happening. We will need ways to also visualize and render such information as well as tracking provenance and uncertainty and enabling evolution of data in the graph.

## 7.4 Major Challenges

Based on what we know about the state of the art in knowledge and data evolution and dynamics, and these use cases we have identified a (probably still incomplete) list of challenges in this area, which are listed below.

- Define the exact notion of evolution, i.e. distinguish it from notions such as change, dynamicity, versioning, etc.
  - Different levels of tracking evolution will be needed for different use cases; best practices and guidelines will need developed.
- Manage the volume of provenance (i.e. preventing provenance explosion) caused by capturing all evolution information, all data versions, etc.
  - This could include patterns for providing provenance information outside the actual knowledge graphs.
- Presenting provenance (and information about evolution) to an end user, developer, knowledge engineer.
  - Lenses could be used (e.g., show just the current state versus showing the evolution path to this statement). This allows for different viewpoints on a subject.
- Managing the full scale of evolution rates, i.e. from slowly changing concepts to rapidly changing streams of data – potentially all in one system.
- Engineering mechanisms for evolution (e.g., detecting and submitting updates)
- Social processes need to be taken into account – someone usually owns a KG and may not want it to be changed.
  - Mechanism for curation, change suggestions, moderation etc. are needed.
    - Managing the computational challenge of assessing and handling the effects of the updates.
  - Developing those mechanisms in a “tolerant” way (i.e., accepting local changes but avoiding global drift), predicting effects of a change on local and global levels.
  - Methods for automating evolution, e.g. detecting signals of change, generating suggestions, finding the most appropriate change action.
  - Tooling and methodology support
- Embracing controversy – we must be able to represent different viewpoints, contexts, inconsistencies, and even fuzzy or unclear notions that may or may not later evolve into more crisp ones. For example, consider a knowledge graph that encompasses religions.
- Overcoming current storage and computation limitations for implementing evolution and dynamics of KGs in real-world systems, i.e. it all needs to scale.

## 7.5 Conclusion

Capturing the evolutionary nature of knowledge is critical as the community moves forward and continues to build large, encompassing knowledge graphs, especially those that aim to capture knowledge as it is created, discovered, or genreated. Of course, there are many challenges inherent to this, from provenance explosion to what it actually means for an knowledge graph to evolve. In this chapter, we have described several motivating use-cases that capture useful knowledge, but in order to be effective, must address the notion of evolving knowledge. In addition, we have described the challenges that these developers must face, but have also included a number of well-studied starting points from similar fields.

## Acknowledgments

The authors would like to thank Sonja Zillner, Michel Dumontier, Sarven Capadisli, Maria-Esther Vidal, Elena Simperl, and Juan Sequeda for their valuable discussion and input.

## References

1 D. Barbieri, D. Braga, S. Ceri, E. Della Valle, and M. Grossniklaus. Stream reasoning: Where we got so far. In *NeFoRS 2010: 4th International Workshop on New Forms of Reasoning for the Semantic Web: Scalable and Dynamic*, 2010.

2 S. Bloehdorn, P. Haase, Y. Sure, and J. Voelker. *Ontology Evolution*, chapter 4, pages 51–70. Wiley-Blackwell, 2006.

3 D. Dell'Aglio, E. Della Valle, F. van Harmelen, and A. Bernstein. Stream reasoning: A survey and outlook. *Data Science*, (Preprint):1–25.

4 J. D. Fernández, J. Umbrich, A. Polleres, and M. Knuth. Evaluating query and storage strategies for RDF archives. In *SEMANTICS*, pages 41–48. ACM, 2016.

5 M. Hartung, J. F. Terwilliger, and E. Rahm. Recent advances in schema and ontology evolution. In *Schema Matching and Mapping*, Data-Centric Systems and Applications, pages 149–190. Springer, 2011.

6 P. D. Leenheer and T. Mens. Ontology evolution. In *Ontology Management*, volume 7 of *Semantic Web and Beyond: Computing for Human Experience*, pages 131–176. Springer, 2008.

7 N. F. Noy and M. C. A. Klein. Ontology evolution: Not the same as schema evolution. *Knowl. Inf. Syst.*, 6(4):428–440, 2004.

8 V. Papakonstantinou, G. Flouris, I. Fundulaki, K. Stefanidis, and G. Roussakis. Versioning for linked data: Archiving systems and benchmarks. In *BLINK@ISWC*, volume 1700 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2016.

9 E. D. Valle, S. Ceri, F. van Harmelen, and D. Fensel. It's a streaming world! reasoning upon rapidly changing information. *IEEE Intelligent Systems*, 24(6):83–89, 2009.

10 F. Zablith, G. Antoniou, M. d'Aquin, G. Flouris, H. Kondylakis, E. Motta, D. Plexousakis, and M. Sabou. Ontology evolution: a process-centric survey. *Knowledge Eng. Review*, 30(1):45–75, 2015.

# 8 Evaluation of Knowledge Graphs

*Heiko Paulheim (Universität Mannheim, DE), Marta Sabou (TU Wien, AT), Michael Cochez (Fraunhofer FIT - Sankt Augustin, DE), and Wouter Beek (University of Amsterdam, NL)*

As there are more and more efforts to build knowledge graphs that complement the "mainstream" KGs such as DBpedia and Wikidata, and a plethora of work that try to improve those knowledge graphs in various directions (e.g., adding missing pieces of information, or flagging incorrect axioms), there is a growing need to define the standards for an evaluation. Moreover, since each research work has to prove itself against the state of the art, there is a stronger focus on reproducibility of scientific experiments, also for knowledge graphs. This chapter discusses some questions and guidelines regarding evaluation methods and protocols for knowledge graphs.

## 8.1 Introduction

The evaluation of conceptual models has its roots in the field of Knowledge-based systems and was adapted to the evaluation of Description Logics ontologies popular in Semantic Web [9], leading to a vibrant field [6, 8, 29, 24]. Ontology evaluation focuses on "checking the technical quality of an ontology against a frame of reference" such as a gold standard ontology, a representative domain corpus, a specification document, or general human knowledge [9]. Evaluation activities have the goal of assessing the ontology's domain coverage, quality of modeling (syntactic, structural, semantic quality), suitability for an application, or community adoption [24]. Evaluation goals are achieved with evaluation methods. Metrics-based methods assess ontology quality by computing a numeric value based on its characteristics [6, 8]. Verification methods identify defects (a.k.a. errors, or pitfalls) in the ontology, i.e., a set of issues related to a part of the ontology that should be corrected [22].

When discussing knowledge graph evaluation, there can be two possible targets:

1. Evaluating a knowledge graph as such, and
2. Evaluating techniques for constructing and/or improving knowledge graphs

The first target is rather related to data profiling [5, 14, 17] – i.e., which data exists and and in which quality – whereas the second is more concerned with the process of data creation and/or manipulation. Nevertheless, it can be argued that both of the targets are actually two sides of the same coin. In both cases, the object of study is a knowledge graph, so that the same metrics can be applied. Evaluating a knowledge graph is also implicitly an evaluation of the process that created that KG, and evaluating a method for construction or improvement of a knowledge graph is usually done by evaluating the knowledge graph that is the outcome of that process [18].

## 8.2 Evaluation Setups

On a coarse grained level, we can distinguish two orthogonal dimensions: intrinsic vs. task-based evaluation, and automatic vs. human-centric evaluation.

### 8.2.1 Intrinsic vs. Task-based Evaluation

Intrinsic evaluation only looks at the knowledge graph per se. It measures, e.g., the size, the fraction of correct statements, or the completeness. A larger set of quality metrics for intrinsic evaluation have been proposed in the literature [7, 30]. While these measures are intuitive and objective, they can only be a first attempt to evaluate a knowledge graph.

Knowledge graphs are not created as a means in themselves, thus, it is questionable whether intrinsic evaluations can be the only means of evaluation, or whether we should rather measure the quality of a knowledge graph by the added value it brings on an actual task, such as question answering or recommender systems. The benefit of this task-based of evaluation is that it clearly shows the potential impact of the system under scrutiny. Another aspect of this type of evaluation is that it can show how efforts from different communities can be integrated.

The main difficulty with task-based evaluations is that the interpretation of the results can become more difficult. Often there are many aspects involved in running an evaluation, and it is not always clear how they interact. In other words, even when the results of the task are looking good, it might be that the actual performance is explainable by the interplay of the other components. It is important to point out that neither of the two can serve as a proxy or approximation for the other. A knowledge graph with good measures on internal quality may perform poorly in a specific task (e.g., since it may have a good quality globally, but bad quality in the domain at hand), and vice versa.

### 8.2.2 Automatic vs. Human-centric evaluation

While some evaluation protocols can be fully automated, especially if there is a gold standard available, others cannot. Thus, human-centric evaluation is often used at least as one building block for knowledge graph evaluation.

It is clear that this kind of evaluation is not equally feasible for all tasks which are reported in research work. Here we think especially about evaluation processes with a human in the loop, evaluations which require special infrastructure or access to datasets which are not publicly available. An important research question refers to identifying those evaluation tasks that cannot (at this moment) be performed with automatic techniques, but rather require input from humans. Some examples are: checking the freshness of the information (i.e., whether it is up-to-date), checking completeness (e.g., does a KG contain *all* German cities) [23], correctness of domain knowledge (e.g., was a person born in the given place or not); correctness appropriateness of modeling decisions (e.g., whether some entities should be modeled as concepts or instances, whether partonomy is modeled as subsumption). Necessarily, the types of tasks will dictate the choice of the suitable human subjects (e.g., experts vs. laymen), as well as the most suitable human computation approach (e.g., gamification [12, 26] vs. expert-sourcing vs. crowdsourcing [1, 13]). A further challenge refers to how to scale up human-centric evaluation [19], especially by combining with automatic approaches [20]. Active learning approaches [25] are a possible solution, but have not yet been applied in the evaluation.

## 8.3 Reproducibility

Reproducibility is an essential part of evaluation. Only if the experiments performed are reproducible, one can independently verify whether reported results are factual. Besides, one can then compare the presented result with the results obtained from different experiments. Or at least see whether these results are in fact produced in similar conditions and whether a comparison would make sense. [21] In our community, we have more or less generally accepted tasks and even evaluation frameworks for SPARQL querying [4, 16], reasoning [10], natural language question answering [27], entity linking [28], and ontology matching [2].

We identified at least one other community which has a reproducibility initiative - i.e., the database community has reproducibility guidelines for SIGMOD[6]. With these guidelines, each paper has the option to prove its reproducibility by sending the code, data, and parameter settings for the experiments to a review board. They will then rerun the experiments to see whether the same, or at least very close results can be obtained. In this process, the evaluator will also investigate how sensitive the evaluation is to changes in both the input data and changes in the parameter settings.

Currently, reproducing or even comparing research results is difficult due to various datasets, protocols, and metrics used (but not always documented) in different experiments [18]. E.g., a sentences such as "‘We achieve an F1 score of 0.89 for type prediction on DBpedia"’ is usually not enough to reproduce the results. Hence, in order to come to reproducible and comparable results, the characteristics of the experiments carried out need to be specified along various dimensions. Those include:

1. Dataset(s) / KG(s) used
2. Evaluation protocol
3. Evaluation metrics
4. Tasks (in case of task-based evaluation)

### 8.3.1 Specifying Datasets and Knowledge Graphs

Specifying the dataset(s) or KG(s) used is the first step towards reproducible results. This does not only include referring to a dataset by name (e.g., “DBpedia”), but being as specific as possible. Recommended attributes to be reported include: which version was used? which subset of the dataset (if any)? The same applies to external sources of knowledge used (e.g., text corpora), if any. One possible way to verify the replicability of the experiments is to report a content-based hash of the dataset(s) used for the evaluation. In case of KGs that are not release-based (such as DBpedia), but constantly changing (such as Wikidata), either a snapshot of the version used or clear instructions on how to obtain the version used should be included.

As far as datasets or KGs are concerned, well-known knowledge graphs like DBpedia are the most widely used [18]. However, the use of synthetic knowledge graphs has also been proposed [11, 15]. While evaluations encompassing a larger variety of KGs are clearly better suited to harden the evidence that an approach works well in general (and not just on a specific KG), evaluations on specific knowledge graphs still have their own utility, e.g., when demonstrating a solution for a particular domain.

### 8.3.2 Specifying the Evaluation Protocol

The evaluation protocol is as important as the datasets. For example, for machine-learning based approaches, cross or split validation may be used, and the random seed for folds can have an impact on the results as well.

[6] http://db-reproducibility.seas.harvard.edu/

### 8.3.3 Specifying Evaluation Metrics

There are quite a few evaluation metrics, and although there is a wide adoption of recall, precision, and F1-score, these are not the only metrics used (and in many cases, it may make sense to use other metrics as well). Here, it is also important to be as specific as possible. Typical distinctions include: macro vs. micro average, subset of entities on which the evaluation is carried out (e.g., for type prediction: is the evaluation only carried out on previously untyped entities?), and exact computation of the metrics (e.g., is the prediction of the type `owl:Thing` for an entity counted as a true positive or ignored due to being trivial?).

### 8.3.4 Specifying Tasks

Finally, for task-based evaluations, the task has to be specified with equal care. For example, for evaluating the performance of particular KGs in tasks like entity linking, question answering, or recommender system, it is important to describe both the task and the KGs used along the dimensions above.

## 8.4 Recommendations and Conclusions

As we can assume that there is no single approach for creating and/or refining a knowledge graph that works best in all tasks and for all knowledge graphs, one higher level goal of evaluations (which is rarely addressed in current research) is to understand which approaches work well under which characteristics of the KGs (e.g.: size, connectivity, etc.). Therefore, when conducting more systematic evaluations, those can be done either by synthesizing datasets with different variations, or by using a variety of datasets with different characteristics. Therefore, datasets for such evaluations may include:

1. Synthetic datasets, which could rely on some existing benchmark datasets such as LUBM [11] or SP$^2$B [16]. Those can be used both for systematic testing of scalability, as well as for analyses of the behavior of approaches when varying certain other properties of the KG.
2. Alternatively, we could index existing datasets in terms of relevant properties that are relevant for evaluation algorithms. These properties include: (a) graph/network properties (degree, connectedness); (b) descriptive statistics of graph elements (e.g., number and distribution of classes, property types, entities); (c) expressiveness of the data (e.g., RDF(S), OWL); (d) whether the dataset is fully materialized or contains implicit knowledge; (e) natural language metrics (languages used, presence/absence of natural language information); (f) the presence of specific domains (e.g., use of certain namespaces, presence of geospatial data); (g) other: what were the main mechanisms for creation. Note that many of these properties are easy to compute automatically, for example, as in LOD Laundromat [3]. One issue is that they are mostly computed at the level of file, rather than at the level of datasets/KGs.
3. Hybrid approaches, e.g., use properties calculated over (2) in order to improve the generation of synthetic benchmarks [15].[7]

There is still some steps to be taken from the way evaluations are mostly carried out today and the vision sketched above. As it is hard to enforce a change over night, we suggest we

[7] See http://ldbcouncil.org for an existing initiative.

propose to have a similar effort ongoing in the major semantic web conferences. One option is to award a "‘most reproducible paper"’ award. This way, we can increase the credibility of our research and hopefully get more reuse of existing effort. This would likely also lead to more research software being open sourced and further built upon as code which is under scrutiny of a reviewer will be written much cleaner.

### References

1 Maribel Acosta, Amrapali Zaveri, Elena Simperl, Dimitris Kontokostas, Sören Auer, and Jens Lehmann. Crowdsourcing linked data quality assessment. In *International Semantic Web Conference*, pages 260–276. Springer, 2013.

2 Alsayed Algergawy, Michelle Cheatham, Daniel Faria, Alfio Ferrara, Irini Fundulaki, Ian Harrow, Sven Hertling, Ernesto Jiménez-Ruiz, Naouel Karam, Abderrahmane Khiat, Patrick Lambrix, Huanyu Li, Stefano Montanelli, Heiko Paulheim, Catia Pesquita, Tzanina Saveta, Daniela Schmidt, Pavel Shvaiko, Andrea Splendiani, Elodie Thiéblin, Cássia Trojahn, Jana Vataščinová, Ondřej Zamazal, and Lu Zhou. Results of the ontology alignment evaluation initiative 2018. In *Ontology Matching*, 2018.

3 Wouter Beek, Laurens Rietveld, Hamid R Bazoobandi, Jan Wielemaker, and Stefan Schlobach. LOD laundromat: a uniform way of publishing other people’s dirty data. In *International Semantic Web Conference*, pages 213–228. Springer, 2014.

4 Christian Bizer and Andreas Schultz. The Berlin SPARQL benchmark. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 5(2):1–24, 2009.

5 Christoph Böhm, Felix Naumann, Ziawasch Abedjan, Dandy Fenz, Toni Grütze, Daniel Hefenbrock, Matthias Pohl, and David Sonnabend. Profiling linked open data with ProLOD. In *Data Engineering Workshops (ICDEW), 2010 IEEE 26th International Conference on*, pages 175–178. IEEE, 2010.

6 Christopher Brewster, Harith Alani, Srinandan Dasmahapatra, and Yorick Wilks. Data Driven Ontology Evaluation. In *Proc. Int. Conf. on Language Resources and Evaluation (LREC)*, 2004.

