DAGSTUHL
REPORTS

Volume 8, Issue 9, September 2018

Dagstuhl Reports, Vol. 8, Issue 9 ISSN 2192-5283

ISSN 2192-5283

Published online and open access by
Schloss Dagstuhl – Leibniz-Zentrum für Informatik GmbH, Dagstuhl Publishing, Saarbrücken/Wadern, Germany. Online available at
http://www.dagstuhl.de/dagpub/2192-5283

Publication date
March, 2019

Bibliographic information published by the Deutsche Nationalbibliothek
The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at `http://dnb.d-nb.de`.

Digital Object Identifier: 10.4230/DagRep.8.9.i

Aims and Scope
The periodical *Dagstuhl Reports* documents the program and the results of Dagstuhl Seminars and Dagstuhl Perspectives Workshops.
In principal, for each Dagstuhl Seminar or Dagstuhl Perspectives Workshop a report is published that contains the following:

- an executive summary of the seminar program and the fundamental results,
- an overview of the talks given during the seminar (summarized as talk abstracts), and
- summaries from working groups (if applicable).

This basic framework can be extended by suitable contributions that are related to the program of the seminar, e. g. summaries from panel discussions or open problem sessions.

Contact
Schloss Dagstuhl – Leibniz-Zentrum für Informatik
Dagstuhl Reports, Editorial Office
Oktavie-Allee, 66687 Wadern, Germany
reports@dagstuhl.de
http://www.dagstuhl.de/dagrep

Report from Dagstuhl Seminar 18361

Measuring the Complexity of Computational Content: From Combinatorial Problems to Analysis

Edited by

Vasco Brattka[1], Damir D. Dzhafarov[2], Alberto Marcone[3], and Arno Pauly[4]

1 **Universität der Bundeswehr – München, DE,** vasco.brattka@cca-net.de
2 **University of Connecticut – Storrs, US,** damir@math.uconn.edu
3 **University of Udine, IT,** alberto.marcone@uniud.it
4 **Swansea University, GB,** arno.m.pauly@gmail.com

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 18361 "Measuring the Complexity of Computational Content: From Combinatorial Problems to Analysis". It includes abstracts of talks presented during the seminar and open problems that were discussed, as well as a bibliography on Weihrauch complexity that was started during the previous meeting (Dagstuhl seminar 15392) and that saw some significant growth in the meantime. The session "Solved problems" is dedicated to the solutions to some of the open questions raised in the previous meeting (Dagstuhl seminar 15392).

Seminar September 2–7, 2018 – http://www.dagstuhl.de/18361
2012 ACM Subject Classification Theory of computation → Models of computation, Theory of computation → Complexity classes, Mathematics of computing → Numerical analysis, Theory of computation → Logic
Keywords and phrases Computability and complexity in analysis, computations on real numbers, reducibilities, descriptive complexity, computational complexity, reverse and constructive mathematics
Digital Object Identifier 10.4230/DagRep.8.9.1
Edited in cooperation with Marta Fiori Carones

1 Executive Summary

Vasco Brattka (Universität der Bundeswehr – München, DE)
Damir D. Dzhafarov (University of Connecticut, US)
Alberto Marcone (University of Udine, IT)
Arno Pauly (Swansea University, GB)

Reducibilities such as many-one, Turing or polynomial-time reducibility have been an extraordinarily important tool in theoretical computer science from its very beginning. In recent years these reducibilites have been transferred to the continuous setting, where they allow us to classify computational problems on real numbers and other continuous data types.

In the late 1980s Weihrauch has introduced a reducibility that can be seen as an analogue of many-one reducibility for (multi-valued) functions on infinite data types. This reducibility,

Measuring the Complexity of Computational Content: From Combinatorial Problems to Analysis,
Dagstuhl Reports, Vol. 8, Issue 09, pp. 1–28
Editors: Vasco Brattka, Damir D. Dzhafarov, Alberto Marcone, and Arno Pauly
Dagstuhl Reports
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

now called *Weihrauch reducibility*, was studied since the 1990s by Weihrauch's school of computable analysis and flourished recently when Gherardi and Marcone proposed this reducibility as a tool for a uniform approach to reverse analysis.

Reverse mathematics aims to classify theorems according to the axioms that are needed to prove these theorems in second-order arithmetic. This proof theoretic approach yields non-uniform classifications of the computational content of certain theorems. However, many of these classifications also have uniform content and Weihrauch complexity allows us to study this uniform computational content directly using methods of computability theory.

This perspective has motivated Dorais, Dzhafarov, Hirst, Mileti and Shafer, on the one hand, Hirschfeldt and Jockusch, on the other hand, to study combinatorial problems using this approach. This research has led to a number of further reducibilities (computable reducibility, generalized Weihrauch reducibility and others) that can be seen as non-uniform or less resource sensitive versions of Weihrauch reducibility. Using this toolbox of reducibilities one can now adjust the instruments exactly according to the degree of uniformity and resource sensitivity that one wants to capture.

A precursor seminar[1] that was also held at Dagstuhl has been instrumental in bringing together researchers from these different communities for the first time. This has created a common forum and fostered several research developments in this field. We believe that the current seminar was very successful in strengthening and deepening the collaborations between the involved communities. Ample time was left and successfully used for research in groups. A novelty of the current seminar was a special session at which solutions of open problems from the previous seminar were presented. To see that several of the major open problems of the previous meetings were solved in the meantime was inspiring and motivating! Some of the solutions involve new techniques with a wider applicability. Hopefully, we will see solutions to some of the open questions presented at the current seminar in the not too far future! Altogether, the seminar did proceed in a highly productive atmosphere, thanks to many excellent contributions from participants. Inspired by these contributions the organizers are planning to edit a special issue of the journal *Computability* dedicated to this seminar.

This report includes abstracts of many talks that were presented during the seminar, it includes a list of some of the open problems that were discussed, as well as a bibliography on Weihrauch complexity that was started during the previous meeting and that saw significant growth in the meantime. Altogether, this report reflects the extraordinary success of our seminar and we would like to use this opportunity to thank all participants for their valuable contributions and the Dagstuhl staff for their excellent support!

[1] 15392 Measuring the Complexity of Computational Content: Weihrauch Reducibility and Reverse Analysis, see https://doi.org/10.4230/DagRep.5.9.77

2 Table of Contents

Executive Summary

Overview of Talks

Solved questions

Open problems

3 Overview of Talks

3.1 Weihrauch goes Brouwerian

Vasco Brattka (Universität der Bundeswehr – München, DE) and
Guido Gherardi (University of Bologna, IT)

Main reference Vasco Brattka, Guido Gherardi: "Weihrauch goes Brouwerian", CoRR, Vol. abs/1809.00380, 2018.
URL http://arxiv.org/abs/1809.00380

We prove that the Weihrauch lattice can be transformed into a Brouwer algebra by the consecutive application of two closure operators in the appropriate order: first completion and then parallelization. The closure operator of completion is a new closure operator that we introduce. It transforms any problem into a total problem on the completion of the respective types, where we allow any value outside of the original domain of the problem. This closure operator is of interest by itself, as it generates a total version of Weihrauch reducibility that is defined like the usual version of Weihrauch reducibility, but in terms of total realizers. From a logical perspective completion can be seen as a way to make problems independent of their premises. Alongside with the completion operator and total Weihrauch reducibility we need to study precomplete representations that are required to describe these concepts. In order to show that the parallelized total Weihrauch lattice forms a Brouwer algebra, we introduce a new multiplicative version of an implication. While the parallelized total Weihrauch lattice forms a Brouwer algebra with this implication, the total Weihrauch lattice fails to be a model of intuitionistic linear logic in two different ways. In order to pinpoint the algebraic reasons for this failure, we introduce the concept of a Weihrauch algebra that allows us to formulate the failure in precise and neat terms. Finally, we show that the Medvedev Brouwer algebra can be embedded into our Brouwer algebra, which also implies that the theory of our Brouwer algebra is Jankov logic.

3.2 Effectivity and Reducibility with Ordinal Turing Machines

Merlin Carl (Universität Konstanz, DE)

Main reference Merlin Carl: "Generalized Effective Reducibility", in Proc. of the Pursuit of the Universal – 12th Conference on Computability in Europe, CiE 2016, Paris, France, June 27 – July 1, 2016, Proceedings, Lecture Notes in Computer Science, Vol. 9709, pp. 225–233, Springer, 2016.
URL http://dx.doi.org/10.1007/978-3-319-40189-8_23

By taking Turing computability as its basic notion of effectivity, the study of Weihrauch reducibility is restricted to realms where objects are countable or can be encoded by countable objects. By replacing Turing machines with Koepke's Ordinal Turing Machines (OTMs), we obtain a notion of effective reducibility that applies to sets of arbitrary size. We can then ask for arbitrary Π_2-statements in the language of set theory whether they are effective or whether one is effectively reducible to the other. As a sample application, we consider several variants of the axiom of choice and see that the versions with systems of representations and choice functions are effectively equivalent, while the well-ordering principle is strictly stronger.

By taking OTMs as the underlying concept of effectivity, we can also reinterpret the realizability interpretation of intutionistic logic, thus obtaining a notion of effectivity for set-theoretical statements of arbitrary quantifier complexity. In this sense, the axioms of KP turn out to be effective, while the power set axiom and the axioms of replacement and separation are not.

3.3 Around finite basis results for topological embeddability between functions

Raphael Carroy (Universität Wien, AT)

We say that a function f embeds (topologically) in a function g when there are two (topological) embeddings σ and τ satisfying $\tau \circ f = g \circ \sigma$. This quasi-order is a strengthening of the topological strong Weihrauch reducibility. In recent years, various subclasses of analytic functions were shown to admit a finite bases under topological embeddability, including non-σ-continuous functions (Solecki-Pawlikovski-Sabok) and non-Baire-class-one functions (in a joint work with Benjamin Miller). In an effort to understand if topological embeddability could be a well-quasi-order, which would mean that every subclass of functions admits a finite basis under embeddability, we recently proved a dichotomy for spaces of continuous functions with compact Polish 0-dimensional domains: embeddability is either analytic complete or a well-quasi-order.

References

1 Raphaël Carroy, Yann Pequignot and Zoltán Vidnyánszky. *Embeddability on functions: Order and Chaos*. To appear in Transactions of the American Mathematical Society. See also https://arxiv.org/abs/1802.08341

3.4 On the Solvability Complexity Index hierarchy, the computational spectral problem and computer assisted proofs

Matthew Colbrook (Cambridge University, GB)

Joint work of Matthew Colbrook, Anders Hansen
Main reference Matthew J. Colbrook, Bogdan Roman and Anders C. Hansen: "How to compute spectra with error control", submitted, 2018.
Main reference Jonathan Ben-Artzi, Matthew J. Colbrook, Anders C. Hansen, Olavi Nevanlinna and Markus Seidel: " On the solvability complexity index hierarchy and towers of algorithms", 2018.

We will discuss the Solvability Complexity Index (SCI) hierarchy, which is a classification hierarchy for all types of problems in computational mathematics that allows for classifications determining the boundaries of what computers can achieve in scientific computing. The SCI hierarchy captures many key computational issues in the history of mathematics including the insolvability of the quintic, Smale's problem on the existence of iterative generally convergent algorithm for polynomial root finding [1] (and McMullen's solution [2]), the computational spectral problem [3], inverse problems, optimisation, PDEs etc., and also mathematical logic.

Perhaps surprisingly, many of the classifications in the SCI hierarchy do not depend on the model of computation used.

The SCI hierarchy allows for solving the long standing computational spectral problem, and reveals potential surprises. For example, the problem of computing spectra of compact operators, for which the method has been known for decades, is strictly harder than the problem of computing spectra of Schrödinger operators with bounded potentials, which has been open for more than half a century. We also provide an algorithm for the latter problem, thus finally resolving this issue [4]. Moreover, the SCI hierarchy helps classifying problems suitable for computer assisted proofs. In particular, undecidable or non-computable problems are used in computer assisted proofs, where the recent example of the resolution of Kepler's conjecture (Hilbert's 18th problem) is a striking phenomenon [5]. However, only certain classes of non-computable problems can be used in computer assisted proofs, and the SCI hierarchy helps detecting such classes. As we will discuss, the problems of computing spectra of compact operators and Schrödinger operators with bounded potentials are both non-computable, however, whereas the compact case is in general unsuitable for computer assisted proofs, the Schrödinger case is indeed suitable. We will also discuss exciting new algorithms for computing spectra with error control and provide some cutting edge numerical examples [6].

References

1 Steve Smale. *On the complexity of algorithms of analysis*, Bull. AMS, 13 (1985), pp 87–121

2 Curt McMullen. *Families of rational maps and iterative root-finding algorithms*, Annals of Mathematics. Second Series, 125 (1987), pp 467–493

3 Anders C. Hansen. *On the solvability complexity index, the n-pseudospectrum and approximations of spectra of operators*, JAMS, 1 (2011), pp 81–124

4 Jonathan Ben-Artzi, Matthew J. Colbrook, Anders C. Hansen, Olavi Nevanlinna and Markus Seidel. *On the solvability complexity index hierarchy and towers of algorithms*, (2018)

5 Thomas C. Hales. *A proof of the Kepler conjecture*, Annals of Mathematics, (2005), pp 1065–1185

6 Matthew J. Colbrook, Bogdan Roman and Anders C. Hansen. *How to compute spectra with error control*, submitted, (2018)

3.5 Some properties of the countable space S_0

Matthew de Brecht (Kyoto University, JP)

In a generalization of Hurewicz's dichotomy theorem, we showed that a countably based co-analytic space is either quasi-Polish or else it contains a Π^0_2 subspace homeomorphic to one of four particular countable spaces (called S_2, S_1, S_D, and S_0). The spaces S_2 (the rationals), S_1 (the cofinite topology on the integers), and S_D (the Alexandrov topology on the natural numbers) are relatively well-known spaces and are often used as counter examples to various completeness properties (such as the Baire category theorem or sobriety).

In this talk we will look more closely at the space S_0 (finite sequences of natural numbers with a very weak topology), which is less well-known. Although S_0 does has some nice completeness properties (it is sober and every closed subset is a Baire space), we will show that it also resembles the space of rationals in several ways.

3.6 Ishihara's Boundedness Principle BD-N and below

Hannes Diener (University of Canterbury – Christchurch, NZ)

The aim of constructive reverse mathematics (CRM) is to classify theorems and principles over intuitionistic logic. The resulting hierarchy in many parts resembles parts of (Simpson style) reverse mathematics and parts of the Weihrauch lattice.

BD-N is one of the weakest principles that is of interest in CRM. It was introduced in the 1990ies by Hajime Ishihara to find a "logical" counterpart to the analytical statement that all sequentially continuous functions defined on a separable metric space are point-wise continuous. That characterisation makes it seem like quite a straightforward principle, however, from 2010 onward, there have been a number of statements identified that are all implied by BD-N, but that surprisingly lie strictly below it. Furthermore there is little understanding between how this statements interact.

This talk tries to present these ideas and hopefully initiate some discussion on whether this situation is reflected in the Weihrauch lattice or some variation thereof.

3.7 Some results in higher levels of the Weihrauch lattice

Jun Le Goh (Cornell University, US)

We present some results regarding higher levels of the Weihrauch lattice. We show that comparability of well-orderings is Weihrauch equivalent to its weak version, answering a question of Marcone. The proof proceeds via the ATR-like problem of producing the jump hierarchy on a given well-ordering. We also formulate a "two-sided" version of ATR: given a linear ordering L and a set of natural numbers A, produce either a jump hierarchy on L which starts with A, or an infinite L-descending sequence. We show that this problem is closely related to Koenig's duality theorem about countable bipartite graphs.

3.8 Trees Describing Topological Weihrauch Degrees of Multivalued Functions

Peter Hertling (Universität der Bundeswehr – München, DE)

We suggest definitions of continuous strong Weihrauch reducibility and of continuous Weihrauch reducibility on the set of functions mapping a subset of the Baire space to some quasi-order. Then we present descriptions of the corresponding topological strong Weihrauch degrees and of the topological Weihrauch degrees of Δ_2^0 measurable functions mapping the Baire space to some better-quasi-order, by suitable trees and forests and suitable reducibility relations on forests. We also consider Wadge degrees. Furthermore, we show that this leads to a similar description of the Wadge degrees, the topological strong Weihrauch degrees and the topological Weihrauch degrees of multivalued functions defined on a subset of a countably based T_0-space with range in a finite discrete space.

3.9 Leaf management

Jeffry L. Hirst (Appalachian State University – Boone, US)

Joint work of Caleb Davis, Jeffry Hirst, Jake Pardo
Main reference Caleb Davis, Jeffry Hirst, Jake Pardo, Tim Ransom: "Reverse mathematics and colorings of hypergraphs", Archive for Mathematical Logic, November, 2018.
URL https://doi.org/10.1007/s00153-018-0654-z

We demonstrate a process for transforming trees into trees with sets of leaf nodes. This process can be used to eliminate bootstrapping in certain reverse mathematics arguments, and may prove useful in calibrating Weihrauch strength of some statements. This talk includes joint work with Caleb Davis and Jake Pardo.

3.10 Degrees of randomized computability (Informal talk)

Rupert Hölzl (Bundeswehr University Munich, DE)

Joint work of Rupert Hölzl, Christopher P. Porter

In this survey we discuss work of Levin and V'yugin on collections of sequences that are non-negligible in the sense that they can be computed by a probabilistic algorithm with high probability. More precisely, Levin and V'yugin introduced an ordering on collections of sequences that are closed under Turing equivalence. Roughly speaking, given two such collections $\mathcal{A}$ and $\mathcal{B}$, $\mathcal{A}$ is less than $\mathcal{B}$ in this ordering if $\mathcal{A} \setminus \mathcal{B}$ is negligible. The degree structure associated with this ordering, the *Levin-V'yugin degrees* (or *LV-degrees*) can be shown to be a Boolean algebra, and in fact a measure algebra.

We demonstrate the interactions of this work with recent results in computability theory and algorithmic randomness: First, we recall the definition of the Levin-V'yugin algebra and identify connections between its properties and classical properties from computability theory. In particular, we apply results on the interactions between notions of randomness and Turing reducibility to establish new facts about specific LV-degrees, such as the LV-degree of the collection of 1-generic sequences, that of the collection of sequences of hyperimmune degree, and those collections corresponding to various notions of effective randomness. Next, we provide a detailed explanation of a complex technique developed by V'yugin that allows the construction of semi-measures into which computability-theoretic properties can be encoded. We provide examples of the uses of this technique by explicating and extending V'yugin's results about the LV-degrees of the collection of Martin-Löf random sequences and the collection of sequences of DNC degree, as well as results concerning atoms of the LV-degrees.

References

1 Leonid A. Levin and Vladimir V. V'yugin. Invariant properties of informational bulks. *Lecture Notes in Computer Science*, 53:359–364, 1977.
2 Vladimir V. V'yugin. On Turing-invariant sets. *Soviet Mathematics Doklady*, 17:1090–1094, 1976.
3 Vladimir V. V'yugin. Algebra of invariant properties of binary sequences. *Problemy Peredachi Informatsii*, 18(2):83–100, 1982.
4 Vladimir V. V'yugin. On empirical meaning of randomness with respect to parametric families of probability distributions. *Theory of Computing Systems*, 50(2):296–312, 2012.

3.11 Average-case polynomial-time computability of the three-body problem

Akitoshi Kawamura (Kyushu University, JP)

Joint work of Akitoshi Kawamura, Holger Thies, Martin Ziegler
Main reference Akitoshi Kawamura, Holger Thies, Martin Ziegler: "Average-Case Polynomial-Time Computability of Hamiltonian Dynamics", in Proc. of the 43rd International Symposium on Mathematical Foundations of Computer Science, MFCS 2018, August 27-31, 2018, Liverpool, UK, LIPIcs, Vol. 117, pp. 30:1–30:17, Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2018.
URL http://dx.doi.org/10.4230/LIPIcs.MFCS.2018.30

We apply average-case complexity theory to physical problems modeled by continuous-time dynamical systems. The computational complexity when simulating such systems for a bounded time-frame mainly stems from trajectories coming close to complex singularities of the system. We show that if for most initial values the trajectories do not come close to singularities the simulation can be done in polynomial time on average. For Hamiltonian systems we relate this to the volume of "almost singularities" in phase space and give some general criteria to show that a Hamiltonian system can be simulated efficiently on average. As an application we show that the planar circular-restricted three-body problem is average-case polynomial-time computable.

3.12 Weihrauch reducibility for some third order principles

Takayuki Kihara (Nagoya University, JP)

In order to examine the degrees of difficulty of separation principles on topological spaces, we introduce Weihrauch reducibilty for some third order principles. For instance, in terms of third order continuous Weihrauch reducibility, we show that (1) LLPO is not reducible to the closed separation principle on a separable metrizable space; (2) the open separation principle on a non-discrete second-countable Hausdorff space is equivalent to the uniform-LPO (the map on 1 returning the Kleene's type 2 object 2E) which is strictly stronger than lim; and (3) the coanalytic separation principle on a Polish space is located strictly between (some versions of) the Borel choice and the analytic choice.

3.13 Cohesiveness in the Tree Ramsey Theorem for Pairs

Wei Li (National University of Singapore, SG)

Joint work of C. T. Chong, Wei Li, Wei Wang, Yue Yang
Main reference C. T. Chong, Wei Li, Wei Wang, Yue Yang: "On the strength of Ramsey's theorem for trees", preprint.

In this talk, we present a version of cohesiveness in the setting of the tree Ramsey Theorem. We prove that the cohesiveness for trees is Pi^1_1 conservative over $P\Sigma_1 + B\Sigma_2$. It is a joint work with C. T. Chong, Lu Liu and Yue Yang.

3.14 Using a Weihrauch degree finitely many times

Arno Pauly (Swansea University, GB)

The closure operator $^\diamond$ introduced in [3] captures the idea of using a Weihrauch degree finitely many times, without any requirements on a priorily bounding the number of uses:

▶ **Definition 1.** $f^\diamond$ has instances

- A register machine program M using f as a primitive operation (could be non-deterministic!)
- An input x for M on which M halts

and provides $M(x)$ as solutions.

It is intimately linked to the generalized Weihrauch reducibility by Hirschfeldt and Jockusch:

▶ **Observation 1.** $f \leq_W g^\diamond$ iff $f \leq_{gW} g$.

The following example (jww Kazuto Yoshimura) shows that it does not even have to hold that the number of oracles uses depends on the input – it can depend on intermediate results to the oracel calls:

▶ **Example 2.** Let $(q_i)_{i\in\omega}$ be strongly Turing-incomparable. Define F by $F(0^\omega) = \{iq_i \mid i \in \omega\}$, $F(q_{i+1}) = q_i$. Then $q_0 \leq_W F^\diamond$, but we have no bounds for the *run-time*.

Various classifications or stability results for $^\diamond$ have been proven. We shall list some of those:

▶ **Theorem 3.**
- $\mathrm{LPO}^\diamond \equiv_W \mathrm{C}_\mathbb{N}$ *(Neumann & Pauly [3])*
- $\mathrm{C}^\diamond_{\{0,1\}^\omega} \equiv_W \mathrm{C}_{\{0,1\}^\omega}$, $\mathrm{C}^\diamond_\mathbb{R} \equiv_W \mathrm{C}_\mathbb{R}$, $\mathrm{C}^\diamond_{\omega^\omega} \equiv_W \mathrm{C}_{\omega^\omega}$
- $\mathrm{Sort}^\diamond \equiv_W \Pi^0_2\mathrm{C}_\mathbb{N}$ *(Gassner, P. & Steinberg)*
- $(\Sigma^0_\alpha\mathrm{LPO})^\diamond \equiv_W \Pi^0_\alpha\mathrm{C}_\mathbb{N}$ *(Brattka, Gherardi, Hölzl, Nobrega & P.)*
- $\mathrm{C}_{\{0,1\}^\omega,\sharp<\infty} \equiv_W \mathrm{C}^\diamond_{\{0,1\}^\omega<\infty}$ *(Pauly & Tsuiki [1])*
- $\mathrm{C}^\diamond_{\{0,1\}^\omega,\sharp\leq 2} \equiv_W \coprod_{n\in\mathbb{N}} \mathrm{C}_{\{0,1\}^\omega,\sharp\leq n} \equiv_W \mathrm{C}^*_{\{0,1\}^\omega,\sharp\leq 2}$ *(Pauly & Tsuiki [1])*
- $\mathrm{C}^\diamond_{\{0,1\}^\omega,\sharp=2} \equiv_W \coprod_{n\in\mathbb{N}} \mathrm{C}_{\{0,1\}^\omega,\sharp=n} \equiv_W \mathrm{C}^*_{\{0,1\}^\omega,\sharp=2}$ *(Pauly & Tsuiki [1])*

For the last three items, we recall:

▶ **Definition 4** (Le Roux & P. [2]; Tsuiki & P. [1]). Let $\mathrm{C}_{\{0,1\}^\omega,\sharp=n}$, $\mathrm{C}_{\{0,1\}^\omega,\sharp\leq n}$, $\mathrm{C}_{\{0,1\}^\omega,\sharp<\infty}$ be closed choice on 2^ω restricted to sets of cardinality n, at most n, or finite.

References

1 Denis Hirschfeldt & Carl Jockusch: On notions of computability-theoretic reduction between Π^1_2-principles. Journal of Mathematical Logic 2016.

2 Stéphane Le Roux & Arno Pauly: Finite choice, convex choice and finding roots. Logical Methods in Computer Science 2015.

3 Eike Neumann & Arno Pauly: A topological view on algebraic computations models. Journal of Complexity 2018.

4 Arno Pauly & Hideki Tsuiki: T^ω-representations of compact sets. arXiv:1604.00258

3.15 Overt choice on CoPolish spaces

Matthias Schröder (Universität der Bundeswehr – München, DE)

Choice principles are cornerstones in the Weihrauch lattice, as many important Weihrauch degrees are characterised by a choice problem. Overt choice means the computational task of picking a point in a closed set given by positive information. From Computable Analysis we know that overt choice is computable on computable Polish spaces.

We show that overt choice is discontinuous on CoPolish spaces like the vector space of polynomials or the space of tempered distributions. The discontinuity is caused by the fact that these spaces are not Frechet-Urysohn spaces. There is a minimal non-Frechet-Urysohn CoPolish space $Smin$ which embeds as a closed subspace into every other such space. Overt choice on $Smin$ turns out to be Weihrauch equivalent to LPO.

On the positive side, we show that overt-compact choice on CoPolish spaces is continuous. It is even computable, if the CoPolish space meets some reasonable effectivity conditions. Finally we present a Choice Elimination Theorem for compact choice on CoPolish spaces.

3.16 Q-Wadge degrees as free structures

Victor Selivanov (A. P. Ershov Institute – Novosibirsk, RU)

Based on ideas, notions and results of P Hertling, J. Duparc and V. Selivanov, T. Kihara and A. Montalban have recently characterized up to isomorphism the structure W_Q of Wadge degrees of Borel Q-partitions of the Baire space, for every countable better quasiorder Q. The characterization is in terms of the so called h-quasiorder on suitably iterated Q-labeled countable well founded forests. Since the corresponding precise definitions are rather long and technical, we attempt here to find a clear shorter characterization.

To achieve this goal, we formulate some easy axioms for a theory T in a language expanding the language of sigma-semilattices. Then we show that many initial segments of W_Q (including W_Q itself) are (reducts of) free structures of suitable subtheories of T. Informally, in this way we obtain a kind of axiomatizations for the initial segments of W_Q.

3.17 Polynomial-time Weihrauch reductions

Florian Steinberg (INRIA Sophia Antipolis, FR)

Main reference Florian Steinberg: "Computational Complexity Theory for Advanced Function Spaces in Analysis", PhD thesis 2017.

The complexity of operators on the real functions has been a topic of interest for some time (see [1]). However, until fairly recently, complexity theoretical considerations on continuous strutures where limited by the framework. While complexity theory for function on the the real numbers worked reasonably well, many function spaces were known to be "to broad" to be captured. Thus, complexity considerations about operators were confined to be point-wise. Nontheless, interesting results were proven in this setting: For instance that the integration operator preserves the class of polynomial-time computable functions if and only if FP = #P.

This changed, when in 2012 Kawamura and Cook introduced a framework for complexity theory for operators from analysis that allowed for a uniform treatment of operators on real functions by relying on type-two complexity theory. The added uniformity requirement often removes the dependence of results on separation results about complexity classes. For instance, within Kawamura and Cooks framework, it is possible to prove that the integration operator is not polynomial-time computable. In his PhD Thesis and subsequent work, Kawamura introduced a corresponding notion of reducibility and provides some examples of uniformizations. This reducibility is a polynomial-time version of Weihrauch reducibility and can be used to gain further insight in the properties of the operators that are related to separation of complexity classes.

We give a short introduction to the framework of Kawamura and Cook and an overview over what is known about polynomial-time Weihrauch reducibility so far. It turns out that there are some interesting differences to non resource-restricted Weihrauch reducibility. For instance, strong Weihrauch reducibility may fail not only for information theoretic reasons but also because the operator to reduce to forgets about the sizes of instances. For illustration we take a closer look at a uniformization of one of Friedmann and Ko's results about integration of real functions that was part of the authors PhD project.

References

1 Ker-I Ko. 1991. *Complexity Theory of Real Functions*. Birkhauser Boston Inc., Cambridge, MA, USA.

2 Akitoshi Kawamura and Stephen Cook. 2012. *Complexity Theory for Operators in Analysis*. ACM Trans. Comput. Theory 4, 2, Article 5 (May 2012), 24 pages.

3.18 Proof-theoretic characterization of Weihrauch reducibility

Patrick Uftring (TU Darmstadt, DE)

First, we discuss some counterexamples to the theorems of the article [2] by Rutger Kuyper about the characterization of Weihrauch reducibility in RCA_0.

Secondly, we present some results of our own: Affine logic is a refinement of classical logic that restricts contraction. We define affine Peano arithmetic in all finite types in order to characterize different formalizations of Weihrauch reducibility for different classes of total problems. We do this by combining a variation of Gödel's Dialectica interpretation for classical affine logic due to Masaru Shirahata [3], a functional interpretation by Benno van den Berg, Eyvind Briseid, and Pavol Safarik for nonstandard arithmetic [1], and a hereditarily defined notion of computability for higher types derived from associates.

References

1 Benno van den Berg, Eyvind Briseid, and Pavol Safarik. "A functional interpretation for nonstandard arithmetic". Annals of Pure and Applied Logic 163.12 (2012), pp. 1962–1994.

2 Rutger Kuyper. "On Weihrauch reducibility and intuitionistic reverse mathematics". The Journal of Symbolic Logic 82.4 (2017), pp. 1438–1458.

3 Masaru Shirahata. "The Dialectica interpretation of first-order classical affine logic". Theory and Applications of Categories 17.4 (2006), pp. 49–79.

3.19 Computable planar curves intersect in a computable point

Klaus Weihrauch (FernUniversität in Hagen, DE)

Main reference Klaus Weihrauch: "Computable planar paths intersect in a computable point", CoRR, arXiv:1708.07460v2, 2017.
URL https://arxiv.org/abs/1708.07460

Consider two paths $f, g : [0;1] \to [0;1]^2$ in the unit square such that $f(0) = (0,0)$, $f(1) = (1,1)$, $g(0) = (0,1)$ and $g(1) = (1,0)$. By continuity of f and g there is a point of intersection. We prove that there is a computable point of intersection if f and g are computable.

The article has been accepted by the journal "Computability" and will appear soon.

4 Solved questions

4.1 Joins in the strong Weihrauch degrees

Damir D. Dzhafarov (University of Connecticut – Storrs, US)

The Weihrauch degrees and strong Weihrauch degrees are partially ordered structures representing degrees of unsolvability of various mathematical problems. Their study has been widely applied in computable analysis, complexity theory, and more recently, also in computable combinatorics. We answer an open question about the algebraic structure of the strong Weihrauch degrees, by exhibiting a join operation that turns these degrees into a lattice. Previously, the strong Weihrauch degrees were only known to form a lower semi-lattice. We then show that unlike the Weihrauch degrees, which are known to form a distributive lattice, the lattice of strong Weihrauch degrees is not distributive. Therefore, the two structures are not isomorphic.

4.2 Separating products of Weihrauch degrees

Takayuki Kihara (Nagoya University, JP)

Joint work of Takayuki Kihara, Arno Pauly
Main reference Takayuki Kihara, Arno Pauly: "Dividing by Zero – How Bad Is It, Really?", in Proc. of the 41st International Symposium on Mathematical Foundations of Computer Science, MFCS 2016, August 22-26, 2016 – Kraków, Poland, LIPIcs, Vol. 58, pp. 58:1–58:14, Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2016.
URL http://dx.doi.org/10.4230/LIPIcs.MFCS.2016.58
Main reference Takayuki Kihara, Arno Pauly: "Finite choice, convex choice and sorting". Preprint.