7 Michael Färber, Frederic Bartscherer, Carsten Menne, and Achim Rettinger. Linked data quality of DBpedia, Freebase, OpenCyc, Wikidata, and YAGO. *Semantic Web*, (Preprint):1–53, 2016.

8 Aldo Gangemi, Carola Catenacci, Massimiliano Ciaramita, and Jos Lehmann. Modelling Ontology Evaluation and Validation. In *Pro. Int. Semantic Web Conf.*, pages 140–154. Springer, 2006.

9 Asunción Gómez-Pérez. Ontology Evaluation. In Steffen Staab and Rudi Studer, editors, *Handbook on Ontologies*, pages 251–273. Springer, 2004.

10 Rafael S Gonçalves, Samantha Bail, Ernesto Jiménez-Ruiz, Nicolas Matentzoglu, Bijan Parsia, Birte Glimm, and Yevgeny Kazakov. OWL reasoner evaluation (ORE) workshop 2013 results. In *ORE*, pages 1–18, 2013.

11 Yuanbo Guo, Zhengxiang Pan, and Jeff Heflin. LUBM: A benchmark for owl knowledge base systems. *Web Semantics: Science, Services and Agents on the World Wide Web*, 3(2-3):158–182, 2005.

12 Jörn Hees, Thomas Roth-Berghofer, Ralf Biedert, Benjamin Adrian, and Andreas Dengel. BetterRelations: using a game to rate linked data triples. In *Annual Conference on Artificial Intelligence*, pages 134–138. Springer, 2011.

13 Dimitris Kontokostas, Amrapali Zaveri, Sören Auer, and Jens Lehmann. TripleCheckMate: A tool for crowdsourcing the quality assessment of linked data. In *International Conference on Knowledge Engineering and the Semantic Web*, pages 265–272. Springer, 2013.

14 Huiying Li. Data profiling for semantic web data. In *International Conference on Web Information Systems and Mining*, pages 472–479. Springer, 2012.

**15** André Melo and Heiko Paulheim. Synthesizing knowledge graphs for link and type prediction benchmarking. In *European Semantic Web Conference*, pages 136–151. Springer, 2017.

**16** S Michael, H Thomas, L Georg, and P Christoph. Sp2Bench: a SPARQL performance benchmark. In *ICDE*, 2009.

**17** Felix Naumann. Data profiling revisited. *ACM SIGMOD Record*, 42(4):40–49, 2014.

**18** Heiko Paulheim. Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic web*, 8(3):489–508, 2017.

**19** Heiko Paulheim and Christian Bizer. Improving the quality of linked data using statistical distributions. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 10(2):63–86, 2014.

**20** Heiko Paulheim and Aldo Gangemi. Serving DBpedia with dolce – more than just adding a cherry on top. In *International Semantic Web Conference*, pages 180–196. Springer, 2015.

**21** Roger D Peng. Reproducible research in computational science. *Science*, 334(6060):1226–1227, 2011.

**22** Maria Poveda-Villalón, Asunción Gómez-Pérez, and Mari Carmen Suárez-Figueroa. OOPS!: A Pitfall-Based System for Ontology Diagnosis. In Miltiadis D. Lytras, Naif Aljohani, Ernesto Damiani, and Kwok Tai Chui, editors, *Innovations, Developments, and Applications of Semantic Web and Information Systems*, pages 120–148. IGI Global, 2018.

**23** Daniel Ringler and Heiko Paulheim. One knowledge graph to rule them all? In *German Conference on Artificial Intelligence*, 2017.

**24** Marta Sabou and Miriam Fernández. Ontology (network) evaluation. In Mari Carmen Suárez-Figueroa, Asunción Gómez-Pérez, Enrico Motta, and Aldo Gangemi, editors, *Ontology Engineering in a Networked World.*, pages 193–212. Springer, 2012.

**25** Burr Settles. Active learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 6(1):1–114, 2012.

**26** Katharina Siorpaes and Martin Hepp. Games with a purpose for the semantic web. *IEEE Intelligent Systems*, 23(3), 2008.

**27** Christina Unger, Corina Forascu, Vanessa Lopez, Axel-Cyrille Ngonga Ngomo, Elena Cabrio, Philipp Cimiano, and Sebastian Walter. Question answering over linked data (QALD-4). In *Working Notes for CLEF 2014 Conference*, 2014.

**28** Marieke Van Erp, Pablo N Mendes, Heiko Paulheim, Filip Ilievski, Julien Plu, Giuseppe Rizzo, and Jörg Waitelonis. Evaluating entity linking: An analysis of current benchmark datasets and a roadmap for doing a better job. In *LREC*, volume 5, page 2016, 2016.

**29** Denny Vrandečić. *Ontology Evaluation.* PhD thesis, Karlsruhe Institute of Technology, 2010.

**30** Amrapali Zaveri, Anisa Rula, Andrea Maurino, Ricardo Pietrobon, Jens Lehmann, and Sören Auer. Quality assessment for linked data: A survey. *Semantic Web*, 7(1):63–93, 2016.

# 9 Combining Graph Queries with Graph Analytics

*Dan Brickley (Google Research - Mountain View, US), Aidan Hogan (IMFD, DCC, University of Chile - Santiago de Chile, CL), Sebastian Neumaier (Wirtschaftsuniversität Wien, AT), and Axel Polleres (Wirtschaftsuniversität Wien, AT)*

## 9.1 Introduction

The topics of data analytics and querying in the context of Knowledge Graphs have been addressed as part of two separate fields. However, most data processing pipelines using Knowledge Graphs require interleaving analytical and query tasks. While there exists infrastructure (languages, tools, algorithms, optimisations, etc.) for performing queries and analytics as separate processes, currently there does not exist an infrastructure for integrating the two. Still, many conceptual questions that a domain expert may wish to ask imply such a combination. There are several applications for combining analytical algorithms and querying relevant to Knowledge Graphs, for example:

- Ranking query results (e.g., ordering solutions based on the centrality of nodes in a graph),
- selecting topical sub-graphs to query (e.g., performing community detection to run queries on parts of a graph relevant to a given topic),
- exploratory search (e.g., finding weighted shortest paths between pairs of nodes returned as results for a query),
- dataset search (e.g., considering various graph metrics to select an external dataset suitable to querying), or
- data quality issues (e.g., analysing the connectivity of the graph).

These examples illustrate that in some cases we may wish to query the results of an analytical process, in other cases we may wish to perform analysis on the results of a query, or in other cases still, we may wish to interleave various query/analysis steps.

In this chapter we attempt to identify some essential steps towards combining graph querying and analytics in terms of useful features. We briefly discuss the state-of-the-art standard technologies to implement these features. We shall also turn towards questions about how to implement those features in a scalable manner, and missing bits and pieces with respect to these standard technologies. With currently available technologies we likely end up with the necessity to store and transfer graphs between different systems and stores to enable all of these features, due to the non-availability of a single system and engine to implement them. We argue that (extensions of) RDF and SPARQL seem to be the most suitable anchor points as a crystallisation point to enable such interchange and integration of query and analytics features.

## 9.2 Potential Starting points & Prior attempts

Although there have been proposals of various languages for querying graphs [2], including for example Cypher [11] and G-CORE [1], in the Semantic Web community, SPARQL [6] has been set as the standard query language for (Knowledge) Graphs, until now remaining the only graph query language backed by a standardisation body and implemented by numerous engines. The following discussion thus focuses on SPARQL, though the topics covered generalise also to other query languages for graphs, such as those mentioned.

Since the original standardisation of SPARQL [9], the scientific community has proposed lots of useful extensions for this language in terms of analytics and data processing features and combinations with other languages. This has led to SPARQL 1.1. which was a conservative extension of features agreed, by the W3C, to be key to the future of the language, essentially taking on board and consolidating the most urgent of these proposed features.

However, as for specific connections to graph analytics, apart from basic path query and aggregation features, many issues and in meanwhile urgent features remain unaddressed. In particular, core features relating to graph algorithms and network analysis have not found their way into the standard, despite being part of many typical (knowledge) graph processing pipelines.

While there have been attempts to combine SPARQL with other Turing-complete languages, e.g. Spark, Gremlin [3], XSPARQL [4], which would allow to address and implement all such features – herein, we rather aim at investigating which are the core features and tasks that typically are needed and that would deserve to be added as first-class citizens (or built-ins) in such a language.

Likewise, extensions of typical analytics languages like R [12], working on data frames, have simple libraries to import/incorporate SPARQL results tables as such data frames, but not allowing per se the reuse of analytical results as graphs again in a SPARQL-like query language, nor providing an integrated graph analytics and query language.

Also, potentially interesting starting points are widely-used graph analysis systems outside of the Semantic Web world; to name a few, e.g.: Shark [14], that allows to run SQL queries and sophisticated analytics functions; Google's Pregel [8], a system to efficiently process large graphs (of billions of vertices and trillions of edges) which powers Google's PageRank; as well as frameworks built on top of Apache Spark [13, 5], as well as various academic projects such as Signal-Collect [10].

## 9.3 Motivating Examples

Before continuing, we enumerate some motivating examples that help to illustrate the importance of considering queries and analytics in a unified framework. We will consider a hypothetical Knowledge Graph of bibliographical data considering scientific articles, the articles they cite, where they were published, their authors, their fields, and relations between fields. Potentially relevant questions on such a Knowledge Graph include:

- Find sub-communities of Computer Science in Mexico.
- Find the most important papers in AI published in IJCAI.
- Find connections (paths) from researchers in UChile and UBA.

Such questions involve goals that are naturally expressed through queries (Computer Science papers, authors in Mexico, papers in IJCAI, researchers in UChile, etc.) and goals that are naturally expressed through analytics (sub-communities, important papers, connections). Inspecting these questions, we can see that querying and analytical goals are interleaved, where we may wish to analyse a graph produced as the result of a query, or querying a graph enriched with the results of analyse, or any such combination.

Rather than pushing data between separate querying and analytical frameworks, the goal would be to combine both into one framework, allowing for the design of a unified language, hybrid algorithms optimised to consider all goals, as well as practical tools, interfaces and implementations.

### Graph Analysis Requirements:

Let us consider some of the common types of algorithms used in the graph analytics community that could be interesting to combine with queries in a unified framework.

- *Centrality.* Centrality of graphs can serve as indicators of finding the most important/most influential vertices of a graph. As an example, centrality measure would allow, e.g., an analysis of the most influential papers in a network of publications, cross-citations, and co-authorships (given the above example bibliographic Knowledge Graph).
- *Community structure/detection.* A graph is said to have communities if there are densely connected structures that can be grouped in node subsets. Community detection algorithms, such as minimum-cut algorithms, allow to discover these sub-communities, which, for instance, could relate to a connected sub-community of researchers, given the above example.
- *Paths/Flows.* A path in a graph generically denotes a connection between two nodes that may traverse multiple edges. Various technical definitions exist that restrict the set of valid paths between such nodes, including simple paths that do not visit the same node twice, or regular path queries that restrict the labels of edges that can be traversed by the path [2]. Additionally, extensions of such regular path queries with more complex conditions on properties have been defined, which are particularly important when dealing with graph data beyond "flat" RDF, such as property graphs that express provenance or other contextual information along the edges.
- *Vertex similarity.* There exist measures for "vertex similarity" that capture the relatedness of nodes in a graph by considering the neighbours they have in common and/or the specificity of the paths that exist between them. These methods allow to understand what connects nodes, and, thereafter, in what ways they are similar.
- *Connectivity/Spanning trees.* The connectivity of a graph – defined as the number of vertices or edges that need to be removed to disconnect the graph – in the context of Knowledge Graphs allows to analyse the resilience and (un)reachability of components. Also, related to the connectivity is the spanning tree of such a graph.

## 9.4 Semantic Graph Analytics

There are various data models that can be used to describe "graphs", including, for example, directed-edge labelled graphs, property graphs, and so forth. However, many of the traditional graph algorithms – though their generality and usefulness have been well-established in a variety of domains – are proposed and studied for simple graphs or directed graphs without labels. Hence the question arises of how to adapt and apply these algorithms to other structures; there may be many options to "project" out a directed graph from a more complex Knowledge Graph model, where each such projection may yield radically different results; respectively, depending on how the Knowledge Graph is stored, computing such a projection and transforming it into a format amenable to these algrotithms itself might impose a significant effort.

Aside from structure, Knowledge Graphs often embed semantics of domain terms expressed, for example, using formal model theory. Such semantics then permit reasoning methods that allow for transforming or extending graphs in a manner that preserves truth (i.e., applying inference); examples include subclass reasoning, or inferences over transitive or inverse properties, identity reasoning, and so forth [2]. Applying analytics before or after such transformations may again yield radically different results, and hence it is important to study such differences, and to study (and justify/evaluate) which transformations better reflect the real-world phenomena under analysis.

Projections here can involve "Inference-based transformations", i.e. materialisation or core-reduction (e.g. removing transitives or inverse edges to reduce a graph to its raw form, resolving non-unique names by choosing canonical representatives for an equivalence class) wrt. semantic rules (related to e.g. spanning trees computation). That is, when doing analytics one often needs to be aware that "semantically equivalent" graphs (with respect to the chosen KG semantics) may behave fundamentally differently when taken as inputs for graph analytics steps.

## 9.5 Conclusions and Next Steps

To present the herein discussed topics to a broader and appropriate audience we plan to submit an extended version of this report as a position paper to the upcoming W3C Workshop on Web Standardization for Graph Data.[8] The scope of the Workshop includes requirements for graph query languages and different kinds of reasoning in graph database systems. Also, it aims at bringing together the adjacent worlds of RDF and Property Graphs (cf. for instance [7]), to achieve productive and interoperable boundaries, and foster information exchange and mutual awareness.

### Acknowledgements

Axel Polleres' and Sebastian Neumaier's work is supported by the EC under the H2020 project SPECIAL and by the Austrian Research Promotion Agency (FFG) under the projects "CitySpin", "Communidata" and "DALICC". Aidan Hogan is funded by the Millennium Institute for Foundational Research on Data (IMFD) and Fondecyt Grant No. 1181896.

### References

1 Renzo Angles, Marcelo Arenas, Pablo Barceló, Peter A. Boncz, George H. L. Fletcher, Claudio Gutierrez, Tobias Lindaaker, Marcus Paradies, Stefan Plantikow, Juan F. Sequeda, Oskar van Rest, and Hannes Voigt. G-CORE: A core for future graph query languages. In Gautam Das, Christopher M. Jermaine, and Philip A. Bernstein, editors, *Proceedings of the 2018 International Conference on Management of Data, SIGMOD Conference 2018, Houston, TX, USA, June 10-15, 2018*, pages 1421–1432. ACM, 2018.

2 Renzo Angles, Marcelo Arenas, Pablo Barceló, Aidan Hogan, Juan L. Reutter, and Domagoj Vrgoc. Foundations of modern query languages for graph databases. *ACM Comput. Surv.*, 50(5):68:1–68:40, 2017.

3 Apache TinkerPop. TinkerPop3 Documentation v.3.2.5. http://tinkerpop.apache.org/docs/current/reference/, June 2017.

4 Stefan Bischof, Stefan Decker, Thomas Krennwallner, Nuno Lopes, and Axel Polleres. Mapping between RDF and XML with XSPARQL. *J. Data Semantics*, 1(3):147–185, 2012.

[8] https://www.w3.org/Data/events/data-ws-2019/

5 Ankur Dave, Alekh Jindal, Li Erran Li, Reynold Xin, Joseph Gonzalez, and Matei Zaharia. Graphframes: an integrated API for mixing graph and relational queries. In *Proceedings of the Fourth International Workshop on Graph Data Management Experiences and Systems, Redwood Shores, CA, USA, June 24 - 24, 2016*, page 2, 2016.

6 Steve Harris and Andy Seaborne. SPARQL 1.1 Query Language. W3C Recommendation, 2013.

7 Olaf Hartig. Reconciliation of RDF* and Property Graphs, Technical Report, *CoRR abs/1409.3288*, 2014.

8 Grzegorz Malewicz, Matthew H. Austern, Aart J. C. Bik, James C. Dehnert, Ilan Horn, Naty Leiser, and Grzegorz Czajkowski. Pregel: a system for large-scale graph processing. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2010, Indianapolis, Indiana, USA, June 6-10, 2010*, pages 135–146, 2010.

9 Eric Prud'hommeaux and Andy Seaborne. SPARQL Query Language for RDF. W3C Recommendation, 2008.

10 Philip Stutz, Daniel Strebel, and Abraham Bernstein. Signal/Collect12. *Semantic Web* 7(2): 139–166, 2016.

11 The Neo4j Team. The Neo4j Manual v3.0. http://neo4j.com/docs/stable/, 2016.

12 The R Foundation. The R Project for Statistical Computing. https://www.r-project.org, since 1992.

13 Reynold S. Xin, Joseph E. Gonzalez, Michael J. Franklin, and Ion Stoica. Graphx: a resilient distributed graph system on spark. In *First International Workshop on Graph Data Management Experiences and Systems, GRADES 2013, co-loated with SIGMOD/PODS 2013, New York, NY, USA, June 24, 2013*, page 2, 2013.

14 Reynold S. Xin, Josh Rosen, Matei Zaharia, Michael J. Franklin, Scott Shenker, and Ion Stoica. Shark: SQL and rich analytics at scale. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2013, New York, NY, USA, June 22-27, 2013*, pages 13–24, 2013.

## 10 (Re)Defining Knowledge Graphs

*Aidan Hogan (IMFD, DCC, University of Chile - Santiago de Chile, CL), Dan Brickley (Google Research - Mountain View, US), Claudio Gutierrez (IMFD, DCC, University of Chile - Santiago de Chile, CL), Axel Polleres (Wirtschaftsuniversität Wien, AT), and Antoine Zimmermann (École des Mines de Saint-Étienne, FR)*

The phrase "Knowledge Graph" has recently gained a lot of attention in both industry and academia. But what is a "Knowledge Graph"? Several definitions have been proposed but – we shall argue – fall short of capturing the full generality of the usage of the term. We argue for a looser, more permissive definition that may be instantiated in various concrete ways, setting the stage for the study and practice of "Knowledge Graphs" to become a commons that unites – rather than divides – previously disparate areas of Computer Science, focused on a shared goal: using graphs as a medium to make sense of large-scale, diverse data.

## 10.1 Introduction

Since the launch of the Google Knowledge Graph in 2012 – and subsequent announcements of Knowledge Graphs by companies such as AirBnB, eBay, Elsevier, Facebook, Microsoft, Springer Nature, and others – the notion of a "Knowledge Graph" has crystallised a number of efforts that draw upon a variety approaches for collecting, managing, integrating, publishing, annotating, processing and analysing diverse data using a graph abstraction.

Given its origins, the phrase "Knowledge Graph" (in its modern use) naturally encourages a pragmatic view of the data management and knowledge representation landscape. The Knowledge Graph (KG) viewpoint emphasises that the expense and difficulty of curating and extracting knowledge from large-scale data motivates interdisciplinary collaboration around common data structures, knowledge extraction and knowledge representation techniques. Initiatives, datasets and systems that self-describe as "KGs" are not (and, we argue, should not be) dominated by a specific scientific research field or application domain, but rather should be seen as a commons within which various complementary perspectives are combined, involving not only academia, but also industry, public organisations, developers, etc.

In terms of academic stakeholders, research on KGs should bring together techniques from scientific disciplines such as Knowledge Representation, Machine Learning, Semantic Web, Databases, Natural Language Processing, Multimedia Processing and Information Extraction, amongst others, leading to applications in a variety of domains such as Life Sciences, Library Science, Astronomy, Economics, Sociology, and more besides.

This inclusive view then leads to a contentious but key question that we address here: "*What is a Knowledge Graph?*". We first begin by reviewing several prior attempts to answer this question, concluding that many of the definitions proposed recently in the literature are too narrow-focused, adding technical requirements that – while concretising the notion of a Knowledge Graph – exclude other viewpoints; some of these definitions arguably even preclude the industrial KGs responsible for the recent popularisation of the phrase. We thus aim to (re)define a Knowledge Graph not based on what it denotes, but rather by what it has become to connote: the use of graphs to represent data from which knowledge can (later) be composed. Our goal with this (re)definition of Knowledge Graphs is to position the topic as a commons that can benefit from work combining various disciplines, outlining a more general scope within which various concrete definitions and research questions can coexist.

## 10.2 Knowledge Graphs: Background

Several works have attempted to provide definitions of what a knowledge graph is (or isn't). We provide a non-exhaustive collection of examples herein to serve as general background.

Long before Google popularised the phrase "Knowledge Graph", mentions can be found in the scientific literature. In 1974, Marchi and Miguel [6] defined a "Knowledge Graph" as a mathematical structure with vertices as knowledge units connected by edges that represent the prerequisite relation; this implies that units of knowledge are only accessible if other units are previously known. In the late 1980's, Bakker, in his Ph.D. Thesis [1], developed his notion of a "Knowledge Graph" as a way of structuring and representing text encoding scientific knowledge. In 1994, van der Berg [9] presented an extension of this work using First Order Logic to model consistency and implication in such Knowledge Graphs. Though related, we assume that the modern incarnation of the notion of a "Knowledge Graph" was derived independently from such earlier definitions; these independent inventions of the phrase do indicate some level of "naturalness" of the abstract idea, which can also be seen in similar proposals, for example, of "Semantic Networks" [3], though under a different name.

Nor did the 2012 announcement of the Google Knowledge Graph appear out of the blue: the direct lineage of the Google Knowledge Graph can, in fact, be traced back to a 2000 essay by Hillis [5] outlining his vision of "Aristotle": a knowledge web in the form of an online database "*organized according to concepts and ways of understanding them [containing] specific knowledge about how the concepts relate, who believes them and why, and what they are useful for*". Hillis would later go on to co-found (in 2005) the Metaweb company, which oversaw the development of the collaboratively-edited Freebase knowledge-base [2]. Metaweb in turn was acquired by Google in 2010, with Freebase subsequently forming an important source for the Google Knowledge Graph (with key involvement from ex-Metaweb personnel), as well as the collaboratively-edited Wikidata knowledge-base [8, 10]. These developments have directly led to the recent popularisation of the phrase "Knowledge Graph".

Turning to more recent times, the following descriptions have been provided by various participants of the Dagstuhl Seminar 18371, Sept. 2018, within the plenary discussions:

- "*A graph-structured knowledge-base*"
- "*Any Dataset that can be seen through the lense of a Graph perspective*"
- "*Something that combines data and semantics in a graph structure*"
- "*Structured data organisation with a formal semantics and labels [that is] computationally feasible & cognitively plausible*"
- "*[Knowledge Graphs are defined] by example: Babelnet, OpenCyc, DBpedia, Yago, Wikidata, NELL and their shared features*"

We see some varying and sometimes orthogonal approaches to the definition, including some descriptions that focus on the graph abstraction, some that emphasise the role of (formal) semantics, one that emphasises the importance of cognitive understanding, with the latter description rather proposing that Knowledge Graphs should be defined extensionally by looking at common characteristics of a class of important graph-structured datasets.

Various other proposed definitions of a "Knowledge Graph" appear in the literature, amongst which McCusker et al. [7] put forward that:

- "*a knowledge graph represents knowledge, and does so using a graph structure.*"

before coming to more binding technical criteria for defining a "Knowledge Graph":

- "*Knowledge graph meaning is expressed as structure.*"
- "*Knowledge graph statements are unambiguous.*"
- "*Knowledge graphs use a limited set of relation types.*"
- "*All identified entities in a knowledge graph, including types and relations, must be identified using global identifiers with unambiguous denotation.*"

Ehrlinger and Wöß [4] also collect proposed definitions, and even go as far as adding:

- "*A knowledge graph acquires and integrates information into an ontology and applies a reasoner to derive new knowledge.*"

The benefit of these latter more specific definitions would be to provide an initial technical agreement upon which further works can be elaborated and made interoperable. However, the consensus at the Dagstuhl Seminar – which included various industry stakeholders – was that these latter proposed definitions:

- are biased towards particular scientific disciplines (particularly the Semantic Web);
- define aspects that are in fact not essential for a Knowledge Graph, and thus, for example, are not satisfied by the industrial Knowledge Graphs that have played a key role in the recent popularisation of the phrase.

Upon reviewing prior proposals, it was decided to seek a more inclusive definition that repositions Knowledge Graphs as a commons for researchers and practitioners from various disciplines that are interested, more generally, in both the practical and scientific challenges stemming from the collection, management, integration, publication, annotation, processing and analysis of graph-structured data at scale. Beyond the (perhaps questionable) exercise of laying claim to "yet another definition" of a Knowledge Graph, our goal is to outline a scope and direction for this topic within which complementary perspectives can coexist.

## 10.3 Knowledge Graphs: A New Definition

Rather than assuming any specific core formalism for representing "knowledge" (e.g., rule-based axiomatisation, description logics, computational linguistics, machine learning models, relational schema and constraints), KGs start with data, and it is the data itself – organised and viewed as a graph of entities and their relationships – that takes centre stage. This graph representation allows data to cross application barriers, be aggregated and integrated at different levels of abstraction, without the encumbrance of sticking rigidly to a particular formalism: a particular schema, notion of logical consistency, ontological language, etc. Various mechanisms for extracting and representing "knowledge" can then be applied to complement the data in its graph abstraction, making explicit more of its meaning, allowing its interpretation in increasing depth and with increasing sophistication.

We thus propose to define a "Knowledge Graph", succinctly, as:

- "*a graph of data with the intent to compose knowledge*".

We elaborate on this definition in the following.

In terms of a "graph of data", we refer to a dataset viewed as a set of entities represented as nodes, with their relations represented as edges; technically this notion can be instantiated with a number of concrete graph models, including for example:

- **directed edge-labelled graphs** (aka sets of triples), composed of named binary relations (labelled edges) between entities (nodes);
- **property graphs**, which extends directed edge-labelled graphs such that both nodes and edges may be additionally annotated with sets of property-attribute pairs;
- **named graphs**, where rather than supposing one large graph, data are represented as a collection of (typically directed edge-labelled) graphs, each associated with an identifier.

We do not see a particular choice of graph model as being necessary for constituting a KG: any such graph model will suffice since – although different models may offer particular conveniences or give rise to particular challenges in particular scenarios – data in one model can be automatically converted to another with a suitable structural mapping. Hence the choice of graph model is not fundamental to the challenges that KGs address. On the other hand, the choice of such a graph model is not sufficient for constituting a KG: a randomly-generated property graph, for example, does not intend to compose knowledge.

We then view the conceptual shift from "data" to "knowledge" as characterised by the "interpretation" of the data. In terms of "composing knowledge", we view this process as starting with the graph of data, and as involving both the extraction and representation of knowledge – potentially drawing upon a variety of formalisms, descriptions, and techniques – in order to enrich the graph, and allow it to be interpreted, in greater and greater detail, by human and machine alike. Such knowledge may originate from the graph itself or from other complementary sources; the resulting knowledge may be represented as part of the graph abstraction, or as an attachment to the graph. Some directions in which the composition of knowledge may follow include, but are not limited to:

- describing the **formal semantics** of terms used in the graph, through (for example) logics founded in model theory; this increases the machine-interpretability of the underlying graph and allows for formal reasoning methods that can infer new data, detect inconsistencies, enable query answering over implicit knowledge, etc.;
- adding **lexical knowledge**, such as multilingual labels and descriptions, increasing the interpretability of the graph by humans who speak particular languages and by machines in relation to natural language in text documents, user questions, etc.;
- capturing the **completeness** and **bias** of the graph, denoting for example which parts of the graph are complete with respect to the real world, how representative are the entities captured in the graph in terms of the complete real-world population studied, etc.; this increases the interpretability of, for example, statistics and machine learning models built on top of the graph;
- providing **links** (particularly relating to identity) from the local graph to external datasets; such links increase the interpretability of the graph in relation to other datasets;
- representing **context** that may capture, for example, the provenance of particular elements of the graph, spatial or temporal settings in which (parts of) the graph are known to be valid, and so forth; such representations of context help to understand how the graph should be interpreted in different settings.

This is intended to be an illustrative rather than a complete list. By "composing knowledge", again we generally refer to a continual process of extracting and representing knowledge in a manner that enhances the interpretability of the resulting Knowledge Graph; there are of course other directions in which this idea could be followed. No single example is necessary to fulfil our definition of a "Knowledge Graph", but rather each gives a concrete direction in which the "intent to compose knowledge" could follow. We deliberately choose not to restrict this notion of the "intent to compose knowledge", but rather allow for different techniques that increase both machine and human interpretability for different purposes; for example, this neither precludes nor prescribes a goal that has often been mentioned in the context of the term Knowledge Graphs: to eventually serve as a key for the "explainability" of data and models built from and around data.

## 10.4 Knowledge Graphs: A Commons

Our definition establishes a deliberately low barrier to entry for what is considered a "Knowledge Graph" to not only fit the diverse views now established in practice, but also to encourage study across a variety of areas, fitting with our goal that it may become a commons for more interdisciplinary research. The study of "Knowledge Graphs" ideally involves participation of researchers from the variety of fields previously mentioned, benefiting from tighter collaborations between such fields, leading to novel research questions, theories and techniques being applied specifically to understanding how to compose knowledge from diverse data at scale. By scale we refer not only to volume, but also the notions of velocity, variety and veracity often referred to under the moniker of "Big Data".

On the other hand, we strongly encourage researchers who wish to refer to "Knowledge Graphs" as their object of study to rigorously define how they instantiate the term as appropriate to their investigation. Such a definition should begin by defining the particular graph-based model/view of data adopted by that particular study (while refraining from excluding other definitions). Thereafter, the study should clarify what is the form of "knowledge" that such a graph intends to compose, how this form of "knowledge" contributes to the interpretability of such graphs, and what techniques are proposed along those lines.

### Acknowledgements

Axel Polleres' work is supported by the EC under the H2020 project SPECIAL and by the Austrian Research Promotion Agency (FFG) under the projects "CitySpin", "Communidata" and "DALICC". Aidan Hogan is funded by the Millennium Institute for Foundational Research on Data (IMFD) and Fondecyt Grant No. 1181896.

### References

1 R. R. Bakker. *Knowledge Graphs: representation and structuring of scientific knowledge.* PhD thesis, University of Twente, Enschede, 1987.

2 K. Bollacker, C. Evans, P. Paritosh, T. Sturge, J. Taylor *Freebase: A Collaboratively Created Graph Database For Structuring Human Knowledge.* In *SIGMOD Conference*, pages 1247-1250, 2008.

3 A. Borgida and J. F. Sowa. *Principles of semantic networks - explorations in the representation of knowledge.* The Morgan Kaufmann Series in representation and reasoning. Morgan Kaufmann, 1991.

4 L. Ehrlinger and W. Wöß. Towards a Definition of Knowledge Graphs. In *International Conference on Semantic Systems (SEMANTiCS2016), Posters & Demos.* CEUR-WS.org, 2016.

5 W. D. Hillis. "Aristotle" (The Knowledge Web). Later republished in *Edge (May 2004)*, https://www.edge.org/conversation/w_daniel_hillis-aristotle-the-knowledge-web, 2000.

6 E. Marchi and O. Miguel. On the structure of the teaching-learning interactive process. *Int. Journal of Game Theory*, 3(2):83–99, 1974.

7 J. P. McCusker, J. S. Erickson, K. Chastain, S. Rashid, R. Weerawarana, and D. L. McGuinness. What is a Knowledge Graph? *Semantic Web*, 2018. (under review; available from http://www.semantic-web-journal.net/content/what-knowledge-graph).

8 T. Pellissier Tanon, D. Vrandecic, S. Schaffert, T. Steiner, and L. Pintscher. From Freebase to Wikidata: The Great Migration. In *International World Wide Web Conference (WWW)*, pages 1419–1428, 2016.

9 H. van den Berg. First-Order Logic in Knowledge Graphs. In C. Martín-Vide, editor, *Current Issues in Mathematical Linguistics*, volume 56 of *North-Holland Linguistic Series: Linguistic Variations*, pages 319–328. Elsevier, 1994.

10 D. Vrandečić and M. Krötzsch. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85, 2014.

## 11 Foundations

*Claudia d'Amato (University of Bari, IT), Sabrina Kirrane (Wirtschaftsuniversität Wien, AT), Piero Andrea Bonatti (University of Naples, IT), Sebastian Rudolph (TU Dresden, DE), Markus Krötzsch (TU Dresden, DE), Marieke van Erp (KNAW Humanities Cluster - Amsterdam, NL), and Antoine Zimmermann (École des Mines de Saint-Étienne, FR)*

Knowledge Graphs (KGs) are becoming more and more popular with an increasing interest of both big industrial players and scientific communities in different research fields such as Semantic Web (SW), Databases, Machine Learning and Data Mining. However, an agreed formal definition of KG is nowadays missing [15] and, as a consequence, also a shared view on the needed KG semantics. Starting from defining a KG, the attention of this document is devoted to fix the semantics that is needed for KGs and importantly, the requirements to be taken into account when fixing such a semantics. The importance of taking into account (different kinds of) contextual information is also analyzed and possible research directions for tackling this aspect are illustrated. Finally, we analyzed another issues related to accessing KGs, specifically when a fully open setting cannot be assumed, hence we report the main research questions that need to be addressed.

### 11.1 Introduction

Knowledge Graphs (KGs) are becoming more and more popular with an increasing interest of both big industrial players, such as Google and Amazon, and scientific communities in different research fields such as Semantic Web (SW), Databases, Machine Learning and Data Mining. However, despite this increasing interest in KGs, an agreed formal definition of KG is nowadays missing [15] and, as a consequence, also a shared view on the needed KG semantics. Starting from defining a KG as a graph-based structured data organization, endowed with formal semantics (i.e. a graph-based data organization where a schema and multiple labelled relations with formal meaning are available), the attention of this document is devoted to fix the semantics needed for KGs and importantly, the requirements to be taken into account when fixing the semantics. The main motivation for looking at this direction is that, as testified by recent studies on understanding the Empirical semantics [4, 14, 27, 43] in SW (where clear and formal semantics is often provided), the actual usage of formal languages by human experts does not always matches the formal specifications.