We show that the compositional product of LLPO and AoUC is not Weihrauch reducible to finite parallelization of AoUC [1], and the the compositional product of IVT and AoUC is not Weihrauch reducible to any finite dimensional convex choice [2]. This solves two open problems raised at a recent Dagstuhl meeting 15392 on Weihrauch reducibility.

References

1 Takayuki Kihara and Arno Pauly. *Dividing by zero – how bad is it, really?*. In Proceedings of MFCS 2016, Leibniz International Proceedings in Informatics 58 (2016), pp. 58:1-58:14.

2 Takayuki Kihara and Arno Pauly. *Finite choice, convex choice and sorting*. preprint.

4.3 ATR_0 in the Weihrauch lattice

Alberto Marcone (University of Udine, IT)

Joint work of Takayuki Kihara, Alberto Marcone, Arne Pauly

This is a survey on the progress made since the previous Dagstuhl workshop on the study within the Weihrauch lattice of problems arising from statement lying at the upper levels of the reverse mathematics hierarchy. In particular, we consider statements equivalent, or closely related, to ATR_0, such as various set-existence axioms, comparability of well-orders, the perfect tree theorem, and open determinacy. The Weihrauch degrees appearing in this research include Unique Choice and Choice on Baire space.

The results will be included in a joint paper with Takayuki Kihara and Arno Pauly.

References

1 Takayuki Kihara, Alberto Marcone, and Arno Pauly. *Searching for an analogue of* ATR_0 *in the Weihrauch lattice*. In preparation.

4.4 RT^2_2 compared to the product of SRT^2_2 and COH

Ludovic Patey (University Claude Bernard – Lyon, FR)

Joint work of Damir D. Dzhafarov, Jun Le Goh, Denis R. Hirschfeldt, Ludovic Patey, Arno Pauly
Main reference Damir D. Dzhafarov, Jun Le Goh, Denis R. Hirschfeldt, Ludovic Patey, Arno Pauly: "Ramsey's theorem and pro ducts in the Weihrauch degrees", CoRR (2018), arXiv:1804.10968
URL https://arxiv.org/abs/1804.10968

Ramsey's theorem for pairs and two colors (RT^2_2) asserts that every 2-coloring of $[\mathbb{N}]^2$ admits an infinite monochromatic set. RT^2_2 can be decomposed into a stable version (SRT^2_2) and the cohesiveness principle (COH). From the viewpoint of Weihrauch reducibility, RT^2_2 is a consequence of the compositional product of SRT^2_2 and COH and implies their coproduct. In a previous Dagstuhl seminar, it was asked which reversals hold.

In this talk, we present a complete overview of the question and show that none of the reversal holds. In particular, we prove that the cartesian product of SRT^2_2 and COH is not Weihrauch reducible to RT^2_2.

This is a joint work with Damir Dzhafarov, Jun Le Goh, Denis Hirschfeldt and Arno Pauly.

4.5 Grouping principle

Keita Yokoyama (JAIST – Ishikawa, JP) and Ludovic Patey (University Claude Bernard – Lyon, FR)

Main reference Ludovic Patey and Keita Yokoyama: "The proof-theoretic strength of Ramsey's theorem for pairs and two colors, Advances in Mathematics", 330:1034–1070, 2018.
URL https://doi.org/10.1016/j.aim.2018.03.035

Grouping principle is a technical combinatorial statement which is a direct consequence of Ramsey's theorem. In the previous seminar (Dagstuhl seminar 15392), Yokoyama posed a question "what is the reverse mathematical strength of the grouping principle for pairs and two colors?" Patey answered this quesiton by showing that any computable instance of the stable version of the grouping principle for pairs admits has a low solution.

5 Open problems

5.1 Density and minimality properties of the Weihrauch lattice

Vasco Brattka (Universität der Bundeswehr – München, DE)

This open problem is related to lattice theoretic properties of the Weihrauch lattice and its variants. These questions apply to the Weihrauch lattice itself, to the strong Weihrauch lattice, to the parallelized Weihrauch lattice, the parallelized total Weihrauch lattice and other variants:

1. What can be said about density properties of the corresponding lattice?
2. Are there regions where the lattice is dense and others where it is not? Can those be classified?
3. Are there minimal pairs or atoms?

Basically nothing is known about the answers to such questions!

5.2 Ramsey's theorem: products versus colors

Vasco Brattka (Universität der Bundeswehr – München, DE)

We consider Ramsey's theorem for a fixed cardinality n and k colors. It is easy to see that the m–fold product of Ramsey's theorem for k colors is strongly Weihrauch reducible to a single instance with k^m colors (all for the fixed cardinality n) [31, Corollary 3.18 (1)]. This means that colors can make up for products. Does the converse hold true, i.e., can products make up for colors? More precisely, is there a number m for each k, such that Ramsey's theorem for k colors is Weihrauch reducible to the m–fold product of Ramsey's theorem for only 2 colors (all for the fixed cardinality n)? (See also [31, Question 3.22].) The answer is yes for cardinality $n = 1$ [31, Proposition 3.23], but not known for higher cardinalities $n \geq 2$.

References

1 Vasco Brattka and Tahina Rakotoniaina. *On the Uniform Computational Content of Ramsey's Theorem*, Journal of Symbolic Logic 82:4 (2017) 1278-1316, see also https://arxiv.org/pdf/1508.00471.pdf

5.3 Weihrauch strength of countable well-orderings

Jeffry L. Hirst (Appalachian State University – Boone, US)

What is the Weihrauch strength of various statements about countable well-orderings? In the reverse mathematics setting, they tend to clump into two groups, one at the ACA_0 level and the other at ATR_0. Do they separate in the Weihrauch hierarchy?

Possibly useful resources include the survey of ordinal arithmetic in Reverse Mathematics 2001 [1] and Sierpinski's text, Cardinal and Ordinal Numbers [2]. Also see the related work by Jun Le Goh and by Alberto Marcone and his affiliates.
A small subproject: Examine statements related to indecomposable ordinals.

- Weak comparability of indecomposable well-orderings.
- If α is well-ordered, then ω^α is well-ordered. (Consider the contrapositive to formulate this as a Weihrauch problem.)
- If α is indecomposable, then there is a β such that $\alpha = \omega^\beta$. (Here, the equality could indicate weak comparability or strong comparability.)

In the subproject, the reverse mathematical analysis of the statements has already been completed. One could also select a previously unanalyzed statement from Sierpinski [2] and do both the reverse mathematical analysis and the Weihrauch analysis.

References

1 Jeffry L. Hirst *A survey of the reverse mathematics of ordinal arithmetic*. In: Reverse Mathematics 2001, Lect. Notes Log., Volume 21, editor: Stephen G. Simpson, Assoc. Symbol. Logic, La Jolla, CA, USA, pages 222-224, 2005.

2 Wacław Sierpiński *Cardinal and ordinal arithmetic, Second revised edition*, Monografie Matematyczne, Volume 34, Państowe Wydawnictwo Naukowe, Warsaw, 1965.

5.4 Some questions around Weihrauch counterparts of ATR

Takayuki Kihara (Nagoya University, JP)

Main reference Paul-Elliot Anglès D'Auriac and Takayuki Kihara: "A comparison of various analytic choice principles", Preprint.

Goh introduced the two-sided version ATR_2 of arithmetical transfinite recursion, and Anglès D'Auriac and Kihara [1] introduced its variant $ATR_{2'}$ which is shown to be arithmetically Weihrauch equivalent to the Σ^1_1-choice on Cantor space.

Q1. Is $ATR_{2'}$ arithmetically Weihrauch equivalent to ATR_2?

Anglès D'Auriac and Kihara [1] showed that the Σ^1_1-choice on Baire space is not Weihrauch reducible to the parallelization of the Σ^1_1-choice on the natural numbers.

Q2. Is the Σ^1_1-choice on Baire space hyperarithmetically Weihrauch reducible to the parallelization of the Σ^1_1-choice on the natural numbers?

References

1 Paul-Elliot Anglès D'Auriac and Takayuki Kihara. *A comparison of various analytic choice principles*. preprint.

5.5 Two open questions from Dagstuhl Seminar 18361

Carl Mummert (Marshall University – Huntington, US)

These two questions concern the Weihrauch degrees of problems in algebra. The first concerns vector spaces. The elements of a countable vector space over $\mathbb{Q}$ can be identified with elements of $\mathbb{N}$, so that the elementary diagram can be encoded canonically as an element of $2^{\mathbb{N}}$. We can use this representation to ask about the degrees of problems in linear algebra. For example, the problem of producing a basis for a countable vector space over $\mathbb{Q}$ has Weihrauch degree $\widehat{\mathsf{LPO}}$, and in the setting of reverse mathematics the analogous principle of second order arithmetic is equivalent to ACA_0 over RCA_0. The first question relates to the problem of finding a proper finite dimensional subspace of a countable vector space over $\mathbb{Q}$.

> **Problem:** Let $P\colon \subseteq 2^{\mathbb{N}} \rightrightarrows 2^{\mathbb{N}}$ be the partial multifuction that, given the atomic diagram of an infinite dimensional vector space over $\mathbb{Q}$, returns the characteristic function of a finite dimensional nonzero subspace of the vector space. What is the Weihrauch degree of P?

Downey, Hirschfeldt, Kach, Lempp, Mileti, and Montalbán [1] proved that the principle of second order arithmetic analogous to P is equivalent to ACA_0 over RCA_0. Their proof has an interesting nonuniformity, as it relies on the ability to choose a basis for the finite dimensional subspace. It follows from their results that $\mathsf{WKL} \leq_W P \leq_W \widehat{\mathsf{LPO}}$. We suspect $P \equiv_W \widehat{\mathsf{LPO}}$, but a new proof method seems to be needed.

The second problem comes from group theory. It is a classical fact that every group with more than 2 elements has a nontrivial automorphism. We represent countably infinite groups by identifying their set of elements with $\mathbb{N}$, so that their elementary diagrams can be viewed as elements of $2^{\mathbb{N}}$. There is no loss of generality in assuming the identity element is identified with $0 \in \mathbb{N}$.

> **Problem:** Let $A\colon \subseteq 2^{\mathbb{N}} \rightrightarrows \mathbb{N}^{\mathbb{N}}$ be the partial multifunction that, given the atomic diagram of a countably infinite group, produces a nontrivial automorphism of the group. What is the Weihrauch degree of A?

The known upper bound is $A \leq_W \mathsf{LPO} \times \mathsf{LPO}$. The two particular questions that LPO is used to answer are whether the group is abelian and whether every element has order 2. In particular, every computable countably infinite group has a computable nontrivial automorphism. If the Weihrauch degree of A is nontrivial, this provides another example of the importance of weak choice principles.

References

1 Downey, Rodney G. and Hirschfeldt, Denis R. and Kach, Asher M. and Lempp, Steffen and Mileti, Joseph R. and Montalbán, Antonio. Subspaces of computable vector spaces. *J. Algebra*. 314(2):888–894, 2007.

5.6 Characterizing the diamond-operator

Arno Pauly (Swansea University, GB)

The $^\diamond$-operator in the Weihrauch lattice captures the idea of making finitely many calls to an oracle available, without any a priori known bound on the number of calls. See the abstract "Using a Weihrauch degree finitely many times" abstract for details. It is clear that if $f \equiv_W f^\diamond$, then $1 \leq_W f$ and $f \equiv_W f \star f$. Our question is whether the converse holds:

Does $1 \leq_W f$ and $f \equiv_W f \star f$ imply $f \equiv_W f^\diamond$?

During the seminar, Linda Brown Westrick obtained a positive answer to this question.

5.7 Compact Hausdorff spaces are regular

Arno Pauly (Swansea University, GB)

It is a well-known result from topology that compact Hausdorff spaces are regular. The traditional proof proceeds as follows: We are given $x \in \mathbf{X}$ and $A \in \mathcal{A}(\mathbf{X})$ with $x \notin A$. For each $y \in A$ there are disjoint opens $U_y \ni x$ and $V_y \ni y$, since $\mathbf{X}$ is Hausdorff. Consider the open cover $A \subseteq \bigcup_{y \in A} V_y$. By compactness of $\mathbf{X}$, there exists some finite $I \subseteq A$ such that already $A \subseteq \bigcup_{y \in I} V_y$. Now $\bigcup_{y \in I} V_y$ and $\bigcap_{y \in I} U_y$ are disjoint open sets separating x and A.

In computable topology, however, this argument does not go through. In order to obtain$\bigcup_{y \in A} V_y$ as an open set, we would require A as an overt set, not merely as a closed set. Restricted to countably-based spaces, a different approach was shown to work in [1]. Here, we ask whether the statement holds in general:

Is every computably Hausdorff computably compact represented spaces already computably regular?

References

1 Arno Pauly & Hideki Tsuiki. *Computable dyadic subbases and $\mathbf{T}^\omega$-representations of compact sets.* arXiv https://arxiv.org/abs/1604.00258

2 Arno Pauly. *On the topological aspects of the theory of represented spaces.* Computability 5(2). 2016. doi 10.3233/COM-150049

5.8 Characterization of overt choice on maximal CoPolish spaces

Matthias Schröder (Universität der Bundeswehr – München, DE)

It is known that there exist maximal CoPolish spaces X in the sense that any other CoPolish space is homeomorphic to a closed subspace of X. A CoPolish space is defined to be the direct limit of an increasing sequence of compact metric spaces. One example of a maximal CoPolish space is the Hilbert space l_2 equipped with the sequentialization of the weak* topology on l_2. Overt choice is the problem of picking a point in a closed subset given with positive information.

Question: Characterize the Weihrauch degree of overt choice $V(l_2)$ on l_2.

Note that overt choice on any CoPolish space is continuously Weihrauch reducible to $V(l_2)$ due to the maximality property.

5.9 Minimal continuous Weihrauch degrees

Matthias Schröder (Universität der Bundeswehr – München, DE)

Let $f \neq 0$ be any multifunction, where 0 denotes the nowhere defined problem.

Question: Does there exist a multifunction $g \neq 0$ such that g is strictly below f in the continuous Weihrauch lattice?

5.10 When can one step function Weihrauch compute another?

Linda Brown Westrick (Pennsylvania State University – University Park, US)

Joint work of Adam Day, Rod Downey, Takayuki Kihara, Linda Westrick

Let $\leq$ denote the lexicographic order on Cantor space. For $A \in 2^\omega$, define the step function $s_A : 2^\omega \to 2$ to be the characteristic function of $\{X \in 2^\omega : A \leq X\}$.

Question: Characterize the pairs (A, B) for which $s_A \leq_W s_B$.

The little that is known about this is strange. If B is computable and s_B is discontinuous, then $s_A \leq_W s_B$ if and only if A is left-c.e. But if B is not computable and $s_A \leq_W s_B$, then A and B are Turing equivalent.

6 Bibliography on Weihrauch Complexity

For an always up-to-date version of this bibliography see http://cca-net.de/publications/weibib.php.

References

1 Measuring the Complexity of Computational Content (Dagstuhl Seminar 15392). Technical Report 9, Dagstuhl, Germany, 2016.

2 Nathanael L. Ackerman, Cameron E. Freer, and Daniel M. Roy. On computability and disintegration. *Mathematical Structures in Computer Science*, 27(8):1287–1314, 2017.

3 Eric P. Astor, Damir D. Dzhafarov, Reed Solomon, and Jacob Suggs. The uniform content of partial and linear orders. *Annals of Pure and Applied Logic*, 168(6):1153 – 1171, 2017.

4 Laurent Bienvenu and Rutger Kuyper. Parallel and serial jumps of Weak Weak König's Lemma. In Adam Day, Michael Fellows, Noam Greenberg, Bakhadyr Khoussainov, Alexander Melnikov, and Frances Rosamond, editors, *Computability and Complexity: Essays Dedicated to Rodney G. Downey on the Occasion of His 60th Birthday*, volume 10010 of *Lecture Notes in Computer Science*, pages 201–217. Springer, Cham, 2017.

5 Ana de Almeida Gabriel Vieira Borges. *On the herbrandised interpretation for nonstandard arithmetic.* PhD thesis, Instituto Superior Técnico, Lisbon, 2016. MSc thesis.

6 Vasco Brattka. *Grade der Nichtstetigkeit in der Analysis.* PhD thesis, Fachbereich Informatik, FernUniversität Hagen, 1993. Diplomarbeit.