Particularly, in [4, 27, 43] the formal and actual meaning of `owl:sameAs` has been investigated, whilst in [14] the empirical proof that some semantics is encoded within IRIs is provided; as a consequence, meanings within IRIs are practically exploited thus generating polysemy issues of IRIs (similarly to texts) and wrong reuse of IRIs due to the misinterpretation of the IRI's intended meanings. Additional problems that have been experimentally shown are: the misuse of classes and instances (classes adopted as instances and vice-versa) [1], the incorrect interpretation of domain and/or range for properties, and the injection of logical inconsistencies due to wrong conceptualizations. The results of these empirical studies, the diffusion of tools aiming at limiting the usage of formal semantics [44], the very large size of (existing) KGs [15], that may also evolve by interlinking existing KGs (the Linked Open Data Cloud[9] can be considered an example in this direction), suggest that the semantics for KGs needs to satisfy the following basic requirements: facilitate interoperability, simplicity (of usage), cognitive plausibility. Nevertheless, besides these basics requirements, additional aspects and potential issues need to be taken into account. Specifically, KGs are meant to represent large corpora of knowledge, possibly referring to several, potentially interconnected, domains, as such conflicting information may arise. This may not only be due to the fact that one piece of information and its opposite are declared, e.g in different KGs, but also to additional factors that are currently almost disregarded. One of them is represented by the fact that the validity of (pieces of) knowledge can be: context dependent, e.g., there exists norms that are valid in some countries whilst they are not applicable in other countries; or it can be time dependent, e.g., Barack Obama as USA president is applicable only to a certain period of time. Many KG-related projects show the need to represent contexts: statement qualifiers in Wikidata; attributes in the property graph data model; temporal and spatial validity in Yago(2). Actually, the word context has several meanings and as such several forms of context may need to be taken into account. Specifically, the following kinds of context are considered of particular importance: a) temporal and spatial; b) domain, application, task and process; c) social, cultural, legal; d) provenance, sourcing circumstances, trust; e) "view" such as looking at knowledge from a perspective that is not entity-centric. This implies that the semantics for KGs needs to allow for expressing (different kinds of) contextual information while granting ambiguity and inconsistency. Consequently, approximate inferences from approximate statements could be allowed but, on the same time, the ability to make sound and complete inferences from correct facts is somehow lost.

Arguably, one semantics may not be able to adequately address all requirements, intended meanings, and usage scenarios. This means there is the need to allow for individual, diverging semantics. At the same time, the “intended” meaning and/or usage should be made transparent for the sake of interoperability, common understanding and appropriate reuse.

A possible solution is to allow the KG to be accompanied by (meta)information about how it is meant to be interpreted. Approaches to be considered are: a) the adoption of a declarative description of the KG formal semantics, e.g., in terms of model theory; b) the specification of a piece of code as an operational or procedural semantics; c) the usage of a pointer to a semantic profile already defined elsewhere.

In the following, possible research directions for tackling the issues and requirements illustrated above are presented.

## 11.2 Taking context into account

To tackle the issue of managing and specifying different kinds of contexts, two main research directions are envisioned:

1. Representing contexts explicitly by extending the representation (reminding the trade-off with the requirement of keeping things simple);
2. Discovering context by exploiting on the actual data available in KGs, e.g., finding consistent subgraphs.

[9] https://lod-cloud.net/

As regards the first direction, some proposals have been made in the literature such as:

- **Reification:** representing contextual information on the level of the graph structure (e.g., in plain RDF) by introducing auxiliary graph vertices to represent anything that needs to be annotated (a comparison of reification approaches is presented in [18]);
- **Named graphs, Nquads:** extensions of graph models that provide the "handle" to edges and groups of edge, avoiding reification [28, 10];
- **Property graph, Wikidata, attributed logics:** enriched graph models that add a second layer for representing contextual data;
- **Semiring annotations:** attaching a value from a well defined algebraic structure (semring) to all statements, that expresses to what extent the assertion can be considered "true". The value can capture contextual information such as provenance, probability, and access permissions in (graph) databases.

The variety of solutions reflects a basic syntactic question, namely, "What belongs in the KG and what belongs in the context?" Furthermore, is it necessary to separate context from data? Answering these syntactic questions represents one of the main priorities when considering the solution of representing contexts explicitly.

Additionally, the explicit representation of contexts also raises a basic related semantic question, that is: "If a statement holds in one context, can we infer that it also holds in another context?" In order to answer this question, hybrid reasoning approaches may need to be formalized and developed. Additionally, the following views need to be taken into account: crispness vs. uncertainty; discrete vs. continuous; declarative vs. procedural; unstructured vs. semi-structured vs. structured; monotonicity; inconsistency tolerance. Such a hybrid reasoning solution should be somewhere on the spectrum between latent semantics (e.g. embeddings in vector spaces) and model-based semantics, and should include statistics, graph theory, and (limited) natural language forms. This also implies that a sort of multidimensional semantics needs to be considered. The semantic question has been answered differently in the literature:

- Semiring-style semantics and annotated logics (such as Annotated Logic Programming [31], Annotated RDF and RDFS [46, 51]): define what can be entailed based on formal, logical semantics and non-obvious consequences can be obtained.
- Reification: makes contexts part of the normal graph data, which might still be evaluated under some general semantics (e.g., RDFS), but the method does not specify how to use context. Different reification models affect entailments differently, as shown in [21, 50]
- Named graphs and Nquads: do not offer a standard semantics for making entailments, but various, conflicting approaches can be used to formalize it [49].
- Property graphs: do not support entailment or formal semantics of any kind whilst attributed logics are a proposal for defining an entailment semantics for property-graphs [34].
- Logics where context is a first class citizen: an example is given by McCarthy's logic of context [39] which has been formalized in different ways [24, 9, 2, 8, 41] and applied to RDF [25].
- Logics where contexts are separated from the universe of discourse: based on Giunchiglia's approach of contextual reasoning [22] and local model theory [20] (DFOL, DDL, E-connection, Package-based DL, IDDL, CKR, E-SHIQ and other variations).

The inconvenients of the last two options are that the logical formalisms are diverse, complicated, and non intuitive. Despite of the multiple existing proposals, converging on a sufficiently flexible, yet not overly general approach remains a major challenge.

As regards the second research direction, that is discovering contexts, the works on empirical semantics (see the previous section) have shown that capturing the intended semantics starting from the evidence provided by the mass of data is actually doable. As such, by extensions, discovering contexts appears to be meaningful in principle. Particularly, latent semantic approaches [35, 36] could be exploited for discovering preliminary (even if weak) notion of context, whilst semantic data mining [37] methods could be extended for tackling more complex notions of contexts. Also pattern mining methods applied to semantically reach representations [19, 13] could be an important direction to be investigated. Specifically, in this case, the pattern discovery process should be goal driven, where goals should be given by different kinds of context to be possibly discovered.

The two research directions represent orthogonal views of the same issue: contextual information can be represented when available whilst additional contextual information can possibly be captured by exploiting the evidence coming from the data (contexts are described extensionally whilst no explicit (intensional) representation is provided). A mixture of the solutions developed in each direction is considered a valuable research perspective potentially delivering even more powerful results (e.g. on performing "context discovery") when applying it to rich data that has "explicit context".

## 11.3 Accessing Knowlege Graphs

Until now, mostly KGs in an open access scenerio have been implicitely considered. However, in broader scenarios, this would not be the case, whilst confidentiality requirements and usage constraints may arise, due to privacy concerns, laws, licensing, and more. The lack of technical instruments for regulating access to knowledge and its usage, may hinder the adoption of knowledge-based technologies in application contexts where such technologies may give significant contributions. Knowledge graphs – and the other forms of digitized, processable knowledge – play different roles with respect to constrained access and usage, as they can be both the object that must be "protected", and the specification of the constraints, namely the policies. The former role leads to specific confidentiality requirements such as being inference-proof. In other words, it should not be possible to reconstruct concealed information by (automatically) reasoning on the visible part. The latter role poses expressiveness requirements on the (knowledge based) policy language. Both roles pose scalability requirements, since the access control layer should not introduce unsustainable overhead. We use two scenarios to illustrate the importance of regulating access to knowledge, and the main open research challenges in this area. The scenarios consist of: (i) integrating different knowledge graphs with different licenses, and (ii) providing support for a knowledge graphs marketplace.

- In the marketplace use case, there is a need to build a marketplace for knowledge that could be used to speed up business processes and systems. The primary challenge relates to the automated matchmaking between knowledge owners and knowledge consumers. In order to support this matchmaking, we must be able to represent not only potential usage constraints (like licenses) but also the knowledge request in a manner in which it can be checked automatically.
- In the biomedical use case we need to give a license to a knowledge graph that has been composed from several other knowledge graphs, each with an associated license. Here the primary challenges relate to extracting terms of use from textual licenses, and representing individual licenses in a manner that enables license composition and conflict resolution.

In terms of enabling inference-proof secure access control for knowledge graphs, efforts in integrating symbolic AI with machine learning will introduce new challenges in preserving the confidentiality of knowledge. In the long term, where machine learning and symbolic knowledge feed into each other and the respective inference mechanisms are integrated or composed, new attack models arise. For example, symbolic knowledge may be exploited as background knowledge to "break" the confidentiality guarantees offered by differentially private learning mechanisms. Conversely, link prediction and other ML-based inferences may be used to attack the (supposedly) secure views on knowledge bases obtained with methods that take into account purely symbolic attacks.

## 11.4 Taking access constraints into account

In terms of supporting constrained access to knowledge graphs, fundamental challenges relate to associating various policies with data and knowledge, enforcing inference proof access control and ensuring knowledge confidentiality.

### 11.4.1 Associating policies with data and knowledge

Generally speaking, the sticky policy concept [29] is used to tightly couple usage policies to data and knowledge. When it comes to the state of the art, sticky policies are usually implemented by using cryptography to strongly associate policies with data [40, 42]. However, it is worth noting that there are currently no standard approaches for attaching policies to data in a linked data setting. Also, it is important to highlight that from a practical perspective it is not possible for said policies to be enforced automatically (more precisely, it is an honors system whereby data controllers and processors can choose to either obey the policy or not). One of the challenges with respect to existing approaches is that there is a need for a trusted third party to ensure that obligations specified in the policy are fulfilled. Methodologies and formats for linking data and policies in a semantic framework are currently being investigated in the SPECIAL H2020 project[10]. When it comes to usage control in the form of licensing, research topics range from using Natural Language Processing to extract license rights and obligations to licenses compatibility validation and Composition [47, 26, 23]. However, there are currently no standard license-aware data querying and processing mechanisms.

### 11.4.2 Inference-proof access control

[32] provide a detailed survey of the various access control models, standards and policy languages, and the different access control enforcement strategies for RDF. However, at the level of (possibly distributed) queries, linked data protocols do not currently support inference-proof access control in a standard way. Considering the array of access control specification and enforcement mechanisms proposed to date, a necessary first step towards is to develop a framework that can be used to evaluate existing access control offerings in terms of expressivity, correctness and completeness. Inference-proof access control for distributed data sources has been extensively discussed in a recent Dagstuhl workshop (n.17262, on Federated Data Management, 2017) in the context of federated query processing. A recent proposal in this direction is [17].

[10] SPECIAL H2020 project, https://www.specialprivacy.eu/

### 11.4.3 Knowledge confidentiality

Several proposals to enable confidentiality of RDFS and OWL knowledge bases that adopt simple confidentiality models exist (cf. [3, 11, 33, 45, 16]).

- These approaches are vulnerable to attacks based on meta-knowledge; this issue has been dealt with in [7, 6], that show how to construct robust secure views. However, this method probably does not scale sufficiently yet.
- Another limitation of this confidentiality criterion lies in its "crisp" nature: what if a secret is not entailed by the available (meta)knowledge but is very likely, given that same knowledge? We cannot still assume the secret to be effectively protected. The probabilistic and correlation information used for this kind of attacks can be obtained through both standard statistical analyses and machine learning algorithms. A refined, probabilistic confidentiality model has already been developed [5], but it is difficult to implement efficiently, given the inherent complexity of probabilistic reasoning. Moreover, the probabilistic criterion does not currently handle the knowledge produced by learning algorithms, that is not strictly probabilistic.
- Conversely, symbolic reasoning may be used to attack the confidentiality of privacy-preserving data mining algorithms. Such algorithms produce k-anonymous or differentially private outputs (a survey is available in deliverable D1.7 of the H2020 project SPECIAL). It is known that anonymization techniques are vulnerable to background knowledge [48, 30, 12, 38]. So it is important to investigate whether and how the symbolic background knowledge encoded in KGs can be used to leak confidential information.

Open research questions in relation to providing constrained access to Knowledge Graphs include:

- How does the whole protocol work?
- How do we attach policies to data and how do we query considering the policy? Can sticky policies be used?
- Can we support automatic negotiation using policies?
- Which additional information shall be removed to avoid inferring knowledge that should not be accessible?
- How do we avoid hybrid symbolic/ML attacks?

## 11.5 Conclusions

In this section we provided our envisioned defintion for Knowledge Graph, since an agreed formal definition of KG is nowadays still missing. Starting from defining a KG as a graph-based structured data organization, endowed with formal semantics we fixed the semantics that should be needed for KGs and most importantly, the necessary requirements to be taken into account when fixing the semantics. We particularly argued on the importance of taking (different kinds of) context and contextual information into account. Hence, we illustrated threee main research directions that we considered appropriate for the the purpose. Furthremore we argued on existing open issues concerning the access to KGs, when a fully open setting cannot be assumed.

## Acknowledgements

Sabrina Kirrane' work is supported by the EC under the H2020 project SPECIAL and by the Austrian Research Promotion Agency (FFG) under the project "DALICC".

### References

1 Luigi Asprino, Valerio Basile, Paolo Ciancarini, and Valentina Presutti. Empirical analysis of foundational distinctions in Linked Open Data. In Jérôme Lang, editor, *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden.*, pages 3962–3969. ijcai.org, 2018.

2 Guiseppe Attardi and Maria Simi. A Formalization of Viewpoints. *Fundamenta Informaticae*, 23(2-4):149–173, 1995.

3 Franz Baader, Martin Knechtel, and Rafael Peñaloza. A generic approach for large-scale ontological reasoning in the presence of access restrictions to the ontology's axioms. In *The Semantic Web - ISWC 2009, 8th International Semantic Web Conference, ISWC 2009, Chantilly, VA, USA, October 25-29, 2009. Proceedings*, pages 49–64, 2009. URL: https://doi.org/10.1007/978-3-642-04930-9_4, `doi:10.1007/978-3-642-04930-9\_4`.

4 Wouter Beek, Stefan Schlobach, and Frank van Harmelen. A contextualised semantics for owl:sameAs. In Harald Sack, Eva Blomqvist, Mathieu d'Aquin, Chiara Ghidini, Simone Paolo Ponzetto, and Christoph Lange, editors, *The Semantic Web. Latest Advances and New Domains - 13th International Conference, ESWC 2016, Heraklion, Crete, Greece, May 29 - June 2, 2016, Proceedings*, volume 9678 of *Lecture Notes in Computer Science*, pages 405–419. Springer, 2016.

5 Joachim Biskup, Piero A. Bonatti, Clemente Galdi, and Luigi Sauro. Inference-proof data filtering for a probabilistic setting. In *Proceedings of the 5th Workshop on Society, Privacy and the Semantic Web - Policy and Technology (PrivOn2017) co-located with 16th International Semantic Web Conference (ISWC 2017), Vienna, Austria, October 22, 2017.*, 2017. URL: http://ceur-ws.org/Vol-1951/PrivOn2017_paper_2.pdf.

6 Piero A. Bonatti, Iliana M. Petrova, and Luigi Sauro. Optimized construction of secure knowledge-base views. In *Proceedings of the 28th International Workshop on Description Logics, Athens,Greece, June 7-10, 2015.*, 2015. URL: http://ceur-ws.org/Vol-1350/paper-44.pdf.

7 Piero A. Bonatti and Luigi Sauro. A confidentiality model for ontologies. In *The Semantic Web - ISWC 2013 - 12th International Semantic Web Conference, Sydney, NSW, Australia, October 21-25, 2013, Proceedings, Part I*, pages 17–32, 2013. URL: https://doi.org/10.1007/978-3-642-41335-3_2, `doi:10.1007/978-3-642-41335-3\_2`.

8 Saša Buvač. Quantificational Logic of Context. In William J. Clancey and Daniel S. Weld, editors, *Proceedings of the Thirteenth National Conference on Artificial Intelligence and Eighth Innovative Applications of Artificial Intelligence Conference, AAAI 96, IAAI 96, Portland, Oregon, August 4-8, 1996, Volume 1*, pages 600–606. AAAI Press / The MIT Press, 1996.

9 Saša Buvač, Vanja Buvač, and Ian A. Mason. Metamathematics of Contexts. *Fundamenta Informaticae*, 23(2/3/4):263–301, 1995.

10 Gavin Carothers. RDF 1.1 N-Quads - A line-based syntax for RDF datasets, W3C Recommendation 25 February 2014. W3c recommendation, World Wide Web Consortium, February 25 2014. URL: http://www.w3.org/TR/2014/REC-n-quads-20140225/.

11 Willy Chen and Heiner Stuckenschmidt. A model-driven approach to enable access control for ontologies. In *Business Services: Konzepte, Technologien, Anwendungen. 9. Internationale Tagung Wirtschaftsinformatik 25.-27. Februar 2009, Wien*, pages 663–672, 2009. URL: http://aisel.aisnet.org/wi2009/65.

**12** Chris Clifton and Tamir Tassa. On syntactic anonymity and differential privacy. *Trans. Data Privacy*, 6(2):161–183, 2013. URL: http://www.tdp.cat/issues11/abs.a124a13.php.

**13** Claudia d'Amato, Steffen Staab, Andrea G. B. Tettamanzi, Tran Duc Minh, and Fabien L. Gandon. Ontology enrichment by discovering multi-relational association rules from ontological knowledge bases. In Sascha Ossowski, editor, *Proceedings of the 31st Annual ACM Symposium on Applied Computing, Pisa, Italy, April 4-8, 2016*, pages 333–338. ACM, 2016. URL: http://doi.acm.org/10.1145/2851613.2851842, doi:10.1145/2851613.2851842.

**14** Steven de Rooij, Wouter Beek, Peter Bloem, Frank van Harmelen, and Stefan Schlobach. Are names meaningful? quantifying social meaning on the Semantic Web. In Paul T. Groth, Elena Simperl, Alasdair J. G. Gray, Marta Sabou, Markus Krötzsch, Freddy Lécué, Fabian Flöck, and Yolanda Gil, editors, *The Semantic Web - ISWC 2016 - 15th International Semantic Web Conference, Kobe, Japan, October 17-21, 2016, Proceedings, Part I*, volume 9981 of *Lecture Notes in Computer Science*, pages 184–199, 2016.

**15** Lisa Ehrlinger and Wolfram Wöß. Towards a definition of Knowledge Graphs. In Michael Martin, Martí Cuquet, and Erwin Folmer, editors, *Joint Proceedings of the Posters and Demos Track of the 12th International Conference on Semantic Systems - SEMANTiCS2016 and the 1st International Workshop on Semantic Change & Evolving Semantics (SuCCESS'16) co-located with the 12th International Conference on Semantic Systems (SEMANTiCS 2016), Leipzig, Germany, September 12-15, 2016.*, volume 1695 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2016. URL: http://ceur-ws.org/Vol-1695/paper4.pdf.

**16** Eldora, Martin Knechtel, and Rafael Peñaloza. Correcting access restrictions to a consequence more flexibly. In *Proceedings of the 24th International Workshop on Description Logics (DL 2011), Barcelona, Spain, July 13-16, 2011*, 2011. URL: http://ceur-ws.org/Vol-745/paper_9.pdf.

**17** Kemele M. Endris, Zuhair Almhithawi, Ioanna Lytra, Maria-Esther Vidal, and Sören Auer. BOUNCER: privacy-aware query processing over federations of RDF datasets. In *Database and Expert Systems Applications - 29th International Conference, DEXA 2018, Regensburg, Germany, September 3-6, 2018, Proceedings, Part I*, pages 69–84, 2018. URL: https://doi.org/10.1007/978-3-319-98809-2_5, doi:10.1007/978-3-319-98809-2_5.

**18** Johannes Frey, Kai Müller, Sebastian Hellmann, Erhard Rahm, and Maria-Esther Vidal. Evaluation of Metadata Representations in RDF stores. *Semantic Web Journal*. To appear.

**19** Luis Galárraga, Christina Teflioudi, Katja Hose, and Fabian M. Suchanek. Fast rule mining in ontological knowledge bases with AMIE+. *VLDB J.*, 24(6):707–730, 2015. URL: https://doi.org/10.1007/s00778-015-0394-1, doi:10.1007/s00778-015-0394-1.

**20** Chiara Ghidini and Fausto Giunchiglia. Local Models Semantics, or contextual reasoning=Locality+Compatibility. *Artificial Intelligence*, 127(2):221–259, 2001.

**21** José Miguel Giménez-García, Antoine Zimmermann, and Pierre Maret. NdFluents: An Ontology for Annotated Statements with Inference Preservation. In Eva Blomqvist, Diana Maynard, Aldo Gangemi, Rinke Hoekstra, Pascal Hitzler, and Olaf Hartig, editors, *The Semantic Web - 14th International Conference, ESWC 2017, Portorož, Slovenia, May 28 - June 1, 2017, Proceedings, Part I*, volume 10249 of *Lecture Notes in Computer Science*, pages 638–654. Springer, May 2017.

**22** Fausto Giunchiglia. Contextual Reasoning. *Epistemologia*, 16:345–364, 1993.

**23** Guido Governatori, Ho-Pun Lam, Antonino Rotolo, Serena Villata, Ghislain Auguste Atemezing, and Fabien L Gandon. LIVE: a tool for checking licenses compatibility between vocabularies and data. In *International Semantic Web Conference*, 2014.

**24** Ramanathan V. Guha. *Contexts: a Formalization and Some Applications*. PhD thesis, Stanford University, Stanford, CA (USA), 1991. Revised version at http://www-formal.stanford.edu/guha/guha-thesis.ps.

**25** Ramanathan V. Guha, Rob McCool, and Richard Fikes. Contexts for the Semantic Web. In Frank van Harmelen, Sheila McIlraith, and Dimitri Plexousakis, editors, *The Semantic Web - ISWC 2004: Third International Semantic Web Conference,Hiroshima, Japan, November 7-11, 2004. Proceedings*, volume 3298 of *Lecture Notes in Computer Science*, pages 32–46. Springer, 2004.

**26** Governatori Guido, Lam Ho-Pun, Rotolo Antonino, Villata Serena, and Gandon Fabien. Heuristics for Licenses Composition. *Frontiers in Artificial Intelligence and Applications*, 2013.

**27** Al Koudous Idrissou, Rinke Hoekstra, Frank van Harmelen, Ali Khalili, and Peter Van den Besselaar. Is my:sameAs the same as your:sameAs?: Lenticular lenses for context-specific identity. In Óscar Corcho, Krzysztof Janowicz, Giuseppe Rizzo, Ilaria Tiddi, and Daniel Garijo, editors, *Proceedings of the Knowledge Capture Conference, K-CAP 2017, Austin, TX, USA, December 4-6, 2017*, pages 23:1–23:8. ACM, 2017.

**28** Jeremy J. Carroll, Christian Bizer, Pat Hayes, and Patrick Stickler. A survey of trust in computer science and the semantic web. *Journal of Web Semantics*, 5(2):58–71, 2007.

**29** Günter Karjoth, Matthias Schunter, and Michael Waidner. Platform for enterprise privacy practices: Privacy-enabled management of customer data. In *International Workshop on Privacy Enhancing Technologies*, pages 69–84. Springer, 2002.

**30** Daniel Kifer and Ashwin Machanavajjhala. No free lunch in data privacy. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2011, Athens, Greece, June 12-16, 2011*, pages 193–204, 2011. URL: http://doi.acm.org/10.1145/1989323.1989345, `doi:10.1145/1989323.1989345`.

**31** Michael Kifer and V. S. Subrahmanian. Theory of generalized annotated logic programming and its applications. *Journal of Logic Programming*, 12(3&4):335–367, 1992.

**32** Sabrina Kirrane, Alessandra Mileo, and Stefan Decker. Access control and the resource description framework: A survey. *Semantic Web*, 2017. URL: http://www.semantic-web-journal.net/system/files/swj1280.pdf.

**33** Martin Knechtel and Heiner Stuckenschmidt. Query-based access control for ontologies. In *Web Reasoning and Rule Systems - Fourth International Conference, RR 2010, Bressanone/Brixen, Italy, September 22-24, 2010. Proceedings*, pages 73–87, 2010. URL: https://doi.org/10.1007/978-3-642-15918-3_7, `doi:10.1007/978-3-642-15918-3\_7`.

**34** Markus Krötzsch, Maximilian Marx, Ana Ozaki, and Veronika Thost. Attributed Description Logics: Reasoning on Knowledge Graphs. In Jérôme Lang, editor, *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 5309–5313. ijcai.org, July 2018.

MC This and the following reference are the only ones in this section which have their names abbreviated.

**35** T. K. Landauer and S. T. Dumais. How come you know so much? from practical problem to theory. In D. Hermann, C. McEvoy, M. Johnson, and P. Hertel, editors, *Basic and applied memory: Memory in context.*, pages 105–126. Mahwah, NJ: Erlbaum, 1996.

**36** T. K. Landauer and S. T. Dumais. A solution to plato's problem: The latent semantic analysis theory of the acquisition, induction, and representation of knowledge. *Psychological Review*, 104:211–240, 1997.

**37** Agnieszka Lawrynowicz. *Semantic Data Mining - An Ontology-Based Approach*, volume 29 of *Studies on the Semantic Web*. IOS Press, 2017. URL: https://doi.org/10.3233/978-1-61499-746-7-i, `doi:10.3233/978-1-61499-746-7-i`.

**38** Changchang Liu, Supriyo Chakraborty, and Prateek Mittal. Dependence makes you vulnerable: Differential privacy under dependent tuples. In *23rd Annual Network and Distributed System Security Symposium, NDSS 2016, San Diego, California, USA, February 21-24,* *2016*, 2016. URL: http://wp.internetsociety.org/ndss/wp-content/uploads/sites/25/2017/09/dependence-makes-you-vulnerable-differential-privacy-under-dependent-tuples.pdf.

39 John McCarthy. Notes on Formalizing Context. In Ruzena Bajcsy, editor, *Proceedings of the 13th International Joint Conference on Artificial Intelligence. Chambéry, France, August 28 - September 3, 1993*, pages 555–562. Morgan Kaufmann, 1991.

40 Marco Casassa Mont, Siani Pearson, and Pete Bramhall. Towards accountable management of identity and privacy: Sticky policies and enforceable tracing services. In *Database and Expert Systems Applications, 2003. Proceedings. 14th International Workshop on*, pages 377–382. IEEE, 2003.

41 Rolf Nossum. A decidable multi-modal logic of context. *Journal of Applied Logic*, 1(1–2):119–133, 2003.

42 Siani Pearson and Marco Casassa Mont. Sticky policies: an approach for privacy management across multiple parties. *IEEE Computer*, 44(9):60–68, 2011.

43 Joe Raad, Wouter Beek, Frank van Harmelen, Nathalie Pernelle, and Fatiha Saïs. Detecting erroneous identity links on the Web using network metrics. In Denny Vrandecic, Kalina Bontcheva, Mari Carmen Suárez-Figueroa, Valentina Presutti, Irene Celino, Marta Sabou, Lucie-Aimée Kaffee, and Elena Simperl, editors, *The Semantic Web - ISWC 2018 - 17th International Semantic Web Conference, Monterey, CA, USA, October 8-12, 2018, Proceedings, Part I*, volume 11136 of *Lecture Notes in Computer Science*, pages 391–407. Springer, 2018.

44 Md. Kamruzzaman Sarker, Adila Alfa Krisnadhi, and Pascal Hitzler. OWLAx: A Protégé plugin to support ontology axiomatization through diagramming. In Takahiro Kawamura and Heiko Paulheim, editors, *Proceedings of the ISWC 2016 Posters & Demonstrations Track co-located with 15th International Semantic Web Conference (ISWC 2016), Kobe, Japan, October 19, 2016.*, volume 1690 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2016.

45 Jia Tao, Giora Slutzki, and Vasant G. Honavar. Secrecy-preserving query answering for instance checking in *EL*\mathcal{EL}. In *Web Reasoning and Rule Systems - Fourth International Conference, RR 2010, Bressanone/Brixen, Italy, September 22-24, 2010. Proceedings*, pages 195–203, 2010. URL: https://doi.org/10.1007/978-3-642-15918-3_16, doi:10.1007/978-3-642-15918-3_16.

46 Octavian Udrea, Diego Reforgiato Recupero, and V. S. Subrahmanian. Annotated RDF. *ACM Transaction on Computational Logics*, 11(2), 2010.

47 Serena Villata and Fabien Gandon. Licenses compatibility and composition in the web of data. In *Third International Workshop on Consuming Linked Data (COLD2012)*, 2012.

48 Xiaokui Xiao, Yufei Tao, and Nick Koudas. Transparent anonymization: Thwarting adversaries who know the algorithm. *ACM Trans. Database Syst.*, 35(2):8:1–8:48, 2010. URL: http://doi.acm.org/10.1145/1735886.1735887, doi:10.1145/1735886.1735887.

49 Antoine Zimmermann. RDF 1.1: On Semantics of RDF Datasets, W3C Working Group Note 25 February 2014. W3c working group note, World Wide Web Consortium, February 25 2014. URL: http://www.w3.org/TR/2014/NOTE-rdf11-datasets-20140225/.

50 Antoine Zimmermann and José M. Giménez-García. Contextualizing DL Axioms: Formalization, a New Approach, and Its Properties. In Daniele Dell'Aglio, Darko Anicic, Payam M. Barnaghi, Emanuele Della Valle, Deborah L. McGuinness, Loris Bozzato, Thomas Eiter, Martin Homola, and Daniele Porello, editors, *Joint Proceedings of the Web Stream Processing Workshop (WSP 2017) and the 2nd International Workshop on Ontology Modularity, Contextuality, and Evolution (WOMoCoE 2017) Co-Located with 16th International Semantic Web Conference (ISWC 2017), Vienna, Austria, October 22nd, 2017*, volume 1936 of *CEUR Workshop Proceedings*, pages 74–85. Sun SITE Central Europe (CEUR), October 2017.

51 Antoine Zimmermann, Nuno Lopes, Axel Polleres, and Umberto Straccia. A general framework for representing, reasoning and querying with annotated Semantic Web data. *Journal of Web Semantics*, 11:72–95, 2012.

# 12 Natural Language Processing and Knowledge Graphs

*Paul Groth (University of Amsterdam - Amsterdam, NL & Elsevier Labs - Amsterdam, NL), Roberto Navigli (Sapienza University of Rome, IT), Andrea Giovanni Nuzzolese (CNR - Rome, IT), Marieke van Erp (KNAW Humanities Cluster - Amsterdam, NL), and Gerard de Melo (Rutgers University - Piscataway, US)*

## 12.1 Introduction

The Natural Language Processing (NLP) community has focused on machine learning and data-driven approaches to linguistic problems for several decades now, leading to the recent proposal of implicit, latent models and representations obtained from large amounts of training data [9, 17]. In contrast, the Semantic Technology community has primarily taken a symbolic approach resulting in the production of explicit, human-readable knowledge representations [7, 19]. However, both communities share core goals of Artificial Intelligence such as making applications more intelligent and interactive, and, in the longer term, enabling machine understanding. Knowledge Graphs provide a key contribution to bridging the gap between the two areas and join forces to achieve their common mission.

## 12.2 Challenges in NLP

A central goal in NLP is to study the design and implementation of abstractive and comprehensive representations of the knowledge captured in natural language. In many cases, this can be achieved by means of knowledge graphs. The problem can be viewed from two perspectives:

- **Capturing the richness of text as a knowledge graph:** this perspective is characterized by challenges such as extracting and representing:
  - quantifiers (e.g., "in this country a woman gives birth *every* 15 minutes");
  - modality (e.g., “it *can* be a good opportunity");
  - negation and logical structures (e.g., “this solution is *not* good");
  - temporal aspects, based on context and tenses (e.g., “Barack Obama served as US President *from 2009 to 2017*") or based on historical context (e.g., the evolution of cultural heritage goods);
  - pragmatics, including coreference and anaphora resolution, common-sense reasoning and irony detection (e.g., referring to the above sentence, “our job is to find that woman and stop her").
- **Enhancing NLP techniques with knowledge graphs:** the key idea here is to leverage graph-structured knowledge to improve tasks that are typically solved via mainstream supervised techniques, such as deep learning. This perspective would benefit several NLP tasks including:
  - Word Sense Disambiguation (WSD), where the use of symbolic, structured knowledge has already been shown to improve the performance [14, 11, 1];

- Named Entity Linking, a task analogous to WSD where, instead of associating word senses, we associate named entities with mentions occurring in context;
- Semantic parsing, where the need for structured knowledge is intrinsic to the task and a graph representing a sentence or a larger text is produced by the system;
- Cultural-centric sentiment analysis, where the sentiment associated with certain situations might vary considerably on a cultural basis, therefore requiring encoding cultural-specific knowledge in structured form.

## 12.3 Existing Approaches

There is a large body of work that touches on these themes, including primers on knowledge graph construction from text[11][12][13] and work from the emerging community of automated knowledge base construction[14]. Likewise, the recent attempts at providing universal structured representations of text such as Abstract Meaning Representation[15], UCCA[16] and Universal Dependencies[17] are of interest, although their representation of world knowledge is less rich in comparison with knowledge graphs. Additionally, the state of the art in Word Sense Disambiguation as well as entity linking and distant supervision often leverages knowledge graphs [12, 8].

To tackle the well-known issue of the knowledge acquisition bottleneck which affects supervised lexical-semantic disambiguation techniques, recent approaches, like Train-O-Matic [16], use lexicalized knowledge graphs such as BabelNet [15] to create large training data and scale to arbitrary languages. Relevant research has also been carried out trying to bridge NLP with the world of knowledge graphs. For example, it is worth mentioning the paradigm of machine reading [4], i.e., systems able to transform natural language text to formal structured knowledge such that the latter can be interpreted by machines, according to a shared semantics. The NELL knowledge base [10], for instance, is built from triples extracted from the Web. Tools such as FRED [6] and PIKES [2] aim at machine reading beyond the level of subject-predicate-object triples. But while linguistic resources are brought together in the Linguistic Linked Open Data Cloud, integrating these sources in (statistical and neural) NLP tools is still an open issue. Relevant examples of datasets part of the Linguistic Linked Open Data initiative are BabelNet [15], Framester [5], Lexvo.org [3], and FrameBase [18].