7 Vasco Brattka. Computable invariance. In Tao Jiang and D.T. Lee, editors, *Computing and Combinatorics*, volume 1276 of *Lecture Notes in Computer Science*, pages 146–155, Berlin, 1997. Springer. Third Annual Conference, COCOON'97, Shanghai, China, August 1997.

8 Vasco Brattka. Computable invariance. *Theoretical Computer Science*, 210:3–20, 1999.

9 Vasco Brattka. Effective Borel measurability and reducibility of functions. *Mathematical Logic Quarterly*, 51(1):19–44, 2005.

10 Vasco Brattka. Computability and analysis, a historical approach. In Arnold Beckmann, Laurent Bienvenu, and Nataša Jonoska, editors, *Pursuit of the Universal*, volume 9709 of *Lecture Notes in Computer Science*, pages 45–57, Switzerland, 2016. Springer. 12th Conference on Computability in Europe, CiE 2016, Paris, France, June 27 - July 1, 2016.

11 Vasco Brattka, Andrea Cettolo, Guido Gherardi, Alberto Marcone, and Matthias Schröder. Addendum to: "The Bolzano-Weierstrass theorem is the jump of weak Kőnig's lemma". *Annals of Pure and Applied Logic*, 168(8):1605–1608, 2017.

12 Vasco Brattka, Matthew de Brecht, and Arno Pauly. Closed choice and a uniform low basis theorem. *Annals of Pure and Applied Logic*, 163:986–1008, 2012.

13 Vasco Brattka and Guido Gherardi. Borel complexity of topological operations on computable metric spaces. *Journal of Logic and Computation*, 19(1):45–76, 2009.

14 Vasco Brattka and Guido Gherardi. Effective choice and boundedness principles in computable analysis. In Andrej Bauer, Peter Hertling, and Ker-I Ko, editors, *CCA 2009, Proceedings of the Sixth International Conference on Computability and Complexity in Analysis*, pages 95–106, Schloss Dagstuhl, Germany, 2009. Leibniz-Zentrum für Informatik.

15 Vasco Brattka and Guido Gherardi. Weihrauch degrees, omniscience principles and weak computability. In Andrej Bauer, Peter Hertling, and Ker-I Ko, editors, *CCA 2009, Proceedings of the Sixth International Conference on Computability and Complexity in Analysis*, pages 83–94, Schloss Dagstuhl, Germany, 2009. Leibniz-Zentrum für Informatik.

16 Vasco Brattka and Guido Gherardi. Effective choice and boundedness principles in computable analysis. *The Bulletin of Symbolic Logic*, 17(1):73–117, 2011.

17 Vasco Brattka and Guido Gherardi. Weihrauch degrees, omniscience principles and weak computability. *The Journal of Symbolic Logic*, 76(1):143–176, 2011.

18 Vasco Brattka, Guido Gherardi, and Rupert Hölzl. Las Vegas computability and algorithmic randomness. In Ernst W. Mayr and Nicolas Ollinger, editors, *32nd International*

Symposium on Theoretical Aspects of Computer Science (STACS 2015), volume 30 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 130–142, Dagstuhl, Germany, 2015. Schloss Dagstuhl–Leibniz-Zentrum für Informatik.

19 Vasco Brattka, Guido Gherardi, and Rupert Hölzl. Probabilistic computability and choice. *Information and Computation*, 242:249–286, 2015.

20 Vasco Brattka, Guido Gherardi, Rupert Hölzl, and Arno Pauly. The Vitali covering theorem in the Weihrauch lattice. In Adam Day, Michael Fellows, Noam Greenberg, Bakhadyr Khoussainov, Alexander Melnikov, and Frances Rosamond, editors, *Computability and Complexity: Essays Dedicated to Rodney G. Downey on the Occasion of His 60th Birthday*, volume 10010 of *Lecture Notes in Computer Science*, pages 188–200. Springer, Cham, 2017.

21 Vasco Brattka, Guido Gherardi, and Alberto Marcone. The Bolzano-Weierstrass theorem is the jump of weak Kőnig's lemma. *Annals of Pure and Applied Logic*, 163:623–655, 2012.

22 Vasco Brattka, Guido Gherardi, and Arno Pauly. Weihrauch complexity in computable analysis. arXiv 1707.03202, 2017.

23 Vasco Brattka, Matthew Hendtlass, and Alexander P. Kreuzer. On the uniform computational content of computability theory. *Theory of Computing Systems*, 61(4):1376–1426, 2017.

24 Vasco Brattka, Matthew Hendtlass, and Alexander P. Kreuzer. On the uniform computational content of the Baire category theorem. *Notre Dame Journal of Formal Logic*, 59(4):605–636, 2018.

25 Vasco Brattka, Rupert Hölzl, and Rutger Kuyper. Monte Carlo computability. In Heribert Vollmer and Brigitte Vallée, editors, *34th Symposium on Theoretical Aspects of Computer Science (STACS 2017)*, volume 66 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 17:1–17:14, Dagstuhl, Germany, 2017. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

26 Vasco Brattka, Stéphane Le Roux, Joseph S. Miller, and Arno Pauly. The Brouwer fixed point theorem revisited. In Arnold Beckmann, Laurent Bienvenu, and Nataša Jonoska, editors, *Pursuit of the Universal*, volume 9709 of *Lecture Notes in Computer Science*, pages 58–67, Switzerland, 2016. Springer. 12th Conference on Computability in Europe, CiE 2016, Paris, France, June 27 - July 1, 2016.

27 Vasco Brattka, Stéphane Le Roux, Joseph S. Miller, and Arno Pauly. Connected choice and the Brouwer fixed point theorem. *Journal of Mathematical Logic*, (accepted for publication), 2018.

28 Vasco Brattka, Stéphane Le Roux, and Arno Pauly. On the computational content of the Brouwer Fixed Point Theorem. In S. Barry Cooper, Anuj Dawar, and Benedikt Löwe, editors, *How the World Computes*, volume 7318 of *Lecture Notes in Computer Science*, pages 57–67, Berlin, 2012. Springer. Turing Centenary Conference and 8th Conference on Computability in Europe, CiE 2012, Cambridge, UK, June 2012.

29 Vasco Brattka and Arno Pauly. Computation with advice. In Xizhong Zheng and Ning Zhong, editors, *CCA 2010, Proceedings of the Seventh International Conference on Computability and Complexity in Analysis*, Electronic Proceedings in Theoretical Computer Science, pages 41–55, 2010.

30 Vasco Brattka and Arno Pauly. On the algebraic structure of Weihrauch degrees. *Logical Methods in Computer Science*, 14(4:4):1–36, 2018.

31 Vasco Brattka and Tahina Rakotoniaina. On the uniform computational content of Ramsey's theorem. *Journal of Symbolic Logic*, 82(4):1278–1316, 2017.

32 Merlin Carl. Generalized effective reducibility. arXiv 1601.01899, 2016.

33 Merlin Carl. Generalized effective reducibility. In Arnold Beckmann, Laurent Bienvenu, and Nataša Jonoska, editors, *Pursuit of the Universal*, volume 9709 of *Lecture Notes in*

Computer Science, pages 225–233, Switzerland, 2016. Springer. 12th Conference on Computability in Europe, CiE 2016, Paris, France, June 27 - July 1, 2016.

34 Raphaël Carroy. *Functions of the first Baire class.* PhD thesis, University of Lausanne and University Paris 7, 2013.

35 Raphaël Carroy. A quasi-order on continuous functions. *Journal of Symbolic Logic*, 78(2):663–648, 2013.

36 Matthew de Brecht. Levels of discontinuity, limit-computability, and jump operators. In Vasco Brattka, Hannes Diener, and Dieter Spreen, editors, *Logic, Computation, Hierarchies*, Ontos Mathematical Logic, pages 93–122. Walter de Gruyter, Boston, 2014.

37 François G. Dorais, Damir D. Dzhafarov, Jeffry L. Hirst, Joseph R. Mileti, and Paul Shafer. On uniform relationships between combinatorial problems. *Transactions of the American Mathematical Society*, 368(2):1321–1359, 2016.

38 Damir D. Dzhafarov. Cohesive avoidance and strong reductions. *Proceedings of the American Mathematical Society*, 143(2):869–876, 2015.

39 Damir D. Dzhafarov. Strong reductions between combinatorial principles. *Journal of Symbolic Logic*, 81(4):1405–1431, 2016.

40 Damir D. Dzhafarov. Joins in the strong Weihrauch degrees. *Mathematical Research Letters*, (to appear), 2018.

41 Damir D. Dzhafarov, Jun Le Goh, Denis R. Hirschfeldt, Ludovic Patey, and Arno Pauly. Ramsey's theorem and products in the Weihrauch degrees. arXiv 1804.10968, 2018.

42 Damir D. Dzhafarov, Ludovic Patey, Reed Solomon, and Linda Brown Westrick. Ramsey's theorem for singletons and strong computable reducibility. *Proceedings of the American Mathematical Society*, 145, 2017.

43 Makoto Fujiwara, Kojiro Higuchi, and Takayuki Kihara. On the strength of marriage theorems and uniformity. *Mathematical Logic Quarterly*, 60(3):136–153, 2014.

44 Lorenzo Galeotti and Hugo Nobrega. Towards computable analysis on the generalized real line. In Jarkko Kari, Florin Manea, and Ion Petre, editors, *Unveiling Dynamics and Complexity*, volume 10307 of *Lecture Notes in Computer Science*, pages 246–257, Cham, 2017. Springer. 13th Conference on Computability in Europe, CiE 2017, Turku, Finland, June 12-16, 2017.

45 Guido Gherardi. An analysis of the lemmas of Urysohn and Urysohn-Tietze according to effective Borel measurability. In A. Beckmann, U. Berger, B. Löwe, and J.V. Tucker, editors, *Logical Approaches to Computational Barriers*, volume 3988 of *Lecture Notes in Computer Science*, pages 199–208, Berlin, 2006. Springer. Second Conference on Computability in Europe, CiE 2006, Swansea, UK, June 30-July 5, 2006.

46 Guido Gherardi. Effective Borel degrees of some topological functions. *Mathematical Logic Quarterly*, 52(6):625–642, 2006.

47 Guido Gherardi. *Some Results in Computable Analysis and Effective Borel Measurability.* PhD thesis, University of Siena, Department of Mathematics and Computer Science, Siena, 2006.

48 Guido Gherardi and Alberto Marcone. How incomputable is the separable Hahn-Banach theorem? In Vasco Brattka, Ruth Dillhage, Tanja Grubba, and Angela Klutsch, editors, *CCA 2008, Fifth International Conference on Computability and Complexity in Analysis*, volume 221 of *Electronic Notes in Theoretical Computer Science*, pages 85–102. Elsevier, 2008. CCA 2008, Fifth International Conference, Hagen, Germany, August 21–24, 2008.

49 Guido Gherardi and Alberto Marcone. How incomputable is the separable Hahn-Banach theorem? *Notre Dame Journal of Formal Logic*, 50(4):393–425, 2009.

50 Guido Gherardi, Alberto Marcone, and Arno Pauly. Projection operators in the Weihrauch lattice. arXiv 1805.12026, 2018.

51 Kirill Gura, Jeffry L. Hirst, and Carl Mummert. On the existence of a connected component of a graph. *Computability*, 4(2):103–117, 2015.

52 Peter Hertling. Stetige Reduzierbarkeit auf Σ^ω von Funktionen mit zweielementigem Bild und von zweistetigen Funktionen mit diskretem Bild. Informatik Berichte 153, FernUniversität Hagen, Hagen, December 1993.

53 Peter Hertling. A topological complexity hierarchy of functions with finite range. Technical Report 223, Centre de recerca matematica, Institut d'estudis catalans, Barcelona, Barcelona, October 1993. Workshop on Continuous Algorithms and Complexity, Barcelona, October, 1993.

54 Peter Hertling. Topologische Komplexitätsgrade von Funktionen mit endlichem Bild. Informatik Berichte 152, FernUniversität Hagen, Hagen, December 1993.

55 Peter Hertling. *Unstetigkeitsgrade von Funktionen in der effektiven Analysis.* PhD thesis, Fachbereich Informatik, FernUniversität Hagen, 1996. Dissertation.

56 Peter Hertling and Victor Selivanov. Complexity issues for preorders on finite labeled forests. In Benedikt Löwe, Dag Normann, Ivan Soskov, and Alexandra Soskova, editors, *Models of computation in context*, volume 6735 of *Lecture Notes in Computer Science*, pages 112–121, Heidelberg, 2011. Springer. 7th Conference on Computability in Europe, CiE 2011, Sofia, Bulgaria, June 27–July 2, 2011.

57 Peter Hertling and Victor Selivanov. Complexity issues for preorders on finite labeled forests. In Vasco Brattka, Hannes Diener, and Dieter Spreen, editors, *Logic, Computation, Hierarchies*, Ontos Mathematical Logic, pages 165–190. Walter de Gruyter, Boston, 2014.

58 Peter Hertling and Klaus Weihrauch. Levels of degeneracy and exact lower complexity bounds for geometric algorithms. In *Proceedings of the Sixth Canadian Conference on Computational Geometry*, pages 237–242, 1994. Saskatoon, Saskatchewan, August 2–6, 1994.

59 Peter Hertling and Klaus Weihrauch. On the topological classification of degeneracies. Informatik Berichte 154, FernUniversität Hagen, Hagen, February 1994.

60 Kojiro Higuchi. *Degree Structures of Mass Problems and Choice Functions.* PhD thesis, Mathematical Institute, Tohoku University, Sendai, Japan, January 2012.

61 Kojiro Higuchi and Takayuki Kihara. Inside the Muchnik degrees I: Discontinuity, learnability and constructivism. *Annals of Pure and Applied Logic*, 165(5):1058–1114, 2014.

62 Kojiro Higuchi and Takayuki Kihara. Inside the Muchnik degrees II: The degree structures induced by the arithmetical hierarchy of countably continuous functions. *Annals of Pure and Applied Logic*, 165(6):1201–1241, 2014.

63 Kojiro Higuchi and Arno Pauly. The degree structure of Weihrauch reducibility. *Log. Methods Comput. Sci.*, 9(2):2:02, 17, 2013.

64 Denis R. Hirschfeldt. *Slicing the Truth, On the Computable and Reverse Mathematics of Combinatorial Principles*, volume 28 of *Lecture Notes Series, Institute for Mathematical Sciences, National University of Singapore*. World Scientific, Singapore, 2015.

65 Denis R. Hirschfeldt. Some questions in computable mathematics. In Adam Day, Michael Fellows, Noam Greenberg, Bakhadyr Khoussainov, Alexander Melnikov, and Frances Rosamond, editors, *Computability and Complexity: Essays Dedicated to Rodney G. Downey on the Occasion of His 60th Birthday*, volume 10010 of *Lecture Notes in Computer Science*, pages 22–55. Springer, Cham, 2017.

66 Denis R. Hirschfeldt and Carl G. Jockusch. On notions of computability-theoretic reduction between Π^1_2 principles. *Journal of Mathematical Logic*, 16(1):1650002, 59, 2016.

67 Jeffry L. Hirst and Carl Mummert. Reverse mathematics of matroids. In Adam Day, Michael Fellows, Noam Greenberg, Bakhadyr Khoussainov, Alexander Melnikov, and Frances Rosamond, editors, *Computability and Complexity: Essays Dedicated to Rodney G. Downey*

on the Occasion of His 60th Birthday, volume 10010 of *Lecture Notes in Computer Science*, pages 143–159. Springer, Cham, 2017.

68 Rupert Hölzl and Paul Shafer. Universality, optimality, and randomness deficiency. *Annals of Pure and Applied Logic*, 166(10):1049–1069, 2015.

69 Mathieu Hoyrup. Genericity of weakly computable objects. *Theory of Computing Systems*, 60(3):396–420, 2017.

70 Mathieu Hoyrup, Cristóbal Rojas, and Klaus Weihrauch. Computability of the Radon-Nikodym derivative. In *Models of computation in context*, volume 6735 of *Lecture Notes in Comput. Sci.*, pages 132–141, Heidelberg, 2011. Springer.

71 Mathieu Hoyrup, Cristóbal Rojas, and Klaus Weihrauch. Computability of the Radon-Nikodym derivative. *Computability*, 1(1):3–13, 2012.

72 Akitoshi Kawamura. Lipschitz continuous ordinary differential equations are polynomial-space complete. In *24th Annual IEEE Conference on Computational Complexity*, pages 149–160. IEEE Computer Soc., Los Alamitos, CA, 2009.

73 Akitoshi Kawamura. Lipschitz continuous ordinary differential equations are polynomial-space complete. *Computational Complexity*, 19(2):305–332, 2010.

74 Akitoshi Kawamura and Stephen Cook. Complexity theory for operators in analysis. In *Proceedings of the 42nd ACM symposium on Theory of computing*, STOC '10, pages 495–502, New York, 2010. ACM.

75 Akitoshi Kawamura and Hiroyuki Ota. Small complexity classes for computable analysis. In *Mathematical foundations of computer science 2014. Part II*, volume 8635 of *Lecture Notes in Comput. Sci.*, pages 432–444. Springer, Heidelberg, 2014.

76 Akitoshi Kawamura, Hiroyuki Ota, Carsten Rösnick, and Martin Ziegler. Computational complexity of smooth differential equations. In *Mathematical foundations of computer science 2012*, volume 7464 of *Lecture Notes in Comput. Sci.*, pages 578–589. Springer, Heidelberg, 2012.

77 Akitoshi Kawamura, Hiroyuki Ota, Carsten Rösnick, and Martin Ziegler. Computational complexity of smooth differential equations. *Logical Methods in Computer Science*, 10:1:6,15, 2014.

78 Akitoshi Kawamura and Arno Pauly. Function spaces for second-order polynomial time. In *Language, life, limits*, volume 8493 of *Lecture Notes in Comput. Sci.*, pages 245–254. Springer, Cham, 2014.

79 Takayuki Kihara. Borel-piecewise continuous reducibility for uniformization problems. *Logical Methods in Computer Science*, 12(4), October 2016.

80 Takayuki Kihara and Arno Pauly. Dividing by zero - how bad is it, really? In Piotr Faliszewski, Anca Muscholl, and Rolf Niedermeier, editors, *41st International Symposium on Mathematical Foundations of Computer Science (MFCS 2016)*, volume 58 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 58:1–58:14, Dagstuhl, Germany, 2016. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

81 Alexander P. Kreuzer. On the strength of weak compactness. *Computability*, 1(2):171–179, 2012.

82 Alexander P. Kreuzer. Bounded variation and the strength of Helly's selection theorem. *Logical Methods in Computer Science*, 10(4:16):1–23, 2014.

83 Alexander P. Kreuzer. From Bolzano-Weierstraß to Arzelà-Ascoli. *Mathematical Logic Quarterly*, 60(3):177–183, 2014.

84 Oleg V. Kudinov, Victor L. Selivanov, and Anton V. Zhukov. Undecidability in Weihrauch degrees. In Fernando Ferreira, Benedikt Löwe, Elvira Mayordomo, and Luís Mendes Gomes, editors, *Programs, Proofs, Processes*, volume 6158 of *Lecture Notes in Computer Science*, pages 256–265, Berlin, 2010. Springer. 6th Conference on Computability in Europe, CiE 2010, Ponta Delgada, Azores, Portugal, June/July 2010.

85 Rutger Kuyper. On Weihrauch reducibility and intuitionistic reverse mathematics. *Journal of Symbolic Logic*, 82(4):1438–1458, 2017.

86 Stéphane Le Roux and Arno Pauly. Closed choice for finite and for convex sets. In Paola Bonizzoni, Vasco Brattka, and Benedikt Löwe, editors, *The Nature of Computation. Logic, Algorithms, Applications*, volume 7921 of *Lecture Notes in Computer Science*, pages 294–305, Berlin, 2013. Springer. 9th Conference on Computability in Europe, CiE 2013, Milan, Italy, July 1-5, 2013.

87 Stéphane Le Roux and Arno Pauly. Finite choice, convex choice and finding roots. *Logical Methods in Computer Science*, 11(4):4:6, 31, 2015.

88 Stéphane Le Roux and Arno Pauly. Weihrauch degrees of finding equilibria in sequential games (extended abstract). In Arnold Beckmann, Victor Mitrana, and Mariya Soskova, editors, *Evolving Computability*, volume 9136 of *Lecture Notes in Computer Science*, pages 246–257, Cham, 2015. Springer. 11th Conference on Computability in Europe, CiE 2015, Bucharest, Romania, June 29–July 3, 2015.

89 Uwe Mylatz. *Vergleich unstetiger Funktionen in der Analysis.* PhD thesis, Fachbereich Informatik, FernUniversität Hagen, 1992. Diplomarbeit.

90 Uwe Mylatz. *Vergleich unstetiger Funktionen: "Principle of Omniscience" und Vollständigkeit in der C–Hierarchie.* PhD thesis, Faculty for Mathematics and Computer Science, University Hagen, Hagen, Germany, 2006. PhD thesis.

91 Eike Neumann. *Computational Problems in Metric Fixed Point Theory and their Weihrauch Degrees.* PhD thesis, Department of Mathematics, Universität Darmstadt, 2014. MSc thesis.

92 Eike Neumann. Computational problems in metric fixed point theory and their Weihrauch degrees. *Logical Methods in Computer Science*, 11:4:20,44, 2015.

93 Eike Neumann and Arno Pauly. A topological view on algebraic computation models. *Journal of Complexity*, 44(Supplement C):1–22, 2018.

94 David Nichols. Strong reductions between relatives of the stable Ramsey's theorem. arXiv 1711.06532, 2017.

95 Hugo Nobrega. *Games for functions - Baire classes, Weihrauch degrees, Transfinite Computations, and Ranks.* PhD thesis, Institute for Logic, Language and Computation, Universiteit van Amsterdam, 2018.

96 Hugo Nobrega and Arno Pauly. Game characterizations and lower cones in the Weihrauch degrees. In Jarkko Kari, Florin Manea, and Ion Petre, editors, *Unveiling Dynamics and Complexity*, volume 10307 of *Lecture Notes in Computer Science*, pages 327–337, Cham, 2017. Springer. 13th Conference on Computability in Europe, CiE 2017, Turku, Finland, June 12-16, 2017.

97 Ludovic Patey. *The reverse mathematics of Ramsey-type theorems.* PhD thesis, Université Paris Diderot, Paris, France, 2016.

98 Ludovic Patey. The weakness of being cohesive, thin or free in reverse mathematics. *Israel Journal of Mathematics*, 216:905–955, 2016.

99 Arno Pauly. *Methoden zum Vergleich der Unstetigkeit von Funktionen.* PhD thesis, FernUniversität Hagen, 2007. MSc thesis.

100 Arno Pauly. How discontinuous is computing Nash equilibria? (Extended abstract). In Andrej Bauer, Peter Hertling, and Ker-I Ko, editors, *6th International Conference on Computability and Complexity in Analysis (CCA'09)*, volume 11 of *OpenAccess Series in Informatics (OASIcs)*, Dagstuhl, Germany, 2009. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

101 Arno Pauly. How incomputable is finding Nash equilibria? *Journal of Universal Computer Science*, 16(18):2686–2710, 2010.

102 Arno Pauly. On the (semi)lattices induced by continuous reducibilities. *Mathematical Logic Quarterly*, 56(5):488–502, 2010.

103 Arno Pauly. *Computable Metamathematics and its Application to Game Theory*. PhD thesis, University of Cambridge, Computer Laboratory, Clare College, Cambridge, 2011. PhD thesis.

104 Arno Pauly. Computability on the space of countable ordinals. arXiv 1501.00386, 2015.

105 Arno Pauly. Many-one reductions and the category of multivalued functions. *Mathematical Structures in Computer Science*, 27(3):376–404, 2017.

106 Arno Pauly and Matthew de Brecht. Towards synthetic descriptive set theory: An instantiation with represented spaces. arXiv 1307.1850, 2013.

107 Arno Pauly and Matthew de Brecht. Non-deterministic computation and the Jayne-Rogers theorem. In Benedikt Löwe and Glynn Winskel, editors, *Proceedings 8th International Workshop on Developments in Computational Models, DCM 2012, Cambridge, United Kingdom, 17 June 2012.*, volume 143 of *Electronic Proceedings in Theoretical Computer Science*, pages 87–96, 2014.

108 Arno Pauly and Matthew de Brecht. Descriptive set theory in the category of represented spaces. In *30th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 438–449, 2015.

109 Arno Pauly, Willem Fouché, and George Davie. Weihrauch-completeness for layerwise computability. *Logical Methods in Computer Science*, 14(2), May 2018.

110 Arno Pauly and Willem L. Fouché. How constructive is constructing measures? *Journal of Logic & Analysis*, 9(c3):1–44, 2017.

111 Arno Pauly and Florian Steinberg. Comparing representations for function spaces in computable analysis. *Theory of Computing Systems*, 62:557–582, 2018.

112 Michelle Porter, Adam Day, and Rodney Downey. Notes on computable analysis. *Theory of Computing Systems*, 60(1):53–111, 2017.

113 Tahina Rakotoniaina. *On the Computational Strength of Ramsey's Theorem*. PhD thesis, Department of Mathematics and Applied Mathematics, University of Cape Town, Rondebosch, South Africa, 2015. PhD thesis.

114 Victor Selivanov. Total representations. *Logical Methods in Computer Science*, 9:2:5, 30, 2013.

115 Reed Solomon. Computable reductions and reverse mathematics. In Arnold Beckmann, Laurent Bienvenu, and Nataša Jonoska, editors, *Pursuit of the Universal*, volume 9709 of *Lecture Notes in Computer Science*, pages 182–191, Switzerland, 2016. Springer. 12th Conference on Computability in Europe, CiE 2016, Paris, France, June 27 - July 1, 2016.

116 Sean Sovine. *Weihrauch Reducibility and Finite-Dimensional Subspaces*. PhD thesis, Marshall University, Huntington, 2017. MSc thesis.

117 Thorsten von Stein. *Vergleich nicht konstruktiv lösbarer Probleme in der Analysis*. PhD thesis, Fachbereich Informatik, FernUniversität Hagen, 1989. Diplomarbeit.

118 Nazanin R. Tavana and Klaus Weihrauch. Turing machines on represented sets, a model of computation for analysis. *Logical Methods in Computer Science*, 7(2):2:19, 21, 2011.

119 Klaus Weihrauch. The degrees of discontinuity of some translators between representations of the real numbers. Technical Report TR-92-050, International Computer Science Institute, Berkeley, July 1992.

120 Klaus Weihrauch. The degrees of discontinuity of some translators between representations of the real numbers. Informatik Berichte 129, FernUniversität Hagen, Hagen, July 1992.

121 Klaus Weihrauch. The TTE-interpretation of three hierarchies of omniscience principles. Informatik Berichte 130, FernUniversität Hagen, Hagen, September 1992.

122 Klaus Weihrauch. *Computable Analysis*. Springer, Berlin, 2000.

123 Klaus Weihrauch. Computable planar curves intersect in a computable point. *Computability*, 2018.

Participants

- Eric P. Astor
Google – New York, US
- Vittorio Bard
University of Torino, IT
- Laurent Bienvenu
University of Montpellier & CNRS, FR
- Vasco Brattka
Universität der Bundeswehr – München, DE
- Merlin Carl
Universität Konstanz, DE
- Raphael Carroy
Universität Wien, AT
- Matthew Colbrook
Cambridge University, GB
- Matthew de Brecht
Kyoto University, JP
- Hannes Diener
University of Canterbury – Christchurch, NZ
- Francois G. Dorais
University of Vermont, US
- Damir D. Dzhafarov
University of Connecticut – Storrs, US
- Marta Fiori Carones
University of Udine, IT
- Guido Gherardi
University of Bologna, IT
- Jun Le Goh
Cornell University, US
- Peter Hertling
Universität der Bundeswehr – München, DE
- Denis R. Hirschfeldt
University of Chicago, US
- Jeffry L. Hirst
Appalachian State University – Boone, US
- Rupert Hölzl
Universität der Bundeswehr – München, DE
- Mathieu Hoyrup
LORIA & INRIA Nancy, FR
- Akitoshi Kawamura
Kyushu University, JP
- Takayuki Kihara
Nagoya University, JP
- Ulrich Kohlenbach
TU Darmstadt, DE
- Wei Li
National University of Singapore, SG
- Alberto Marcone
University of Udine, IT
- Kenshi Miyabe
Meiji University – Kawasaki, JP
- Carl Mummert
Marshall University – Huntington, US
- Takako Nemoto
JAIST – Ishikawa, JP
- Eike Neumann
Aston University – Birmingham, GB
- Keng Meng Ng
Nanyang TU – Singapore, SG
- Sabrina Ouazzani
Université Paris-Est Créteil, FR
- Ludovic Patey
University Claude Bernard – Lyon, FR
- Arno Pauly
Swansea University, GB
- Cristóbal Rojas
Universidad Andres Bello – Santiago, CL
- Matthias Schröder
Universität der Bundeswehr – München, DE
- Victor Selivanov
A. P. Ershov Institute – Novosibirsk, RU
- Paul Shafer
University of Leeds, GB
- Giovanni Soldà
University of Leeds, GB
- Florian Steinberg
INRIA Sophia Antipolis, FR
- Patrick Uftring
TU Darmstadt, DE
- Manlio Valenti
University of Udine, IT
- Klaus Weihrauch
FernUniversität in Hagen, DE
- Linda Brown Westrick
Pennsylvania State University – University Park, US
- Keita Yokoyama
JAIST – Ishikawa, JP

Report from Dagstuhl Seminar 18371

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

Edited by

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

1 **University of Naples, IT,** `pieroandrea.bonatti@unina.it`
2 **RWTH Aachen, DE,** `decker@informatik.rwth-aachen.de`
3 **Wirtschaftsuniversität Wien, AT,** `axel.polleres@wu.ac.at`
4 **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

2012 ACM Subject Classification Information systems → Semantic web description languages, Computing methodologies → Artificial intelligence, Computing methodologies → Knowledge representation and reasoning, Computing methodologies → Natural language processing, Computing methodologies → Machine learning, Human-centered computing → Collaborative and social computing

Keywords and phrases knowledge graphs, knowledge representation, linked data, ontologies, semantic web

Digital Object Identifier 10.4230/DagRep.8.9.29

Edited in cooperation with Michael Cochez

Acknowledgements The editors are grateful to Michael Cochez for his valuable contributions to the planning and organization of this workshop. He was in all respects the fifth organizer.

Knowledge Graphs: New Directions for Knowledge Representation on the Semantic Web, *Dagstuhl Reports*, Vol. 8, Issue 09, pp. 29–111
Editors: Piero Andrea Bonatti, Stefan Decker, Axel Polleres, and Valentina Presutti
Dagstuhl Reports
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Executive Summary

Piero Andrea Bonatti (University of Naples, IT)
Michael Cochez (Fraunhofer FIT, DE)
Stefan Decker (RWTH Aachen, DE)
Axel Polleres (Wirtschaftsuniversität Wien, AT)
Valentina Presutti (STLab, ISTC-CNR – Rome, IT)

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 summarisation), 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 statistical) 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-fertilisation 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 summarised 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 behaviour 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).

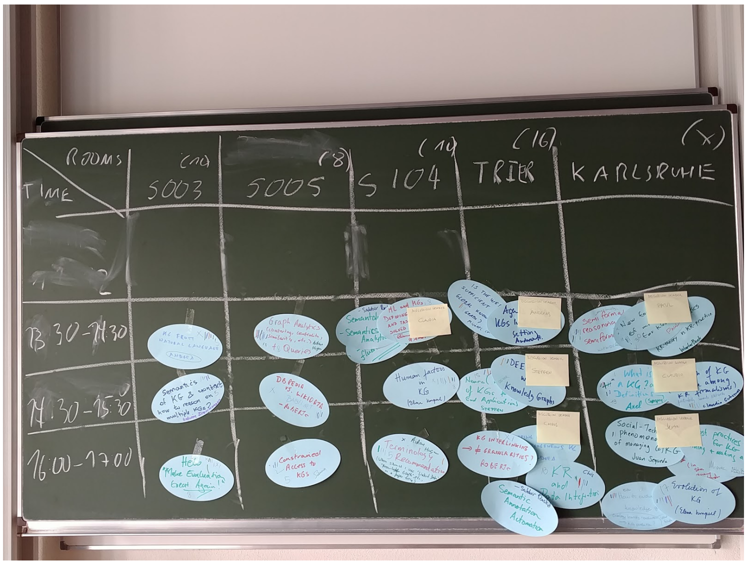

Figure 1 Blackboard with post-its from the open session.

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 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.

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 Table of Contents

Executive Summary

Overview of Short Talks

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 (∼ 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.

3.14 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.

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.

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.

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?

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.)

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.

3.15 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.

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.

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.

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].

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.

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

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 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.

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.

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.

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.

3.16 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)

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.

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.

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.

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.

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. de 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.