## 12.4 Opportunities

The following opportunities are open at the intersection between knowledge graphs and NLP:

- **Mutual exchange between knowledge graphs and NLP:** Explicit knowledge may help filter out incorrect named entity linking candidates based on temporal constraints (a car model cannot be involved in an accident before it is produced). Accordingly:

[11] https://kgtutorial.github.io
[12] https://kdd2018tutorialt39.azurewebsites.net
[13] http://usc-isi-i2.github.io/AAAI18Tutorial/
[14] http://www.akbc.ws/
[15] https://amr.isi.edu
[16] http://www.cs.huji.ac.il/~oabend/ucca.html
[17] http://universaldependencies.org

  - **Is a knowledge graph expressive enough for NLP?** An opportunity is to investigate new formalisms or theories for enabling knowledge graphs to represent the richness of natural language;
  - **Can lexicalized knowledge graphs improve NLP?** Multilingual knowledge graphs on Web scale can be used as background knowledge for addressing NLP tasks more effectively. For example, NLP tasks that needs to be context-aware or require commons sense might benefit from lexical knowledge graphs.
- **Representation issues:** for instance, the semantics of an apparently unambiguous word like copyright has distinct culturally-specific meanings in different countries; some words, such as *ikigai* or *gezellig*, cannot be expressed in other languages and alternative (typically more general) meanings have to be provided.
- **How to address cultural specificity?** Applications can be better tailored to users needs, and barriers in cross-cultural communication can be overcome. This, together with the above point, are important research opportunities to tailor solutions to the culture which speaks (or translates from) a certain language.
- **Is there any effective and usable formal semantics** that, from an NLP perspective, can be consistently adopted to capture the meaning of language (independently of which language is used)? For instance, work in the field of semantic parsing is still struggling for the right type of structured representation [13].

## 12.5 Conclusions

The time is ripe for NLP and knowledge graphs to get together. Several opportunities are open which can provide the two areas with mutual benefits and clear performance improvements, on one hand, in the type and quality of the represented knowledge, and, on the other hand, on the use of general knowledge for improving text understanding.

### References

1 Eneko Agirre, Oier Lopez de Lacalle, and Aitor Soroa. Random walks for knowledge-based word sense disambiguation. *Computational Linguistics*, 40(1):57–84, 2014.

2 Francesco Corcoglioniti, Marco Rospocher, and Alessio Palmero Aprosio. Frame-based ontology population with pikes. *IEEE Trans. on Knowl. and Data Eng.*, 28(12):3261–3275, December 2016.

3 Gerard de Melo. Lexvo.org: Language-related information for the Linguistic Linked Data cloud. *Semantic Web*, 6(4):393–400, August 2015.

4 Oren Etzioni, Michele Banko, and Michael J. Cafarella. Machine reading. In *Machine Reading, Papers from the 2007 AAAI Spring Symposium, Technical Report SS-07-06, Stanford, California, USA, March 26-28, 2007*, pages 1–5, 2007.

5 Aldo Gangemi, Mehwish Alam, Luigi Asprino, Valentina Presutti, and Diego Reforgiato Recupero. Framester: A wide coverage linguistic linked data hub. In *Knowledge Engineering and Knowledge Management - 20th International Conference, EKAW 2016, Bologna, Italy, November 19-23, 2016, Proceedings*, pages 239–254, 2016.

6 Aldo Gangemi, Valentina Presutti, Diego Reforgiato Recupero, Andrea Giovanni Nuzzolese, Francesco Draicchio, and Misael Mongiovì. Semantic Web Machine Reading with FRED. *Semantic Web*, 8(6):873–893, 2017.

7 Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, and Christian Bizer. Dbpedia - A large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195, 2015.

8 Diana Maynard, Kalina Bontcheva, and Isabelle Augenstein. *Natural Language Processing for the Semantic Web*. Morgan & Claypool Publishers, 2016.

9 Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013.

10 Tom M. Mitchell, William W. Cohen, Estevam R. Hruschka Jr., Partha Pratim Talukdar, Justin Betteridge, Andrew Carlson, Bhavana Dalvi Mishra, Matthew Gardner, Bryan Kisiel, Jayant Krishnamurthy, Ni Lao, Kathryn Mazaitis, Thahir Mohamed, Ndapandula Nakashole, Emmanouil A. Platanios, Alan Ritter, Mehdi Samadi, Burr Settles, Richard C. Wang, Derry Wijaya, Abhinav Gupta, Xinlei Chen, Abulhair Saparov, Malcolm Greaves, and Joel Welling. Never-ending learning. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA.*, pages 2302–2310, 2015.

11 Andrea Moro, Alessandro Raganato, and Roberto Navigli. Entity linking meets word sense disambiguation: a unified approach. *TACL*, 2:231–244, 2014.

12 Roberto Navigli. Word sense disambiguation: A survey. *ACM Comput. Surv.*, 41(2):10:1–10:69, 2009.

13 Roberto Navigli. Natural language understanding: Instructions for (present and future) use. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden.*, pages 5697–5702, 2018.

14 Roberto Navigli and Mirella Lapata. An experimental study of graph connectivity for unsupervised word sense disambiguation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(4):678–692, 2010.

15 Roberto Navigli and Simone Paolo Ponzetto. Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artif. Intell.*, 193:217–250, 2012.

16 Tommaso Pasini and Roberto Navigli. Train-o-matic: Large-scale supervised word sense disambiguation in multiple languages without manual training data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 78–88, 2017.

17 Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar*, pages 1532–1543, 2014.

18 Jacobo Rouces, Gerard de Melo, and Katja Hose. Framebase: Enabling integration of heterogeneous knowledge. *Semantic Web*, 8(6):817–850, 2017.

19 Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. YAGO: A large ontology from wikipedia and wordnet. *J. Web Sem.*, 6(3):203–217, 2008.

# 13 Machine Learning and Knowledge Graphs

*Steffen Staab (Universität Koblenz-Landau, DE), Gerard de Melo (Rutgers University - Piscataway, US), Michael Witbrock (IBM Research - Yorktown Heights, US), Volker Tresp (Siemens AG - München, DE), Claudio Gutierrez (University of Chile - Santiago de Chile, CL), Dezhao Song (Thomson Reuters - Eagan, US), and Axel-Cyrille Ngonga-Ngomo (Universität Paderborn, DE)*

Machine learning with deep networks, and Knowledge Representation with knowledge graphs are both advancing rapidly, both in depth and scale. Each has distinct advantages: symbolic (knowledge graph) representation and inference brings high reliability, explainability and reusability; machine learning brings the ability to learn from very weak signals (whole image labels, or reinforcement signals), and to learn to do tasks that humans cannot program. These sources of power are largely orthogonal, but applicable to similar problems and domains. Although their advances have been proceeding largely independently, and sometimes in ignorance of the other, there are early indications that they can be combined effectively. We believe that the potential value of this combination warrants immediate joint action.

To draw a roadmap for such action, the section is structured as follows:

1. position knowledge graphs in relation to the most closely related machine learning paradigms (see Sect. 1)
2. identify some major opportunities that knowledge graphs may offer to machine learning to improve overall learning and inference (see Sect. 3).
3. analyse commonalities and differences that machine learning systems and knowledge graph systems offer with regard to manipulation of knowledge (see sect. 2).
4. provide a brief survey of methods and representations that benefit from knowledge graphs already today (see Sect. 4).
5. sketch future challenges for an integration of the two disciplines.
6. conclude with a call for action.

## 13.1 Positioning Knowledge Graphs with respect to Machine Learning Paradigms

Traditional machine learning algorithms operate over feature vectors representing objects in terms of numeric or categorical attributes. The main learning task is to learn a mapping from such feature vectors to an output prediction in some form. This could be class labels, a regression score, an unsupervised cluster ID or a latent vector (embedding). In knowledge induction generally, the representation of an object, event or type can contain explicit representations of the properties and simple or complex relationships with other objects, types or events. In knowledge graph learning more specifically, the representation of an object can include representations of its direct relationships to other individual objects, thus, the data is in the form of a graph, consisting of nodes (entities) and labelled edges (relationships between entities).

There are three main paradigms (for a survey consult [13] and [17]) that can be used for performing the aforementioned learning task.

- In Inductive Logic Programming, the learned machine learning models make deterministic or close to deterministic based predictions, formulated in some logic language. ILP-type approaches also permit the integration of ontological background knowledge. In ILP, the main object of research is not making the predictions per se, but techniques for automatically acquiring components of the theory that enables them, in a logical language,
- In Statistical Relational Learning approaches, learned machine learning models make probabilistic statements formulated either as a conditional probability (as in Probabilistic Relational Models) or as a potential function (as in Markov Logic Programming).
- Currently most popular are models using latent representations. Popular examples are RESCAL, TransE, HolE and ComplEx. More recent developments are Graph Convolutional approaches.

When considering KGs, most often the learning problem is formulated as a link prediction problem ("Is component A suitable for problem B?") or a class (type) prediction problem ("Is this a promising customer?"). Recently, KGs have been used also in transfer learning approaches, where the latent representations are shared between knowledge graph models, and the representations are used in a particular application. This has been the basis, for instance, for a medical decision support system. Particularly interesting is the pairing of knowledge graph-based learning with the use of unstructured data. For instance, this research direction has been used to better understand texts and images, and, conversely, also for knowledge graph completion.

## 13.2 Managing and Manipulating Knowledge: Comparing Machine Learning and Knowledge Graphs

The symbolic representations of knowledge encountered in knowledge graphs have a number of distinct characteristics in comparison with the kinds of latent representations at the subsymbolic level that can be induced using machine learning.

Latent representations excel at *capturing knowledge that is not crisp*, e.g., statistical regularities and similarities [3, 4]. A knowledge graph, in contrast, captures discrete facts and it is non-trivial – though still possible [5, 22] – to quantify the strength of association between arbitrary items.

An important advantage of latent representations is their ability to *generalize* beyond what is known explicitly [3]. For instance, while a knowledge graph might store genre information for several thousands of movies, latent representations may enable us to infer the genre of additional movies based on various informative cues. However, latent representations do not straightforwardly allow us to keep track of exceptions to typical patterns. They may learn the typical attributes of 14-year old Canadian teens, but would not be able to keep track of the fact that Computer scientist Erik Demaine completed his Bachelor's degree at the age of 14 and was awarded his PhD degree at the University of Waterloo at the age of 20.

More generally, latent representations typically fail to *record precise identities*. Dense vector representations do not normally accurately keep track of who is married to whom. Services such as Google or Siri would not try to rely on latent representations to deliver answers to queries such as "Where was Einstein born?' [10]'. Depending on the application, it may be important to distinguish precisely whether someone won the Nobel Prize for Literature or rather a similar – but distinct – literature award.

This affects the *interpretability* of the knowledge and the *explainability* and thus also *trustworthiness* of results derived from such knowledge. Latent representations consist of series of numbers without any obvious human interpretation. The knowledge in a knowledge graph, in contrast, can straightforwardly be inspected.

The inability to record information precisely further also affects the *updatability* of the stored knowledge. While one can easily add a new fact to a knowledge graph, updating latent representations is often non-trivial and even if it is possible, the newly added information may still fail to be reflected the predictions made from the representations. The same applies when removing facts. One can easily remove a fact from a knowledge graph, but updating a machine learning model to capture such a change is challenging.

This suggests that representation learning alone cannot exhaustively address all knowledge needs of modern AI-driven applications. The machine learning components in conversational agents, Web search, and intelligent decision support systems will have to draw on large repositories of knowledge to obtain the desired results.

## 13.3 Knowledge Graph Assets for Machine Learning: Grand Opportunities

Most of human learning is intrinsically linked to knowledge that the individual possesses. Thus, we believe that bringing knowledge graphs to machine learning will systematically improve the accuracy and extend the range of machine learning capabilities. In particular, we see the following four grand opportunities:

1. **Data efficiency:** When data is sparse, knowledge structures may help to fill the gap. Knowledge graph abstractions may be used for coping with data sparsity by generating additional training data, e.g. negative data that would lead to inconsistent interpretations, or by giving indications of how to aggregate uncertain predictions over categories related or similar in the knowledge graph. Thus, knowledge graphs may improve data efficiency.
2. **Zero-shot learning** describe the challenge to cope with previously unencountered types of situations. The combination of induction from machine learning and deduction from knowledge graphs provides for the opportunity to deal, e.g., with pictures where the type of situation did not appear in the training data. One example might be a group of dark wild boars roaming a street at night where an autonomous car is driving and thus encountering a situation that might not have been in the training data at all.
3. **Consistent and coherent structured predictions:** The more complex the actual prediction the less accurate it tends to be. Structured predictions may aim at predicting a whole course of events (e.g. how a video will continue, how a patient's illness will develop), but not all of these predictions may exhibit the same level of consistency. Knowledge graphs let us check which predictions might be consistent/inconsistent or coherent/incoherent with our available knowledge. For instance, the most likely classification of traffic signs – even under adversarial attacks – might be re-assessed with regard to a vehicle's knowledge graph.
4. **Succinct explanations:** Several difficulties arise when explaining predictions made by machine learning systems. One issue is the implicit representations causing the predictions (e.g. neural curve fitting). A second problem is the possible low level of explanation a system might give, even when it works on explicit representations. A third issue may result from the sheer volume of explanation produced, and, fourth, an actual user might only be interested in a specific detail of the overall explanation. Knowledge graphs may alleviate all these four issues by (i) mapping the explanation to (ii) an appropriate level

of generalization, (iii) summarizing the found phenomenon and (iv) comparing this to a state of knowledge that the user might already have.

These are a few opportunities arising from the usage of knowledge graphs in machine learning, which we consider could lead to step changes. More will likely be developed and many gradual improvements may result from progressing machine learning to learn not only from tables, but also from knowledge graphs of various levels of expressiveness and size.

Vice versa, as reported also in other sections of this report, creating and maintaining knowledge tasks constitutes a major effort, which benefits already know benefits a lot from machine learning [7] – and probably even more so in the future.

## 13.4 Representations and Methods

Some prominent cases

1. Language/Images:
   - Question answering

     Question answering is one important application of knowledge graphs. KGs contain a wealth of information and question answering could be a good way to help end users to more effectively and also more efficiently retrieve information from the KGs. Various approaches have been developed in this area and they are targeting different types of knowledge bases. A number of challenges pertaining to using knowledge graphs for question answering are presented in [9].

     Berant et al. [1] developed the SEMPRE system that translates/parses a natural language question into a logic form. The approach takes into account both free text and a knowledge base (more specifically, freebase) when translating the questions.

     TR Discover [20] takes advantage of a feature-based context free grammar in order to translate natural language questions into a first-order logic representation. The logic representation serves as an intermediate representation and it is further translated to executable queries in different query languages, including Cypher (Neo4J), SPARQL and SQL. By further adopting a deep learning-based approach, the system tags the tokens in a natural language question in order to reduce its dependence on the grammar [19].

     Deep learning-based approaches have been actively developed for question answering, not only for RDF-based knowledge bases but also other types of databases. decaNLP [11] is a recent system that provides question answering capability against relational databases. It provides the general concept that a natural language question can be represented by the question itself and its context (e.g., PoS tagging, semantic role labeling result and relation extraction result). Instead of learning on a single task, it tries to perform multi-task learning on various tasks and observed better performances on some of the tasks.

     In addition to adopting fully automatic query parsing, CrowdQ [6] also incorpor-

ates crowd-sourcing techniques for understanding natural language questions.
- Machine translation. [12]
- Natural language generation (NLG) for structured data.

  The NLG community has an increasing interest in the generation of natural language from knowledge graphs, especially as this interface promises to generate the interface between the data-centric world in which knowledge graphs dwell and humans. Challenges such as E2E[18],[19] [14] and the WebNLG challenge [8] provide first indications as to KGs a potential lingua franca for human-machine interaction.