3.17 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.

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.

The rest of the chapter is organized as follows. Section 3.17 briefly explores related work that we may use as as starting points for further studying evolving knowledge graphs. Section 3.17 describes some use-cases for evolving knowledge graphs, as identified during this Dagstuhl seminar. Section 3.17 presents the initial challenges facing developers for evolving knowledge graphs. Finally, in Section 3.17, we summarize and conclude.

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.

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

[4] as the term is classically used in knowledge representation

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 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.

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.

[5] https://www.w3.org/TR/prov-o/

- **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.

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.

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.

3.18 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.

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].

Evaluation Setups

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

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.

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.

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)

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

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.

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.

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?).

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.

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 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.

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

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.

3.19 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)

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.

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].

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.

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.

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.

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

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.

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.

3.20 (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.

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.

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.

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.

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.

3.21 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.

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.

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

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.

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".

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.

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.

3.21.1 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.

3.21.2 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

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

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].

3.21.3 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?

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.

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.

3.22 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)

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.

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.

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].

[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

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:
  - **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].

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.

3.23 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.

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.

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.

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.

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 incorporates 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. Chal-

    lenges 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
   - 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

[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

Novel Representations and Paradigms

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

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.

3.24 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.

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.

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.

3.25 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.

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].

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.

3.26 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)

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.

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.

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.

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.

Participants

- Wouter Beek
University of Amsterdam, NL
- Christian Bizer
Universität Mannheim, DE
- Eva Blomqvist
Linköping University, SE
- Piero Andrea Bonatti
University of Naples, IT
- Daniel 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
Elsevier Labs – Amsterdam, NL
- Claudio Gutierrez
University of Chile – Santiago de Chile, CL
- Andreas Harth
Fraunhofer IIS – Nürnberg, DE
- Aidan Hogan
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 – Germany, DE
- Axel Polleres
Wirtschaftsuniversität Wien, AT
- Valentina Presutti
CNR – Rome, 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
Free University Amsterdam, NL
- Maria-Esther Vidal
TIB – Hannover, DE
- Michael Witbrock
IBM Research – Yorktown Heights, US
- Sonja Zillner
Siemens AG – München, DE
- Antoine Zimmermann
Ecole des Mines – St. Etienne, FR

Report from Dagstuhl Seminar 18381

Quantum Programming Languages

Edited by

Michele Mosca[1], Martin Roetteler[2], and Peter Selinger[3]

1 **University of Waterloo, CA,** michele.mosca@uwaterloo.ca
2 **Microsoft Corporation – Redmond, US,** martinro@microsoft.com
3 **Dalhousie University – Halifax, CA,** selinger@mathstat.dal.ca

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 18381 "Quantum Programming Languages", which brought together researchers from quantum computing and classical programming languages.

Seminar September 16–21, 2018 – http://www.dagstuhl.de/18381

2012 ACM Subject Classification Computer systems organization → Quantum computing, Hardware → Quantum computation, Theory of computation → Quantum computation theory

Keywords and phrases compilers, functional programming, quantum computing, reversible computing, verification

Digital Object Identifier 10.4230/DagRep.8.9.112

1 Executive Summary

Michele Mosca (University of Waterloo, CA)
Martin Roetteler (Microsoft Corporation – Redmond, US)
Peter Selinger (Dalhousie University – Halifax, CA)

This report documents the program and the outcomes of Dagstuhl Seminar 18381 "Quantum Programming Languages".

The aim of the seminar was to bring together researchers from quantum computing—in particular those focusing on quantum algorithms and quantum error correction—and classical programming languages. Open questions that were of interest to this group include new methods for circuit synthesis and optimization, compiler optimizations and rewriting, embedded languages versus non-embedded languages, implementations of type systems and error reporting for quantum languages, techniques for verifying the correctness of quantum programs, and new techniques for compiling efficient circuits and protocols for fault-tolerant questions and their 2D layout.

Quantum computing is getting real. Several laboratories around the world are implementing hardware platforms. For instance, systems based on superconducting qubits, such as those at IBM, Google, Intel, the University of Maryland, ionQ, and Rigetti are now scaling into the 50-150 qubit range.

While research on the theoretical side of the field addressed fundamental questions such as how to best leverage this new model of computation for algorithmic applications, a topic that has received significantly less attention is how to actually program quantum computers. To take advantage of the immense computing power offered by quantum computers as they

Quantum Programming Languages, *Dagstuhl Reports*, Vol. 8, Issue 09, pp. 112–132
Editors: Michele Mosca, Martin Roetteler, and Peter Selinger
DAGSTUHL REPORTS Dagstuhl Reports
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

come online in the coming years, software tools will be essential. We want these tools to be available, efficient and reliable, so that we can quickly and reliably reap the positive benefits that quantum computers have to offer.

It is clear that quantum programming will require tools for automatically generating large-scale circuits and for synthesizing circuits from elementary fault-tolerant gates which then can be carried out by a future quantum computer. However, it is less clear what the best way will be to go about these challenging issues. Questions that were discussed at the seminar include the following:

- How can we program a quantum computer? What are the basic structures that a language should support and how can a compiler help a user develop abstract/high-level reasoning about algorithms?
- How do we model the underlying instruction set? As currently the underlying hardware is quickly evolving, how can we best model a fault-tolerant quantum computer?
- How to compile and optimize quantum programs? Automatic translation of high-level programs into circuits will be key to program quantum computers. How to design good tools for this?
- How to we test and verify quantum programs? Given that it is hard for classical computers to simulate the time evolution of a quantum computer, how can we ascertain correctness of a circuit?

The seminar brought together some 44 researchers with diverse skill sets from quantum computing, mathematical foundations of programming languages, implementation of programming languages, and formal verification. The seminar consisted of 23 talks, as well as a number of vibrant discussion sessions and a software demonstration session. The sessions where:

- Wine Cellar discussion, moderated by Sabine Glesner. This was our first discussion session. We discussed the questions raised by Sabine Glesner during her talk: Why do we need quantum programming languages? Which "killer applications" would make quantum programming languages successful? What are appropriate abstractions from quantum hardware? What are theoretical models for quantum computing?
- Discussion session on Debugging, moderated by Rodney Van Meter. This session focused on what are appropriate debugging techniques for quantum computing. The issue arises because the most common classical debugging technique, setting break points and examining the program state, cannot be applied in the context of quantum computing.
- Discussion session on Challenge Problems for Quantum Computing, moderated by Earl Campbell. In this session, we discussed coming up with well-defined problems with some success quantifier for quantum computation, similar to the successful SAT competitions.
- Group survey session on a Bird's Eye View on Quantum Languages, moderated by Robert Rand. In this session, the group compiled a list of all quantum programming languages and toolkits we are currently aware of, and classified them according to various criteria, for example, whether the languages are imperative or functional, whether the computational paradigm is circuit generation or Knill's QRAM model, whether the language is high-level or assembly, whether it supports type-safety and/or verification, etc.
- Group survey session on Tools for Quantum Optimization, moderated by Matthew Amy. In this session, the group compiled a list of available tools for optimization of quantum circuits.
- Group discussion on Opportunities for Education and Outreach, moderated by Rodney Van Meter. The discussion centered on new opportunities for public outreach and education that are enabled by the emergence of new quantum tools.

- Software demonstration session, moderated by Martin Roetteler. In this session, 10 researchers gave rapid demonstrations, of a about 10 minutes each, of various software tools they have designed.

Most of the participants rated the seminar as a success. We managed to connect researchers from different communities, and engaged in a vibrant exchange of novel ideas, and started to tackle important problems such as the analysis of quantum algorithms for real-world computational problems, compiler optimizations, reversible computing, and fault-tolerant quantum computing.

2 Table of Contents

Executive Summary

Overview of Talks

Working groups

3 Overview of Talks

3.1 Functional Verification of Quantum Circuits

Matthew Amy (University of Waterloo, CA)

Main reference Matthew Amy: "Towards Large-scale Functional Verification of Universal Quantum Circuits", CoRR, Vol. abs/1805.06908, 2018.
URL https://arxiv.org/abs/1805.06908

We introduce a framework for the formal specification and verification of quantum circuits based on the Feynman path integral. Our formalism, built around exponential sums of polynomial functions, provides a structured and natural way of specifying quantum operations, particularly for quantum implementations of classical functions. Verification of circuits over all levels of the Clifford hierarchy with respect to either a specification or reference circuit is enabled by a novel rewrite system for exponential sums with free variables.

We evaluate our methods by performing automated verification of optimized Clifford+T circuits with up to 100 qubits and thousands of T gates, as well as the functional verification of quantum algorithms using hundreds of qubits. We further show that our method can perform the simulation of a Hidden Shift algorithm due to Roetteler with 100 qubits in just minutes on a tablet computer.

3.2 Phase polynomials, T-count optimisation and Lempel's algorithm

Earl Campbell (University of Sheffield, GB)

Main reference Luke E Heyfron, Earl T Campbell: "An efficient quantum compiler that reduces T count", in Quantum Science and Technology, Vol. 4 (1), p. 015004, 2018.
URL https://doi.org/10.1088/2058-9565/aad604

I review the basics of the phase polynomials and the connection to T-count optimisation, summarising the work of Amy and Mosca as well as my own work on this with Luke Heyfron. This leads to a 3-tensor optimisation problem that is hard. But is it closely related to an easier (relaxed) optimisation problem solved in the 1970s by Lempel (https://epubs.siam.org/doi/abs/10.1137/0204014) All of my work on the problem has been based on modifying Lempel's algorithm to build a heuristic for the harder quantum problem. Rather than getting into the details of the hard quantum problem, I sketch Lempel's method as I believe it is not well known and could have further applications in the field.

3.3 Low overhead quantum computation using lattice surgery

Austin G. Fowler (Google Research – Mountain View, US)

The surface code is a method of detecting errors in a quantum computer. Many different methods of computing using this code exist. We fully analyze lattice surgery and show that this method is unambiguously better than braiding defects, the previous standard method.

3.4 Dependent types in Proto-Quipper

Frank Fu (Dalhousie University – Halifax, CA)

Joint work of Peter Selinger
Main reference Francisco Rios, Peter Selinger: "A categorical model for a quantum circuit description language", in Proc. of the 14th International Conference on Quantum Physics and Logic, QPL 2017, Nijmegen, The Netherlands, 3-7 July 2017., EPTCS, Vol. 266, pp. 164–178, 2017.
URL https://doi.org/10.4204/EPTCS.266.11

Are dependent types useful for quantum circuit programming? In this talk, I will present an implementation of dependently typed Proto-Quipper, a stand-alone language for quantum circuit description. I will argue that dependent types are useful in three aspects:

1. Precise types for quantum circuit description functions.
2. Precise notion of boxing a family of circuits.
3. Encapsulating unused wires via existential dependent data types.

3.5 Reflections on what programming languages are good for – traditionally and in the face of quantum computing

Sabine Glesner (TU Berlin, DE)

When I first heard about quantum computing, which was in 1994, it was at about the same time I first heard about internet browsers. At that time, I could not tell which of these two developments would be faster. Today we know that it was not quantum computing. Nevertheless, quantum computing has made enormous progress during the last years, so much that we even have a Dagstuhl seminar on quantum programming languages. In this talk, I want to sum up what traditional programming languages have been good for and raise the question of what the situation looks like for quantum computing.

3.6 Reversible Programming Languages – From Classical Results to Recent Developments

Robert Glück (University of Copenhagen, DK)

Joint work of Martin Holm Cservenka, Robert Glück, Tue Haulund, Torben Ægidius Mogensen
Main reference Martin Holm Cservenka, Robert Glück, Tue Haulund, Torben Ægidius Mogensen: "Data Structures and Dynamic Memory Management in Reversible Languages", in Proc. of the 10th International Conference on Reversible Computation, RC 2018, Leicester, UK, September 12–14, 2018, Lecture Notes in Computer Science, Vol. 11106, pp. 269–285, Springer, 2018.
URL https://doi.org/10.1007/978-3-319-99498-7_19

This talk highlighted the principles and main ideas of reversible programming languages with which we have been working for the past several years (the "Copenhagen Interpretation of Reversible Computing"). Reversible languages form their own distinct class of programming languages because they are deterministic in both computation directions. They complement the mainstream programming languages like C and Haskell that are backward nondeterministic. Recent developments with dynamic memory management allowed the design and implementation of reversible object-oriented and functional languages. This enables, for the first time, the reversible manipulation of high-level dynamic data abstractions such as binary trees, lists and queues.

3.7 Quantum Linguistic Relativity

Christopher Granade (Microsoft Corporation – Redmond, US)

In this talk, I will consider a set of goals for new quantum programming languages, motivated by applications and with an eye to making it easier for new quantum programmers to get started. To meet these goals, I propose thinking of quantum language design in terms of linguistic relativity, the hypothesis that the language in which we express an idea affects how we think about that idea. Finally, I present Q# as a case study for this approach to design, and discuss how we chose Q# features according to linguistic relativity.

3.8 The OpenQL programming framework

Nader Khammassi (TU Delft, NL)

Quantum computing is rapidly evolving, especially after the discovery of several efficient quantum algorithms solving intractable classical problems. Expressing these quantum algorithms using a high-level programming language and making them executable on a quantum processor, while abstracting hardware details and targetting different qubit technologies, is an important problem. After discussing the different compilation challenges, we present the OpenQL programming framework, and show how its modular design allows the integration of a full-stack quantum computer architecture for different qubit technologies.

3.9 Cheaper alternative to Euler decomposition for SU(2) gates and fall-back circuits

Vadym Kliuchnikov (Microsoft Corporation – Redmond, US)

We give an alternative to Euler decomposition that leads to average case circuit complexity scaling as $7\log_5(1/\varepsilon)$ as opposed to $9\log_5(1/\varepsilon)$ for Pauli+V gate sets. The idea readily generalizes to many other gate sets.

3.10 Operator algebras and their role in quantum programming languages

Albertus Johannis Lindenhovius (Tulane University – New Orleans, US)

Quantum systems are usually described in the Hilbert space formalism. Operator algebras, which were introduced by von Neumann, form an alternative formalism with several advantages over the Hilbert space formalism, such as the possibility of describing the interaction between quantum and classical phenomena in one framework.

We discuss how operator algebras, which are algebras of operators on a Hilbert space, can be used in the semantics of quantum programming languages. Furthermore, we discuss which categorical properties of a certain class of operator algebras correspond to what features of the quantum programming language for which it is used in the semantics. Can we single out one class of operator algebras that has all categorical properties sufficient for a higher-order quantum programming language with recursion?

3.11 NISQ optimization for CNOT and CNOT+T circuits

Beatrice Nash (MIT – Cambridge, US)

Near-term quantum devices have limited physical qubit connectivity, and performing operations between non-adjacent qubits can be very expensive. In this talk, I will discuss ways to extend current circuit optimization methods to take into account these restrictions.

3.12 Verified Quantum Programming in QWIRE: Optimization and Error Correction

Robert Rand (University of Maryland – College Park, US)

Joint work of Robert Rand, Jennifer Paykin, Dong-Ho Lee, Steve Zdancewic, Kesha Hietala, Michael Hicks, Xiaodi Wu

Main reference Jennifer Paykin, Robert Rand, Steve Zdancewic: "QWIRE: a core language for quantum circuits", in Proc. of the 44th ACM SIGPLAN Symposium on Principles of Programming Languages, POPL 2017, Paris, France, January 18-20, 2017, pp. 846–858, ACM, 2017.

URL https://doi.org/10.1145/3009837.3009894

We present QWIRE, a quantum circuit language and formal verification tool. QWIRE possesses a denotational semantics in terms of density matrices and is embedded in the Coq proof assistant, allowing us to check our quantum circuits against their mathematical specifications. In this talk, we look at a variety of proofs of circuit specifications in Coq. We then examine two pressing issues for quantum programming: Verified optimization and error-aware semantics and the challenges of incorporating them in QWIRE.

3.13 Proto-Quipper-M: A Categorically Sound Quantum Circuit Description Language.

Francisco Rios (Dalhousie University – Halifax, CA)

Joint work of Francisco Rios, Peter Selinger
Main reference Francisco Rios, Peter Selinger: "A categorical model for a quantum circuit description language", in Proc. of the 14th International Conference on Quantum Physics and Logic, QPL 2017, Nijmegen, The Netherlands, 3-7 July 2017., EPTCS, Vol. 266, pp. 164–178, 2017.
URL https://doi.org/10.4204/EPTCS.266.11

Quipper is a practical programming language for describing families of quantum circuits. In this talk, we formalize a small, but useful fragment of Quipper called Proto-Quipper-M. Unlike its parent Quipper, this language is type-safe and has a formal denotational and operational semantics. Proto-Quipper-M is also more general than Quipper, in that it can describe families of morphisms in any symmetric monoidal category, of which quantum circuits are but one example. We design Proto-Quipper-M from the ground up, by first giving a general categorical model of parameters and states. After finding some interesting categorical structures in the model, we then define the programming language to fit the model. We cement the connection between the language and the model by proving type safety, soundness, and adequacy properties.

3.14 Toward the first quantum simulation with quantum speedup

Neil Julien Ross (Dalhousie University – Halifax, CA)

Joint work of Andrew M. Childs, Dmitri Maslov, Yunseong Nam, Neil J. Ross, Yuan Su

As we approach the development of a quantum computer with tens of well-controlled qubits, it is natural to ask what can be done with such a device. Specifically, we would like to construct an example of a practical problem that is beyond the reach of classical computers, but that requires the fewest possible resources to solve on a quantum computer. We address this problem by considering quantum simulation of spin systems, a task that could be applied to understand phenomena in condensed matter physics such as many-body localization. We synthesize explicit quantum circuits for three leading quantum simulation algorithms, one based on product formulas (PF), one based on implementing the Taylor series as a linear combination of unitaries (TS), and another using the recent quantum signal processing approach (QSP). We employ a wide range of techniques to develop tighter error bounds and optimize gate-level implementations. Surprisingly, even for simulations of small systems, we find that the fourth-order PF algorithm outperforms lower-order PF algorithms and that the TS and QSP algorithms require even fewer gates (although at the cost of requiring more qubits). However, the cost of PF algorithms can be reduced significantly by using empirical error bounds, so that PF algorithms remain competitive in contexts where a rigorous guarantee on the accuracy of the simulation is not essential. Our circuits are smaller by several orders of magnitude than those for the simplest classically-hard instances of problems such as factoring and quantum chemistry, and we hope they will pave the way toward the first practical application of a quantum computer.

3.15 Automatic Synthesis in Quantum Programming Languages

Mathias Soeken (EPFL – Lausanne, CH)

Joint work of Mathias Soeken, Bruno Schmitt, Giulia Meuli, Fereshte Mozafari, Giovanni De Micheli, Martin Roetteler, Thomas Häner
Main reference Mathias Soeken, Thomas Häner, Martin Roetteler: "Programming quantum computers using design automation", in Proc. of the 2018 Design, Automation & Test in Europe Conference & Exhibition, DATE 2018, Dresden, Germany, March 19-23, 2018, pp. 137–146, IEEE, 2018.
URL https://doi.org/10.23919/DATE.2018.8341993

When translating quantum programs into (technology-dependent) quantum circuits, we are often confronted with translating conventional classical logic in forms of permutations, Boolean functions, or logic networks. RevKit is a toolkit that combines several state-of-the-art approaches for synthesis, optimization, and mapping. RevKit is powered by the EPFL logic synthesis libraries, such as tweedledum, alice, mockturtle, and kitty.

In the presentation, we showcase several applications of RevKit in modern quantum programming flows. The examples include integrations with Qiskit, ProjectQ, pyQuil, Q#, and Cirq.

3.16 Representing quantum control

Benoit Valiron (Centrale Supelec – Orsay, FR)

Joint work of Amr Sabry, Benoît Valiron, Juliana Kaizer Vizzotto
Main reference Amr Sabry, Benoît Valiron, Juliana Kaizer Vizzotto: "From Symmetric Pattern-Matching to Quantum Control", in Proc. of the 21st International Conference on the Foundations of Software Science and Computation Structures, FOSSACS 2018, Thessaloniki, Greece, April 14–20, 2018, Lecture Notes in Computer Science, Vol. 10803, pp. 348–364, Springer, 2018.
URL https://doi.org/10.1007/978-3-319-89366-2_19

One perspective on quantum algorithms is that they are classical algorithms having access to a special kind of memory with exotic properties. This perspective suggests that, even in the case of quantum algorithms, the control flow notions of sequencing, conditionals, loops, and recursion are entirely classical. There is however, another notion of control flow, that is itself quantum. The notion of quantum conditional expression is reasonably well-understood: the execution of the two expressions becomes itself a superposition of executions. The quantum counterpart of loops and recursion is however not believed to be meaningful in its most general form. In this talk (based on [1]), we discuss how, under the right circumstances, a reasonable notion of quantum loops and recursion is possible. To this aim, we first propose a classical, typed, reversible language with lists and fixpoints based on Theseus [2]. We then extend this language to the closed quantum domain (without measurements) by allowing linear combinations of terms and restricting fixpoints to structurally recursive fixpoints whose termination proofs match the proofs of convergence of sequences in infinite-dimensional Hilbert spaces. We additionally give an operational semantics for the quantum language in the spirit of algebraic lambda-calculi. This permits to reconciliate several approaches, such as the quantum tests of QML [3], Ying's quantum loops [4] and linear algebraic approaches [5]

References

1 Sabry, A., Valiron, B., and Vizzotto, J. K. (2018). From Symmetric Pattern-Matching to Quantum Control. In International Conference on Foundations of Software Science and Computation Structures (pp. 348–364). Springer.

2 James, R. P., and Sabry, A. (2014). Theseus: a high-level language for reversible computing. Unpublished manuscript.

3 Altenkirch, T., and Grattage, J. (2005). A Functional Quantum Programming Language. In P. Panangaden, Proceedings of the 20th Symposium on Logic in Computer Science (LICS'05) (pp. 249–258). Chicago, Illinois, US. IEEE Computer Society Press.

4 Ying, M. (2016). Foundations of quantum programming. Morgan Kaufmann.

5 Arrighi, P., and Dowek, G. (2008). Linear-algebraic lambda-calculus: higher-order, encodings, and confluence. In A. Voronkov, Proceedings of the 19th International Conference on Rewriting Techniques and Applications (RTA'08) (pp. 17–31). Hagenberg, Austria: Springer.

3.17 Error-aware compilation for the IBM 20-qubit machine

Rodney Van Meter (Keio University – Fujisawa, JP)

Joint work of Yulu Pan, Shin Nishio, Takahiko Satoh, Rodney Van Meter

Roughly, our first project [1] is to characterize qubits, operation fidelity, and path fidelity for moving qubits (or doing long-distance gates), and the second [2] is to take that information and place qubits on processor and plan their movement.

While various projects have worked on compiling programs to meet the constraints of small quantum processors, we believe this is the first work to focus on the inhomogeneity in actual gate errors due to minor differences between qubits as the key metric for placing variable qubits on the physical qubits on the processor.

The error rates for qubits and gates are measured using random benchmarking. For the placement phase, we are using beam search combined with a modified form of Dijkstra's shortest path first, and using the product of gate fidelities as our prediction for success probability on complex circuits. We use the Cuccaro adder circuit as our application for testing the compilation, and KL-divergence as a measure of the quality of circuits. We compared the existing QISkit compiler with QOPTER.

Our work, while not yet complete, suggests that the single number of gate fidelity is not an adequate measure of success rate. We matched the QISkit success probability while using minimal computational resources in the compilation phase for a 5-qubit adder circuit, but the actual success probability is still low, and the estimated success probability actually far lower. The relative ranking of choice of circuit shows an intermediate level of correlation with the ordering of success probability in circuits. Thus, this complex problem is still ripe for new approaches and hard work, and our future work includes continuing development of these tools.

References

1 Shin Nishio, Yulu Pan, Takahiko Satoh, Rodney Van Meter. "High Fidelity Qubit Mapping for IBM Q," *Proc. 2nd International Workshop on Quantum Compilation*, 2018

2 Yulu Pan, Shin Nishio, Takahiko Satoh, Rodney Van Meter, Hideharu Amano. "QOPTER –Quantum program OPTimizER– ," *Proc. 2nd International Workshop on Quantum Compilation*, 2018

3.18 Data-structures and Methods for the Design of Quantum Computations

Robert Wille (Johannes Kepler Universität Linz, AT)

Joint work of Alwin Zulehner, Robert Wille
Main reference Alwin Zulehner, Robert Wille: "Advanced Simulation of Quantum Computations", TCAD, 2018.
URL https://doi.org/10.1109/TCAD.2018.2834427

In the past decades, the Computer-Aided Design (CAD) community was frequently faced with tremendously complex challenges that often required the efficient consideration of problems of exponential (or even greater) size. In order to tackle these, researchers and engineers developed sophisticated CAD methods employing, e.g., decision diagrams or sophisticated reasoning engines. In contrast, many design problems in the quantum domain are still addressed in a rather straight-forward fashion, e.g., by exponential array-based descriptions or enumerative search algorithms. This talk illustrates how established concepts from the conventional design of circuits and systems can be applied to improve the design of quantum computations. By this, the talk will "bridge" the CAD and the quantum communities by showing how the combination of expertise from both domains eventually yields efficient design methods for quantum computation. The application of those data-structures and methods is exemplarily demonstrated by means of simulation of quantum computation.

3.19 Logic level circuit optimization for topological quantum computation

Shigeru Yamashita (Ritsumeikan University – Shiga, JP)

The TQC (Topological Quantum Computing) model has been receiving a lot of attention because it has proven to be one of the most promising fault-tolerant quantum computation models. In the TQC conceptual model, we arrange physical measurement sequences corresponding to computational steps of quantum computation in a three-dimensional space. While some transformation rules for this arranged three-dimensional space have been known, there was no known systematic way to use the rules to optimize the arranged space. This talk proposes an efficient systematic way to use the known transformation rules by considering the arranged space as a set of loops.

3.20 Reasoning about Parallel Quantum Programs

Mingsheng Ying (University of Technology – Sydney, AU)

We initiate the study of parallel quantum programming by defining the operational and denotational semantics of parallel quantum programs. The technical contributions include: (1) finding a series of useful proof rules for reasoning about correctness of parallel quantum programs; and (2) proving a strong soundness theorem of these proof rules, showing that partial correctness is well maintained at each step of transitions in the operational semantics of a parallel quantum program. This is achieved by partially overcoming the following conceptual challenges that are never present in classical parallel programming: (i) the intertwining of nondeterminism caused by quantum measurements and introduced by parallelism; (ii) entanglement between component quantum programs; and (iii) combining quantum predicates in the overlap of state Hilbert spaces of component quantum programs with shared variables. It seems that a full solution to these challenges and developing a (relatively) complete proof system for parallel quantum programs are still far beyond the current reach.

3.21 Recursive types for linear/non-linear quantum programming

Vladimir Zamdzhiev (LORIA – Nancy, FR)

Linear/non-linear lambda calculi provide a natural framework for quantum programming. By making a distinction between intuitionistic (non-linear / classical) and linear types, we may model classical data and quantum data. The latter cannot be copied or deleted, which is conveniently ensured by the linearity of the type system, whereas the former may be freely copied and discarded, which is also conveniently allowed by the non-linear part of the type system.

In this talk, we consider the problem of extending such a lambda calculus with recursive types. We design the type system such that we may distinguish between intuitionistic recursive types and linear recursive types. We also describe some work in progress on a conjectured denotational model that soundly models our lambda calculus.

3.22 Quantum Calculi: from theory to language design

Margherita Zorzi (University of Verona, IT)

Joint work of Margherita Zorzi, Andrea Masini, Ugo Dal Lago, Luca Paolini, Luca Roversi
Main reference Margherita Zorzi: "On quantum lambda calculi: a foundational perspective", Mathematical Structures in Computer Science, Vol. 26(7), pp. 1107–1195, 2016.
URL https://doi.org/10.1017/S0960129514000425

In the last 20 years several approaches to quantum programming have been introduced. In this report we will focus on functional calculi and in particular on the QRAM architectural model. We explore the twofold perspective (theoretical and concrete) of the approach and we will list the main problems one has to face in quantum language design.

3.23 Compiling Quantum Circuits to NISQ Devices

Alwin Zulehner (Johannes Kepler Universität Linz, AT)

Joint work of Alwin Zulehner, Alexandru Paler, Robert Wille
Main reference Alwin Zulehner, Alexandru Paler, Robert Wille: "An efficient methodology for mapping quantum circuits to the IBM QX architectures", to appear in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems.
URL https://doi.org/10.1109/TCAD.2018.2846658

The Noisy Intermediate-Scale Quantum (NISQ) technology is currently investigated by major players in the field to build the first practically useful quantum computer. IBM QX architectures are the first ones which are already publicly available today. However, in order to use them, the respective quantum circuits have to be compiled for the respectively used target architecture. This demands solutions for automatically and efficiently conducting this compilation process. In this work, we offer solutions to this problem that satisfy all constraints given by the architecture and, at the same time, aim to keep the overhead in terms of additionally required quantum gates minimal. Our experimental evaluation shows that the proposed approach significantly outperforms IBM's own solution regarding fidelity of the compiled circuit as well as runtime. Moreover, to emphasize development, IBM launched a challenge with the goal to optimize such compilers for a certain set of random quantum circuits. Since these circuits represent a worst case scenario for our approach, we developed a correspondingly adjusted version. It has been declared the winner of this so-called QISKit developer challenge, since it yields compiled circuits with at least 10% better costs than the other submissions, while generating them at least 6 times faster (according to IBM). Implementations of the proposed methodologies are publicly available at http://iic.jku.at/eda/research/ibm_qx_mapping.

4 Working groups

4.1 Tools for Quantum Optimization

Matthew Amy (University of Waterloo, CA)

In this session, the group compiled a list of available tools for optimization of quantum circuits. We came up with the following list of tools. While the list is probably highly incomplete, we hope that it is a useful starting point.

- **Feynman,** by Matthew Amy. Optimizations: z-rotation optimization, CNOT-count optimization, T-depth optimization. Language: Haskell library, command-line interface. License: BSD-2. Availability: Github.
- **Newsynth/Gridsynth,** by Neil J. Ross and Peter Selinger. Optimizations: single-qubit Z-rotations to Clifford+T. Language: Haskell library, command-line interface. License: GPL-3. Availability: Hackage.
- **TOpt,** by Earl Campbell. Optimizations: Clifford+T to Clifford+T, T-gate minimization. Language: C++, command-line interface. License: GPL-3. Availability: Github.
- **IonQ's tool,** by IonQ. Optimizations: Clifford+z-rotations+Toffoli to Clifford+z-rotations. Language: Fortran. License: Proprietary.

- **RevKit,** by Mathias Soeken. Optimizations: Look-up table hierarchical reversible synthesis (LHRS). CNOT minimization. Produces Clifford+T. Language: C++, command-line interface. Python bindings. License: MIT.
- **pQCS,** by Olivia Di Matteo and Michele Mosca. Optimizations: Multi-qubit unitary to Clifford+T. Availability: from https://qsoft.iqc.uwaterloo.ca/. License: For research only.
- **IBM QX mapping SU(4),** by Alwin Zulehner and Robert Wille. Optimizations: $SU(4)$ to CNOT+SWAP+1-qubit gates. Availability: Github. License: Non-commercial use only.

4.2 Challenge problems in quantum computation

Earl Campbell (University of Sheffield, GB)

We discussed possible "challenge problems" in the optimisation of quantum circuits. The idea was to come up with very well defined problems with some success quantifer, similar to the successful SAT competitions. The most popular idea was to consider Hamiltonian simulation of small systems of 10–20 qubits and minimise the number of gates required to achieved a precision of 10^{-3}. Suggested Hamiltonians included:

1. The 1D Heisenberg chain;
2. Jellium;
3. Quantum chemistry hamiltonians available in the openFermion packages.

Given a Hamiltonian, one could consider implementing the operator e^{iHt}, but an additional interesting problem is phase estimation, where one implemented a controlled e^{iHt}. Here, t and the number of circuit repetitions ought to be optimised to minimise the Fisher information of the parameter estimated.

4.3 Wine Cellar Discussion on Quantum Programming Languages

Sabine Glesner (TU Berlin, DE)

Joint work of all the participants of the Dagstuhl seminar 18381

This is an abbreviated summary of a discussion in the Dagstuhl wine cellar that took place during the Dagstuhl seminar 18381 on Quantum Programming Languages on Monday, September 20, 2018. The starting point for the discussion was the list of questions raised in Sabine Glesner's talk.