  Early approaches to NLG are mostly template-based, which prevents them from being easily adapted to other domains [16]. There are also approaches that compare and contrast deep learning-based and template-based approaches [15, 2].
- Caption generation for images
- Video prediction

2. Medical / Drug:
   - Drug activity prediction
   - Transparency in the knowledge, behaviour and assumption change in the clinical decision process.
   - The Clinical Data Intelligence Project. The notion of "clinical data intelligence" refers to the integration of medical data from heterogeneous resources, the combination of the extracted information, and the generation of medical clinically-relevant knowledge about patients or treatments. [21]. Again here the semantic integration, that is, the codification of data, metadata and relationships with other sources and user description together in one standard format, allows us to uniformly apply machine learning techniques to this now unified knowledge.
   - Blue Brain Nexus[20] is a data repository and metadata catalogue organizing (agnostic of the domain) that treats provenance as a first-class citizen, thus facilitating the tracking of the origin of data and how it is being used, thus allowing to assess data quality.
3. Link Recommendation
4. Robotics. RoboBrain (2014): "Building such an engine brings with it the challenge of dealing with multiple data modalities including symbols, natural language, haptic senses, robot trajectories, visual features and many others. The knowledge stored in the engine comes from multiple sources including physical interactions that robots have while performing tasks (perception, planning and control), knowledge bases from the Internet and learned representations from several robotics research groups." [18].
   Integrating these data sources to build a dataset requires importing and documenting in the data the sources, types, interrelations, etc., that is, building a knowledge graph.
5. Deep Learning
   - Knowledge Graphs supporting DL for prediction and decision support
   - Transfer Learning: from KGs to DL
   - Knowledge Graph supported DL Perception Systems

[18] http://www.macs.hw.ac.uk/InteractionLab/E2E/
[19] https://inlg2018.uvt.nl/special-session-generation-challenges/
[20] https://bluebrain.epfl.ch/page-153280-en.html

- Autonomous training of new DL models using data stored in KG, based on problem description (completely automate training of a DL classifier or transformer by storing the data in a KG and identifying the relevant training examples using inference).
- Similarly, supporting completely automated Data Science
- Demonstrate storing outputs from a DL system in a KG in a form that allows another DL system to perform better on a new task
  - E.g. translation system learns a new language with few examples using grammatical or semantic knowledge acquired elsewhere
  - E.g. same KG improves caption generation

## 13.5 Novel Representations and Paradigms

- Knowledge about inference; rules, higher order, Meta knowledge, problem-solving knowledge
- Learning to do inference or do it better
- Interpretation

## 13.6 Conclusions and Calls to Action

Conclude by comparing the two:

| | Explicit KG representations | Implicit KG representations |
|---|---|---|
| Retrieving facts | Trivial | Noisy |
| Adding facts | Trivial | Challenging |
| Removing facts | Trivial | Challenging |
| Generalizing facts | Requires additional machine learning | Straightforward |
| Computing similarities | Requires graph algorithms / weights | Straightforward |
| Interpretability | Trivial | Very limited |

Call for action for academia (machine learning, data base and knowledge graph communities):

- Treat knowledge/metadata/provenance as first order citizens

Call for action for funding agencies

- China does it! Japan does it?

Call-for-action for knowledge graph users (industry)

- Build and publish knowledge graphs in various domains for machine learnig
- Example proponents: Thomson Reuters (https://permid.org/), Blue Brain Nexus, Amazon Product Knowledge Graphs and Alexa AI Knowledge Graph

### References

1 Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1533–1544. ACL, 2013.

2 Charese Smiley, Elnaz Davoodi, Dezhao Song, Frank Schilder. The E2E NLG Challenge: A Tale of Two Systems. In *INLG 2018*, pages 472-477, 2018.

3 Jiaqiang Chen, Niket Tandon, Charles Darwis Hariman, and Gerard de Melo. WebBrain: Joint neural learning of large-scale commonsense knowledge. In *Proceedings of ISWC 2016*, pages 102–118. Semantic Web Science Association, 2016.

4 Gerard de Melo. Inducing conceptual embedding spaces from Wikipedia. In *Proceedings of WWW 2017*. ACM, 2017.

5 Gerard de Melo and Gerhard Weikum. On the utility of automatically generated wordnets. In *Proceedings of the Fourth Global WordNet Conference (GWC 2008)*, pages 147–161. University of Szeged, December 2007.

6 Gianluca Demartini, Beth Trushkowsky, Tim Kraska, and Michael J. Franklin. Crowdq: Crowdsourced query understanding. In *CIDR 2013, Sixth Biennial Conference on Innovative Data Systems Research, Asilomar, CA, USA, January 6-9, 2013, Online Proceedings*. www.cidrdb.org, 2013.

7 Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. Knowledge vault: a web-scale approach to probabilistic knowledge fusion. In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pages 601–610, 2014.

8 Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. Creating training corpora for micro-planners. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vancouver, Canada, August 2017. Association for Computational Linguistics.

9 Konrad Höffner, Sebastian Walter, Edgard Marx, Ricardo Usbeck, Jens Lehmann, and Axel-Cyrille Ngonga Ngomo. Survey on challenges of question answering in the semantic web. *Semantic Web*, 8(6):895–920, 2017.

10 Huadong Li, Yafang Wang, Gerard de Melo, Changhe Tu, and Baoquan Chen. Multimodal question answering over structured data with ambiguous entities. In *Proceedings of WWW 2017 (Cognitive Computing Track)*. ACM, 2017.

11 Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730*, 2018.

12 Diego Moussallem, Matthias Wauer, and Axel-Cyrille Ngonga Ngomo. Machine translation using semanticweb technologies: A survey. *Web Semantics: Science, Services and Agents on the World Wide Web*, 0(0), 2018.

13 Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104(1):11–33, 2016.

14 Jekaterina Novikova, Ondrej Dušek, and Verena Rieser. The E2E dataset: New challenges for end-to-end generation. In *Proceedings of the 18th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, Saarbrücken, Germany, 2017. arXiv:1706.09254.

15 Yevgeniy Puzikov and Iryna Gurevych. E2e nlg challenge: Neural models vs. templates. In *E2E NLG Challenge*, 2017.

16 Ehud Reiter and Robert Dale. Building applied natural language generation systems. *Natural Language Engineering*, 3(1):57–87, 1997.

17 Achim Rettinger, Uta Lösch, Volker Tresp, Claudia d'Amato, and Nicola Fanizzi. Mining the semantic web. *Data Mining and Knowledge Discovery*, 24(3):613–662, 2012.

18 Ashutosh Saxena, Ashesh Jain, Ozan Sener, Aditya Jami, Dipendra Kumar Misra, and Hema Swetha Koppula. Robobrain: Large-scale knowledge engine for robots. *CoRR*, abs/1412.0691, 2014.

19 Dezhao Song and Frank Schilder. Systems and methods for automatic semantic token tagging, August 9 2018. US Patent App. 15/889,947.

20 Dezhao Song, Frank Schilder, Charese Smiley, Chris Brew, Tom Zielund, Hiroko Bretz, Robert Martin, Chris Dale, John Duprey, Tim Miller, and Johanna Harrison. TR discover: A natural language interface for querying and analyzing interlinked datasets. In Marcelo Arenas, Óscar Corcho, Elena Simperl, Markus Strohmaier, Mathieu d'Aquin, Kavitha Srinivas, Paul T. Groth, Michel Dumontier, Jeff Heflin, Krishnaprasad Thirunarayan, and Steffen Staab, editors, *The Semantic Web - ISWC 2015 - 14th International Semantic Web Conference, Bethlehem, PA, USA, October 11-15, 2015, Proceedings, Part II*, volume 9367 of *Lecture Notes in Computer Science*, pages 21–37. Springer, 2015.

21 Daniel Sonntag, Volker Tresp, Sonja Zillner, Alexander Cavallaro, Matthias Hammon, André Reis, Peter A. Fasching, Martin Sedlmayr, Thomas Ganslandt, Hans-Ulrich Prokosch, Klemens Budde, Danilo Schmidt, Carl Hinrichs, Thomas Wittenberg, Philipp Daumke, and Patricia G. Oppelt. The clinical data intelligence project - A smart data initiative. *Informatik Spektrum*, 39(4):290–300, 2016.

22 Ganggao Zhu and Carlos A. Iglesias. Computing semantic similarity of concepts in knowledge graphs. *IEEE Trans. on Knowl. and Data Eng.*, 29(1):72–85, January 2017.

## 14 Human and Social Factors in Knowledge Graphs

*Marta Sabou (TU Wien, AT), Elena Simperl (University of Southampton, GB), Eva Blomqvist (Linköping University, SE), Paul Groth (University of Amsterdam, NL & Elsevier Labs, NL), Sabrina Kirrane (Wirtschaftsuniversität Wien, AT), Gerard de Melo (Rutgers University - Piscataway, US), Barend Mons (Leiden University Medical Center, NL), Heiko Paulheim (Universität Mannheim, DE), Lydia Pintscher (Wikimedia Deutschland, DE), Valentina Presutti (STLab, ISTC-CNR, IT), Juan F. Sequeda (Capsenta Inc. - Austin, US), and Cogan Matthew Shimizu (Wright State University - Dayton, US)*

Knowledge graphs are created in socio-technical systems. The fewest of them are produced without any form of human or social activity. In most cases, whether it is community projects such as Wikidata and the Linked Open Data Cloud or enterprise knowledge graphs such as Amazon's product graph, various knowledge acquisition and curation aspects can hardly be fully automated for technical or operational reasons. Through their applications, knowledge graphs also reach people as end-users or consumers, and fuel algorithmic decision-making with potentially far-reaching economic and social impact.

Exploring the human and social factors of knowledge graphs has hence extensive benefits:

- It aids tech designers come up with methods and tools that support people at their knowledge graph work.
- It gives insight into the range of skills required to create, maintain and use knowledge graphs, hence making projects more effective.
- It helps develop an understanding of the social processes that underpin community projects such as Wikidata and DBpedia, and of the links between the social make-up of the community and the qualities of its outcomes.
- In the same context, it can assist community managers in improving teams' performance and collaboration, and spot areas that could benefit from a greater diversity of skills, interests and opinions.
- It informs the design of useful interfaces and representations of knowledge graphs, particularly towards application end-users with limited technical abilities.
- It supports the generation of richer explanations and other forms of transparency and accountability of AI systems.

People and communities engage with knowledge graphs in multiple ways. Interactions can be explicit (for instance, adding a statement to Wikidata) or implicit (for instance, clicking on a product in Amazon's knowledge graph when browsing the Amazon's website). To characterise them, we considered three dimensions:

- Scenarios (centred around the types of knowledge graphs created and the underlying processes): open vs. closed (e.g. in an enterprise); generic vs. domain-specific.
- Lifecyle of KGs (types of activities and the ways each activity impacts KGs): from KG creation to developing KG-based applications to using those applications.
- Types of users (roles across scenarios and KG lifecycle and their characteristics, needs, and expectations): individual vs. community contributors; lay people vs. experts; developers vs. end-users.

Following these dimensions, we discussed open research challenges, which require input from several disciplines: knowledge representation, knowledge engineering, social computing, social sciences, behavioural economics, software engineering and HCI.

## 14.1 Challenges

**Tailored support to KG engineering.** The scenarios introduced earlier greatly influence the choice of methods and tools to create and curate knowledge graphs. Organisations operate within existing process, project and quality assurance frameworks. Grassroots initiatives tend to define such frameworks as they advance with their work, leveraging the skills, ideas and views of the participants. Hence, an approach suitable to develop a knowledge graph in an enterprise context is likely to need substantial alterations if applied to a bottom-up, community-driven project.

Existing knowledge and ontology engineering literature is rich in methodologies, tools, and best practices, which should be revisited and updated to state-of-the-art technology and practice. For open, decentralised scenarios, we have access to several notable initiatives which could be subject to observational studies, interventions, and experiments to map the research space and derive guidelines for system designers and community managers. As organisations embrace participatory approaches in internal contexts to seek a broader base of contributions and ideas, it would be equally interesting to study how ideas and lessons learned in large community projects could transfer to enterprise or government contexts.

We have identified the following needs:

- Update existing methodological guidelines to match the skills and requirements of their audiences, offer bespoke support to different scenarios, and be clear about the scenarios they work best in.
- Add advanced features for the use of patterns in tool, for example through search and composition.
- Provide platforms and champion the publication of case studies and experience reports, for example in the form of data study groups that bring together experts in databases, RDF, knowledge acquisition, online communities etc.
- Create a stronger culture of user-centric research and encourage comprehensive evaluations of new methodologies, for example through the use of crowdsourcing.

**Transparency and accountability in AI** Knowledge graphs are a valuable resource. They empower decision-making algorithms and support people in seeking information. In a world of filter bubbles and fake news, is it more important than ever to be able to explains how particular outcomes or conclusions came about and knowledge graphs can turn into powerful gateways to produce more transparent and accountable AI ecosystems. Considering the three dimensions introduced earlier, any approach to explainability will have to be tailored to the specific scenario, activities and roles involved, from developers issuing SPARQL queries against Wikidata to end-users asking questions to intelligent assistants such as Siri, which leverages Wikidata to generate answers.

We discussed features of knowledge graph representation which would help developers add transparency and accountability by design to their knowledge graph applications. Provenance and trust, as well as the ability to capture knowledge diversity were mentioned in this context. Challenges remain in understanding the best ways to use these features, which some knowledge graphs already offer, in applications and to create tools and incentives for developers to explore them in greater detail. More research is needed, for example in the form of studies using ethnographic,as well as other qualitative and quantitative methods to build a better understanding how developers work with knowledge graphs and how it could be improved to support them in delivering transparent, accountable knowledge graph experiences.

In addition, the knowledge graph community should consider existing work from related fields on frameworks and approaches to communicate provenance and uncertainty of data and analyses built on top of knowledge graphs to support question answering, information retrieval and decision making. There are also opportunities to advance the state of the art in human data interaction and data visualisation, as most relevant literature has focused on tabular, numerical data rather than graphs, possibly labelled in multiple languages.

Finally, while some knowledge graph projects have been ahead of the curve in raising awareness about the need to promote knowledge diversity, in most cases we have a rather limited understanding of how biased knowledge graphs are. By the nature of knowledge representation, knowledge graphs will capture an simplified view of the world. They may abstract from particular details, make choices on how to model specific aspects and vary in the quality and level of detail of the information they cover.

We need more research into knowledge representation approaches that can tackle complexity and methods and tools to make it easier for developers to support and facilitate contextual depth, diversity and potentially conflicting viewpoints when processing and analysing information from a knowledge graph. Studies in other disciplines, including cognitive, behavioural, social and political sciences can offer very interesting impulses to put forward proposals that are not only novel from a technology point of view, but also match the capabilities, needs and expectations of the people engaging with the information - some participants noted there are lessons to be learned from existing proposals such as link sets which were not effective in user studies [1].

Besides ways to capture and use additional knowledge graph features, there is a need for representations and methods to study the inherent biases knowledge graphs suffer from as they evolve in time. Data ingestion, social processes and the availability of resources and expertise inadvertently lead to imbalances in the content and quality of a knowledge graph, which may be remedied in time. To be able to do so effectively, we need new models and techniques to analyse and improve the quality of knowledge graphs (e.g. completeness, correctness) and to conceptualise and measure diversity. We discussed the challenges arising from defining a suitable framework of reference, as biases in a knowledge graph may merely reflect biases in the data sources it relies on. Visual analytics and dashboards, similar to what Wikidata does with geographical coverage of entities, could be used to monitor additions and changes to a knowledge graph and detect anomalies, though more research is needed to understand user requirements for these tools in a knowledge graph context.

**Make knowledge graph research truly interdisciplinary** There are many techniques that can facilitate engagement with knowledge graphs: entity-centric exploratory search; narrative generation; or games with a purpose. To be effective, these techniques need to be better aligned with theory and empirical evidence from cognitive sciences, which teach us how different demographics and professional groups create, organise and make sense of knowledge. For example, healthcare practitioners are used to work with lists of concepts, rather than networks and graphs. Studies in cognitive sciences can help evaluate knowledge representation decisions and inform the design of the tools and applications through which people interact with knowledge graph structures, suggesting effective ways to render and present them.

The UX of entity-centric exploratory tasks could be greatly enhanced if considering theories about the nature of learning and the best ways to support it. For example, tools could start by showing users entities they are familiar with, or entities that are far away from their area of expertise, depending on the effect that needs to be achieved. Learning considerations should also inform the design of evaluations and benchmarks, which should be extended to capture the effects a particular technique or algorithm had on the reference model of their users.

Natural language generation (NLG) can be used to create text summaries of datasets which are more accessible to audiences that are not familiar with the particulars of data modeling and engineering. Accessibility is critical to allow a greater range of people and communities to contribute to knowledge graphs. In open contexts, this is very much aligned with their diversity and inclusivity agendas. For enterprise knowledge graphs, broadening the base of contributors is a pathway to sustainability. Existing narrative generation techniques follow a tech-centric approach, using rule-based templates to capture domain knowledge or data-heavy deep learning. The underlying models should incorporate insights from theoretical and empirical cognitive science to ensure that the text they produce matches the abilities and needs of the people using it.

Beyond text, the human data interaction community is exploring alternative data representations, interfaces and experiences, for instance by using games with a purpose, interactive storytelling, or virtual and augmented reality. These approaches could be applied to knowledge graphs, which through their rich content and connectedness offer an interesting playground for the development of bespoke projects that appeal to broader, non-expert audiences in various professional roles.

In summary, there is a need to reach out to communities traditionally involved in studying human and social factors. All challenges discussed by the group would benefit from insights and methods from complementary disciplines, including HCI and behavioural, cognitive, social and political sciences. We should as a community strive to support the organisation of workshops at venues such as CSCW, CHI, Information Science and Web Science, including researchers and practitioners from other fields early on.

## 14.2 Summary

In summary, this working group concluded that a major challenge related to human and social factors of knowledge graphs lies in leveraging theory, methods and empirical evidence from other disciplines in order to:

- Understand the cognitive and social processes by which knowledge (and knowledge shaped as a graph) emerges.
- Identify patterns and best practices to support these processes.
- Improve developer experience to allow them to create, curate and reuse knowledge graph effectively.
- Build an understanding of the frameworks, methods and tools required to support developers create knowledge graph applications that are transparent and accountable by design.
- Provide guidelines and best practices to help developers use and appreciate large-scale knowledge graphs that are inherently messy, diverse and evolving.
- Understand what social features (for example, expertise, motivation, team composition) influence what qualities of the graph.

### References

1 Al Koudous Idrissou, Rinke Hoekstra, Frank van Harmelen, Ali Khalili, and Peter van den Besselaar. Is my:sameas the same as your:sameas?: Lenticular lenses for context-specific identity. In *Proceedings of the Knowledge Capture Conference*, K-CAP 2017, pages 23:1–23:8, New York, NY, USA, 2017. ACM.

## 15 Applications of Knowledge Graphs

*Sarven Capadisli (TIB - Hannover, DE) and Lydia Pintscher (Wikimedia - Germany, DE)*

We report on two application areas for knowledge graphs as well as potential research directions.

### 15.1 Scholarly Knowledge

Findability and use of scholarly and scientific knowledge is integral to the advancement of human knowledge. Scholarly knowledge includes a range of research artefacts that needs to be described and connected. These include research articles, peer reviews, research data, social interactions like review requests and notifications in general, as well as different kinds of annotations on research objects.

The common methods of access and (re)use of scholarly output is technically and ethically inadequate for the society at large. By enabling accessible scholarly knowledge graphs as well as applications which make use of it, we hope to enable universal access to previous research. By improving the availability through accessible knowledge graphs, we can facilitate discovery and building on existing research.

The construction of a scholarly commons that meets the core requirements of a scholarly communication system, i.e. registration, awareness, certification, and archiving functions in scientific communication [3], that is accessible to both humans and machines would require capturing information at different parts of the process. The incentive for content creators and consumers of content would require lowering the barriers through useful and accessible applications.

Some of the open challenges include identifying and building mechanisms to deal with disagreements both within scholarly knowledge as well as knowledge on the Web at large. Improving tooling to create knowledge graphs also facilitates the discovery of knowledge. For example, academic authors can find relevant scientific assertions during the writing process. Hence, one of the research directions is to investigate and develop effective ways to represent fine-grained information. Knowledge graphs at different degrees of abstraction can be formulated, whether they are about a collection of documents, at the document level, or for any unit of information.

Other research directions would include the creation and management of scholarly journals that are machine-readable. One of the related challenges in this respect is an aim to decouple the *registration* and *certification* functions in scientific communication so that free expression can be exercised towards materialising open and accessible scholarly knowledge [4].

Further research directions would include the development of interoperable applications to improve discovery, accessibility, integrability, and reusability of knowledge graphs. More generally, facilitating interactions with knowledge graphs where applications enabling read-write operations on information at distributed locations with different access controls, factoring in user's privacy and security, as well as, allowing multiple researcher identities (eg personal, professional) working on different parts of a graph. For example, along the lines of research and development around decentralised and interoperable systems [1, 2].

## 15.2 Wikidata

Wikidata [5] is a knowledge base that has gained a lot of attention over the last years and is growing. It has various interesting features that the knowledge graph community can learn from. It also faces a number of challenges.

Wikidata's data model has a number of features that make dealing with diverse knowledge possible for its community. These include the ability to record conflicting data as well as qualifying data with additional statements to for example give it temporal information and record the provenance of the data. It strikes an interesting balance between a strict ontology that makes re-use easy and a flexible data model that makes it possible to capture the complexity of the world.

Through its open and flexible nature Wikidata also faces a number of challenges. These can roughly be grouped along the data lifecycle of import, clean-up/maintenance and export. Some of them include:

- During import it is a challenge to understand what consequences adding a particular change will have. This includes violating of constraints in the system or unintended changes to the larger ontology.
- Wikidata lacks provenance information in the form of references for a considerable part of the existing statements it contains. It is a challenge to find references in an automated way for this existing data.
- Some of the concepts in Wikidata are not separated cleanly (e.g. a website and the company operating the website being treated as one accidentally). It is not trivial to find such concepts in order to separate them more cleanly.
- By its nature Wikidata is concept-centric. This makes it harder to see and understand the ontology and how individual classes fit into the ontology. Becauses of this lack of high-level overview it is easy for editors to make mistakes that have a significant influence on the ontology without realizing it.
- With its hundreds of millions of statements it is unavoidable that some of them become outdated, vandalized or otherwise wrong. These are hard to spot for editors and without more visibility this data is propagated to the users of that data.
- Users of Wikidata's data want to understand the state of the data before committing to using it in their application or visualisation. It is currently not possible to get an overview of the quality of a particular subset of the data.
- When using Wikidata's data a number of statements will have multiple values like the prizes a famous person won. Wikidata's built-in mechanism to rank those values by are not always sufficient.
- Querying Wikidata's data is an important way to access the data and gain new insights. It requires writing SPARQL queries, which is not possible for a significant part of the intended users - especially considering the data model's special features (qualifiers, ranks, references).

We invite the research community in helping us address these challenges in order to build and maintain a general-purpose knowledge graph for everyone.

### References

1 S. Capadisli, A. Guy, R. Verborgh, C. Lange, S. Auer, and T. Berners-Lee. Decentralised authoring, annotations and notifications for a read-write web with dokieli. 2017.

2 E. Mansour, A. V. Sambra, S. Hawke, M. Zereba, S. Capadisli, A. Ghanem, A. Aboulnaga, and T. Berners-Lee. A demonstration of the solid platform for social web applications. In

*Proceedings of the 25th International Conference Companion on World Wide Web*, WWW '16 Companion, pages 223–226. International World Wide Web Conferences Steering Committee, 2016.

3 H. Roosendaal and P. Geurts. Forces and functions in scientific communication: an analysis of their interplay. cooperative research information systems in physics, august 31-september 4 1997.(oldenburg, germany), 1997.

4 H. Van de Sompel and A. Treloar. A perspective on archiving the scholarly web. In *Proceedings of the International Conference on Digital Preservation (iPRES)*, pages 194–198, 2014.

5 D. Vrandečić and M. Krötzsch. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85, 2014.

## 16 Knowledge Graphs and the Web

*Sarven Capadisli (TIB - Hannover, DE), Michael Cochez (Fraunhofer FIT - Sankt Augustin, DE), Claudio Gutierrez (University of Chile - Santiago de Chile, CL), Andreas Harth (Fraunhofer IIS - Nürnberg, DE), and Antoine Zimmermann (École des Mines de Saint-Étienne, FR)*

### 16.1 Introduction

The web provides a large store of data and information from which knowledge graphs can be constructed. The raw input that can be sourced from the web for constructing knowledge graphs differs in formality, in variety and scope, from text to HTML and XSL documents to RDF. Some knowledge graphs rely on information extracted from unstructured sources, while other knowledge graphs focus on information already available as RDF.

In the text, we focus on different ways to organise knowledge graphs built from structured data such as RDF or other graph-centric frameworks.

In this report we focus essentially on the counterpoint between centrally organized versus decentralized knowledge graphs. Some current examples of centralised or organisational knowledge graphs are the Google Knowledge Graph (former: Freebase), Wikidata, and DBpedia. Examples of decentralized or personal knowledge graphs are the Linking Open Data (LOD) cloud and the collection of all Linked Data on the web. The web with its decentralised architecture is able to accommodate in principle both approaches.

### 16.2 Problem Statement

Centralised knowledge graphs and decentralised knowledge graphs exhibit different qualities. To make informed decisions on which of these designs to use, we need to find out the trade-offs between them. Once these quality criteria are established, we want to know whether these aspects are inherent to the fact whether the knowledge graph is centralised or decentralised. If they are not, we want to be able to transfer approaches and techniques in one design approach to the other.

### 16.3 The Centralised vs. Decentralised Spectrum

There is a spectrum of knowledge graphs being published: on the one hand, the more organisational-centric, which are centralised, and on the other hand the more individual-centric, which are decentralised.

The spectrum between centralised and decentralised approaches has been a topic of research in information systems [2, 1] and databases [7] in general and in personal data [4], social networks [3] and querying the Web of Data [5, 6] in particular.

In the following, we contrast properties of centralised and decentralised knowledge graphs. We identified the following characteristics. We start with centralised knowledge graphs and first list characteristics related to organization, then move on to characteristics related to implementation.

- *Survivability and Robustness:* Centralised knowledge graphs can be controlled by a single entity that can decide strategic issues, thus giving more powerful control over all aspects of the knowledge graph. This can be beneficial for their owners but, with regard to survivability and robustness, not necessarily for its users (e.g., the shutdown of Freebase). Besides organizational aspects, centralised knowledge graphs rely on one big node and can thus can become unavailable for technical reasons.
- *Stability:* Centralised knowledge graphs have uniform terms of use, a degree of promise of longevity and availability. Hence, they have a more stable behaviour and are less likely to suffer from disappearing links or can at least give a clear indication when a link would disappear, or preserve the provenance information of changes made.
- *Curation:* Centralised knowledge graphs have a clear curation mechanism in place, which is expected to have a positive effect on the consistency and quality of the data.
- *Rights and License:* Centralized knowledge graphs usually have a clear license. Wikidata and DBpedia are intended as public resources, indicated by their respective CC0 and CC BY-SA 3.0 Unported licence. Other organizational knowledge graphs are typically seen as an internal organizational asset. Sometimes parts of these are also made publicly available, especially for reasons of interoperability with other systems. Examples include the schemas developed by schema.org used by search engines to extract information from web pages and Thomson Reuters' permid.org, which has a public part licensed using CC-BY 4.0 and CC-NC 3.0, and a part of the data not publicly available.
- *Service Level Agreements:* Centralized knowledge graphs may have Service Level Agreements (SLAs), giving clear guarantees on aspects such as response time and correctness. One could even imagine a company being liable for the information provided.
- *Privacy and Security:* If the KG contains personal information, with centralized knowledge graphs, users give up some privacy in exchange of some convenience. In particular, the access logs are also centralised, which facilitates analysing the logs. However, because the graph is centralized, the company responsible for it might have more means to devise proper security measures for the system. Indeed, users would not be prepared to share their data if it appears the solution is not secure.

Characteristics related to implementation include:

- *Identifiers:* Centralised knowledge graphs control and manage their own identifiers and namespaces. Not all knowledge graphs (e.g., the Google Knowledge Graph) expose their identifiers to the outside. Sometimes centralised knowledge graphs do link to external sources as well, leading to a hub-and-spoke structure of the graph.
- *Schema:* Centralised knowledge graphs have a single schema, used all over the data.
- *Query:* Centralised knowledge graphs offer an API for data retrieval (or provide data dumps) and querying. These systems are furthermore optimised for the type of queries they support.
- *Location:* Centralised knowledge graphs require substantial computing infrastructure to be able to handle the load, especially for knowledge graphs that offer a query interface.
- *Timeliness:* If data for centralised knowledge graphs needs to be aggregated from outside sources, the update interval of the aggregator influences the overall timeliness of the knowledge graph.
- *Modularity/Allocation:* Resources are allocated at the central point, so infrastructure has to be provided at the central point.
- *Data Volume:* Centralized knowledge graphs can be partitioned according to centrally defined criteria.

- *Consistency:* Given a centrally managed curation process, in combination with test cases run on the integrated data, the overall consistency of centralised knowledge graphs can be ensured.
- *Load Balancing:* Access load can be easily balanced in an organisation's internal infrastructure.

Decentralised knowledge graphs promise benefits which cannot be offered by centralised variants. Again, we start with aspects related to organisation, and then list aspects related to implementation.

- *Survivability and Robustness:* Decentralized knowledge graphs do not assume a central authority over all resources. With decentralised variants, resources are independently created, managed, and distributed such that unavailability of any particular part will not necessarily influence the other knowledge graphs. Hence, one could say that the system has a higher survivability and robustness, at the cost of a chance that parts become unavailable.
- *Stability:* The decentralised architecture, once socialised and embraced by a critical mass of users, gives strong stability to applications, not depending on external influences (like change of owner, company commercial decisiones, etc.). Although parts of the knowledge graphs can become unavailable if individual providers go offline, the overall knowledge graph is potentially more resilient than centralized systems.
- *Curation:* As the experience of free software has shown, curating open artifacts is far more convenient that closed or private ones.
- *Rights and License:* The different parts are owned by their creators; parts of decentralised knowledge graphs tend to be copyrighted or unlicensed, however often with intention for public reuse.
- *Service Level Agreements:* As data is not centralized, there are no SLAs. Indeed, it would be impossible to enforce them in case they are not met. On the other hand, because the data is decentralized, one could in case of need decide to make local copies of the data, i.e., centralize parts for which a specific level of service is expected.
- *Privacy and Security:* With decentralised knowledge graphs, users retain their privacy to the extent that usage patterns are distributed across systems. In these systems the security aspect is often not stringent as the data is intended to be shared. One exception would be the provenance of the data. Techniques like signing statements using public-private key infrastructure exist to enable guaranteed authenticity of statements.

Characteristics related to implementation include:

- *Identifiers:* Even when knowledge is distributed, there could be shared vocabularies. Besides, one could implement a system for shared identifiers where people can create their own (persistent) identifiers, hoping that others will reuse these. However, one would not expect there to be a centralized identifier system which all users have to adhere to, as this would become a single point of failure and essentially make this a centralized system. In effect, one cannot prevent users from using identifiers in a wrong way or even for completely wrong entities, nor can it be prevented that new identifiers are created for already existing entities.
- *Schema:* As mentioned, in a distributed system, there can be shared schemas. However, anyone would be able to add their own schema parts to the system. One could also imagine a distributed system in which users get to vote for schemas. This could be combined with micropayments.

- *Query:* As the system is not centralized, it would be expected that querying results in more communication and total processing overhead. Data has to be aggregated from many sources during query time. Parallelized data access can mitigate some of the performance penalty, however, wide-area network access remains much slower than access to local storage.
- *Location:* Decentralised knowledge graphs can be made available from any location (e.g., own personal web server)
- *Timeliness:* The data in decentralised knowledge graphs is updated immediately whenever the data owner wants to update it.
- *Modularity/Allocation:* The distributed nature results in a distributed allocation of resources (processing, storage, and communication). The cost of deployment of the graph is a) distributed over many parties and b) the upfront cost (capital expenditure) is low for each participant.
- *Data Volume:* Distributed knowledge graphs are already partitioned into smaller parts where each has a specific purpose.
- *Consistency:* Parts of the knowledge graphs are likely more consistent as providers maintain their own data. But the combination of parts of the graph is likely to be less consistent as in a centralized approach.
- *Load Balancing:* The decentralisation works effectively as a load balancing when considering the knowledge graph as a whole.

## 16.4 Conclusion

We have identified the above directions for further research and development of applications, to enable content creators and consumers to better access distributed resources. Such applications should aim to preserve the benefits decentralised approaches (e.g., be privacy-aware, support multiple online user identities, provide mechanisms for extensible resource descriptions, enable read-write operations on personal and group data storages) and conform to interoperable open web standards. Many of such applications could benefit from visualisations of schema-less data.

It is possible that a compromise emerges: knowledge graphs with a curated, centrally organised core, which on the fringes of the graph link external knowledge graphs under diverse ownership. The way DBpedia and Wikidata evolved points in such a direction.

In all but the fully centralised approaches, a research direction concerns the notion of agents that are able to access, integrate and interact with distributed knowledge graphs. Instead of having a single centralised component that provides a single integrated knowledge graph, the notion of agents could bring more flexibility and distribution. A starting point could be both resident agents (running on one user's machine) and transient agents that roam from machine to machine in the network. Especially in the area of connected devices, further research is needed to provide the ability to access knowledge graphs from many devices.

On the non-technical side, further research could consider moving organisational structures from the centralised approach to the decentralised approach, including the design of business models in decentralised settings.

### References

1 Roger Alan Pick. Shepherd or servant: Centralization and decentralization in information technology governance. *International Journal of Management & Information Systems (IJMIS)*, 19:61, 03 2015.

2 John Leslie King. Centralized versus decentralized computing: Organizational considerations and management options. *ACM Comput. Surv.*, 15(4):319–349, December 1983.

3 Ching man Au Yeung, Ilaria Liccardi, Kanghao Lu, Oshani Seneviratne, and Tim Berners-lee. Decentralization: The future of online social networking. In *In W3C Workshop on the Future of Social Networking Position Papers*, 2009.

4 Arvind Narayanan, Vincent Toubiana, Solon Barocas, Helen Nissenbaum, and Dan Boneh. A critical look at decentralized personal data architectures. *CoRR*, abs/1202.4503, 2012.

5 J. Umbrich, C. Gutierrez, A. Hogan, M. Karnstedt, and J. Xavier Parreira. Eight fallacies when querying the web of data. In *2013 IEEE 29th International Conference on Data Engineering Workshops (ICDEW)*, pages 21–22, April 2013.

6 Jürgen Umbrich, Claudio Gutierrez, Aidan Hogan, Marcel Karnstedt, and Josiane Xavier Parreira. The ACE theorem for querying the web of data. In *Proceedings of the 22Nd International Conference on World Wide Web*, WWW '13 Companion, pages 133–134, New York, NY, USA, 2013. ACM.

7 Patrick Valduriez. Principles of distributed data management in 2020? *CoRR*, abs/1111.2852, 2011.