4.3.1 What are programming languages good for, classically?

Programming languages are central in computer science. Already a brief look at the Turing award winners and their research areas reveals that programming languages never go out of date. In the past, programming languages have been helpful to abstract from hardware details, which gives more programming comfort to software developers. Type systems have been developed to enhance program correctness by allowing programmers to detect misfittings statically before the program is executed. Also, data management is important and programming languages offer a rich variety of structured data mechanisms, e.g., via class hierarchies. There are many programming language paradigms around (e.g., functional, imperative, object-oriented, logical, etc., as well as combinations thereof). It turns out that each of them supports a certain class of problems. A definite achievement in the area of programming languages is program analyses. They analyse programs statically, and are usually conservative. For example, live variables analysis will find variables that are definitely no longer used, but might err on the side of marking a variable as live when it *might* be live.

4.3.2 What are killer applications for quantum computing?

History has shown that new technologies are usually only successful if they are necessary to accomplish something new (a "killer application"). We brainstormed on what the likely killer applications for quantum computing would be. It was suggested that the field of *quantum chemistry* has many well-formulated and ready-made problems for which quantum computing will be very helpful, and that this will be one of the first "real" applications of quantum computing. Solving problems in quantum chemistry has potential real-world applications, such as the discovery of new catalysts to make chemical reactions more efficient (e.g., carbon capture, the production of fertilizers, or the conversion of solar energy into fuel). Also, physical qubits can be used to build up quantum sensors, which may be another early application. The participants generally agreed that cryptanalysis is not likely to be a killer application for quantum computing, both because it requires a relatively large number (compared to quantum chemistry) of fault-tolerant qubits, and also because the world will switch to post-quantum cryptography as soon as current protocols become insecure.

4.3.3 Quantum hardware

We discussed the size of current actual quantum hardware, the difference between physical and logical qubits, and their respective error rates, which are significantly higher for qubits (10^{-3}) compared to classical transistors (10^{-14}).

4.3.4 Quantum programming languages

Quantum programming languages are not necessarily complete stand-alone languages but often libraries or packages built on top of classical languages. OpenFermion is an example for such a package which is itself implemented in Python. The focus in the development of programming languages is often on the translation, which is often targeted to quantum circuits. A major concern when running quantum programs is the appearance and accumulation of errors. The longer quantum hardware runs, the higher the error rate is. Hence, it is important to understand how error correction can be done. It would be very helpful if such analyses could be done automatically. While there are lots of analyses around, still a major amount of uncertainty comes from the inputs. It is also an interesting question what level of error is tolerable.

It would be good to have benchmarks so that optimizations could be developed (see the analogy for SAT/SMT solvers for which also benchmarks exist). We pursued this question further in a separate working group "Challenge problems in quantum computation".

Debugging is another important and very difficult issue in quantum programming languages, as there are no checkpoints available. We pursued this question further in a separate working group on "Debugging".

4.3.5 Theoretical Models for Quantum Computing

This point did not receive much discussion, as it was generally agreed that Quantum Turing Machines are considered to be too cumbersome to work with, while quantum circuits are typically the formalism of choice, at least during our discussion.

4.4 Survey of Quantum Languages

Robert Rand (University of Maryland – College Park, US)

The purpose of this session was to compile a list of quantum programming languages and toolkits of interest, and to classify them according to various criteria. We considered the following languages:

Proto-Quipper	F	C	H	Ac	T/L	As	St	V (partial)	O/D/K	—
QWIRE	F	CF	H	Ac	T/L	As	E	V	D/K	—
Quipper	F	CF	H	G	T (partial)	As	E	—	—	In
ProjectQ	I	N	H	G	—	?	E	—	—	In/De (partial)
Q#	I/F	N	H	G	T (partial)	As	St	—	—	—
PyQuil	I	N	H	G	—	—	E	—	—	De
Cirq	I	C	H	S	—	?	E	—	—	—
QISKit	I	CL	H	G	—	—	E	—	—	In/De
Scaffold	I	?	H	G	—	—	St	—	—	In
(Open)QASM	I	CL	A	G	—	—	Ta	—	—	—
Quil	I	N	A	G	—	—	Ta	—	O	—

The letters after each language mean the following:

- Language paradigm: Functional (F), imperative (I).
- Target: Circuit based (C), circuits with feedback from measurements (CF), circuits with limited feedback (CL), or not circuit based (N).
- Abstraction: High-level language (H) or assembly-type language (A).
- Intended audience: General public (G), academic research (Ac), or special-purpose (S).
- Safety: type-safety (T), linearity (L).
- Run-time checks: assertions (As).
- Implementation: embedded language (E), standalone language (St), or target language (Ta).
- Support for verification (V).
- Semantics: operational (O), denotational (D), and/or categorical (K).
- Support for optimization: machine dependent (De) or machine independent (In).

4.5 Software demonstration session

Martin Roetteler (Microsoft Corporation – Redmond, US)

In this session, researchers gave rapid demonstrations of various software tools they have designed. The following is a list of the presentations, which lasted about 10 minutes each.

- Vadym Kliuchnikov: Asserts, unit tests, and Q# tracer tool
- Damian Steiger: ProjectQ: Shor's algorithm, quantum chemistry, rendering
- Frank Fu: Dependent types in Proto-Quipper
- Nader Khammassi: QASM generator
- Chris Granade: Quantum Katas in Q#
- Matt Amy: Feynman tool to optimize and verify quantum circuits
- Andrew Cross: IBM Quantum Experience demo
- Bruno Schmitt: Tweedledee and tweedledum: IRs for RevKit
- Robert Rand: QWIRE proofs in Coq
- Alwin Zulehner: Simulation with QMDDs

4.6 Debugging of Quantum Programs

Rodney Van Meter (Keio University – Fujisawa, JP)

This session focused on identifying appropriate debugging techniques for quantum computing. The issue arises because the most common classical debugging technique, setting break points and examining the program state, cannot be used in the context of quantum computing.

It was suggested that we must draw a distinction between debugging and program verification. In fact, debugging may potentially be used to answer three different questions:

- Specification: did we define the algorithm correctly?
- Code: have we correctly translated the specification to working, bug-free code?
- Runtime: does the simulator or real quantum computer execute the program as expected?

According to the experience of some former members of the IARPA QCS program, typical quantum circuits contain large classical subcircuits (usually oracles, sometimes more than 100× the size of the quantum portion). It may be beneficial to debug these parts separately, as they can be simulated efficiently.

Here is an incomplete list of classical debugging techniques. Some of these may be applicable to quantum programming, although others will not be.

- assertions (invariants in the program)
- interactive debugging (breakpoints)
- time travel debuggers (those that can step backwards in time from a crash)
- inserting print statements (all too common)
- unit testing (e.g., establishing whether executed gate count matches expected gate count)
- static analysis (including type checking)
- dynamic analysis (memory leaks?)

- code review
- testing with random instances
- post-mortem (what is a “quantum crash dump”?)

Additional discussion focused on the distinction between regression testing and performance testing, and on the difficulty of discriminating between hardware bugs and software bugs. The group identified the following research questions and directions:

- How applicable are probabilistic techniques? Is probabilistic model checking applicable?
- How do we debug on logical qubits, as opposed to physical qubits?
- Checking if a circuit is the identity is QMA-hard; is checking it up to ε still QMA-hard?
- What classes of properties do we want to check?
- How can we make verification tools useful to programmers?
- What concrete tools can we develop?

4.7 Opportunities for Education and Outreach

Rodney Van Meter (Keio University – Fujisawa, JP)

The discussion centered on new opportunities for public outreach and education that are enabled by the emergence of new quantum tools. In general, the audience for quantum tools can be divided into three categories:

- The general public, e.g., popular science enthusiasts (learners we hope to attract), who just want to learn the key ideas.
- Black box library users, who don’t care how it works.
- Algorithmists, who will need to learn how to create new interference patterns.

The first group is the most difficult to reach. It was suggested that people in this group generally have three questions, in this order:

- What does it do?
- When will I have it?
- How does it work?

Physicists tend to answer the questions in exactly the opposite order.

Participants

- Matthew Amy
University of Waterloo, CA
- Sébastien Bardin
CEA LIST, FR
- Xiaoning Bian
Dalhousie University – Halifax, CA
- Earl Campbell
University of Sheffield, GB
- Andrew Cross
IBM TJ Watson Research Center – Yorktown Heights, US
- Olivia Di Matteo
University of Waterloo, CA
- Austin G. Fowler
Google Research – Mountain View, US
- Frank Fu
Dalhousie University – Halifax, CA
- Vlad Gheorghiu
University of Waterloo, CA
- Sabine Glesner
TU Berlin, DE
- Robert Glück
University of Copenhagen, DK
- Christopher Granade
Microsoft Corporation – Redmond, US
- Markus Grassl
Max Planck Institute for the Science of Light, DE
- Thomas Häner
ETH Zürich, CH
- Shih-Han Hung
University of Maryland – College Park, US
- Nader Khammassi
TU Delft, NL
- Vadym Kliuchnikov
Microsoft Corporation – Redmond, US
- Sriram Krishnamoorthy
Pacific Northwest National Lab. – Richland, US
- Andrew John Landahl
Sandia National Labs – Albuquerque, US
- Albertus Johannis Lindenhovius
Tulane University – New Orleans, US
- Michael W. Mislove
Tulane University – New Orleans, US
- Michele Mosca
University of Waterloo, CA
- Beatrice Nash
MIT – Cambridge, US
- Jennifer Paykin
Galois – Portland, US
- Robert Rand
University of Maryland – College Park, US
- Mathys Rennela
CWI – Amsterdam, NL
- Francisco Rios
Dalhousie University – Halifax, CA
- Martin Roetteler
Microsoft Corporation – Redmond, US
- Neil Julien Ross
Dalhousie University – Halifax, CA
- Bruno Schmitt Antunes
EPFL – Lausanne, CH
- Peter Selinger
Dalhousie University – Halifax, CA
- Mathias Soeken
EPFL – Lausanne, CH
- Damian Steiger
ETH Zürich, CH
- Rainer Steinwandt
Florida Atlantic University – Boca Raton, US
- Benoit Valiron
Centrale Supelec – Orsay, FR
- Rodney Van Meter
Keio University – Fujisawa, JP
- Michael Walter
University of Amsterdam, NL
- Robert Wille
Johannes Kepler Universität Linz, AT
- Shigeru Yamashita
Ritsumeikan University – Shiga, JP
- Mingsheng Ying
University of Technology – Sydney, AU
- Vladimir Zamdzhiev
LORIA – Nancy, FR
- Margherita Zorzi
University of Verona, IT
- Alwin Zulehner
Johannes Kepler Universität Linz, AT
- Paolo Zuliani
Newcastle University, GB

Report from Dagstuhl Seminar 18391

Algebraic Methods in Computational Complexity

Edited by

Markus Bläser[1], Valentine Kabanets[2], Jacobo Torán[3], and Christopher Umans[4]

1 **Universität des Saarlandes, DE,** mblaeser@cs.uni-saarland.de
2 **Simon Fraser University – Burnaby, CA,** kabanets@cs.sfu.ca
3 **Universität Ulm, DE,** jacobo.toran@uni-ulm.de
4 **Caltech – Pasadena, US,** umans@cs.caltech.edu

Abstract

Computational Complexity is concerned with the resources that are required for algorithms to detect properties of combinatorial objects and structures. It has often proven true that the best way to argue about these combinatorial objects is by establishing a connection (perhaps approximate) to a more well-behaved algebraic setting. Indeed, many of the deepest and most powerful results in Computational Complexity rely on algebraic proof techniques. The Razborov-Smolensky polynomial-approximation method for proving constant-depth circuit lower bounds, the PCP characterization of NP, and the Agrawal-Kayal-Saxena polynomial-time primality test are some of the most prominent examples.

In some of the most exciting recent progress in Computational Complexity the algebraic theme still plays a central role. There have been significant recent advances in algebraic circuit lower bounds, and the so-called chasm at depth 4 suggests that the restricted models now being considered are not so far from ones that would lead to a general result. There have been similar successes concerning the related problems of polynomial identity testing and circuit reconstruction in the algebraic model (and these are tied to central questions regarding the power of randomness in computation). Also the areas of derandomization and coding theory have experimented important advances.

The seminar aimed to capitalize on recent progress and bring together researchers who are using a diverse array of algebraic methods in a variety of settings. Researchers in these areas are relying on ever more sophisticated and specialized mathematics and the goal of the seminar was to play an important role in educating a diverse community about the latest new techniques.

Seminar September 23–28, 2018 – http://www.dagstuhl.de/18391

1998 ACM Subject Classification Theory of computation → Algebraic complexity theory, Theory of computation → Problems, reductions and completeness, Theory of computation → Circuit complexity

Keywords and phrases computational complexity, algebra, (de-) randomization, circuits, coding, lower bounds

Digital Object Identifier 10.4230/DagRep.8.9.133

Algebraic Methods in Computational Complexity, *Dagstuhl Reports*, Vol. 8, Issue 09, pp. 133–153
Editors: Markus Bläser, Valentine Kabanets, Jacobo Torán, and Christopher Umans
Dagstuhl Reports
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Executive Summary

Markus Bläser
Valentine Kabanets
Jacobo Torán
Christopher Umans

The seminar brought together more than 40 researchers covering a wide spectrum of complexity theory. The focus on algebraic methods showed the great importance of such techniques for theoretical computer science. We had 24 talks, most of them lasting about 45 minutes, leaving ample room for discussions. We also had a much appreciated rump session on Tuesday evening in which Antonina Kolokolova, Bill Gasarch, Lance Fortnow, Chandran Saha, William Hoza, Neeraj Kajal and Arpita Korwar presented some open questions. In the following we describe the major topics of discussion in more detail.

Circuit Complexity

This is an area of fundamental importance to Complexity. Circuits studied from many different perspectives were one of the main topics in the seminar. *Eric Allender* gave an overview of the Minimum Circuit Size Problem (MCSP): given the truth-table for a Boolean function, what is the size of the minimum circuit computing it? In his talk he mentioned some interesting results proving that some low complexity classes cannot be reduced to the problem of computing superlinear approximations to circuit size.

Arithmetic circuits and formulas are a special computation model that uses + and × as operators for computing polynomials instead of Boolean operations. *Nutan Limaye* presented a depth hierarchy theorem for this model showing that there is a polynomial computed by a depth $D+1$ polynomial sized multilinear formula such that any depth D multilinear formula computing the polynomial must have exponential size.

Chandan Saha considered a further restriction to depth three circuits C computing a polynomial $f = T_1 + T_2 + \cdots + T_s$, where each T_i is a product of d linear forms in n variables. He presented a randomized algorithm to reconstruct non-degenerate homogeneous depth three circuits, for the case $n > (3d)^2$, given black-box access to f. The algorithm works in polynomial time in n, s and d.

Depth-2 circuits with polynomial size and linear threshold functions were presented by *Meena Mahajan*. She surveyed the landscape below these circuits and present one new result concerning decision lists.

Algebraic Complexity

There were also several presentations discussing the complexity of several problems over algebraic structures.

Nitin Saxena considered in his talk the problem of testing whether a set F of polynomials given as algebraic circuits has an algebraic dependence. He showed that this problem can be computed in AM ∩ coAM thus solving an open question from 2007.

Problems related to the minimum code-word problem and the existence of non-trivial automorphism moving few vertices in graphs or hypergraphs, were presented by *V. Arvind* in his talk. He discuss the parameterized complexity of this and related algebraic problems.

Josh Alman gave an interesting talk on Matrix Multiplication (MM). He surveyed the two main approaches for MM algorithms: the Laser method of Strassen, and the Group theoretic approach of Cohn and Umans and defined a generalization which subsumes these two approaches. He then explained ways to obtain lower bounds for algorithms for MM when using these algorithmic methods.

Rohit Gurjar studied the class of matrices A for which the lattice $L(A)$ formed by all integral vectors v in the null-space of A, has only polynomially many near-shortest vectors. He proved that this is the case when the matrix A is totally unimodular (all sub-determinants are 0, $+1$, or -1). As a consequence he could show a deterministic algorithm for PIT for any polynomial of the form $\det(\sum x_i A_i)$ for rank-1 matrices A_i.

Pseudo-Randomness and Derandomization

Derandomization is an area where there are tight connections between lower bounds and algorithms. Strong enough circuit lower bounds can be used to construct pseudo-random generators that can then be used to deterministically simulate randomized algorithms. A central question in derandomization is whether randomized logspace RL equals deterministic logspace L. To show that RL = L, it suffices to construct explicit pseudorandom generators that fool polynomial-size read-once (oblivious) branching programs (roBPs). There were two talks related to this question. *Michael Forbes* presented a method to obtain an explicit PRG with seed-length $O(\log^3 n)$ for polynomial-size roBPs reading their bits in an unknown order. *William Hoza* gave an explicit hitting set generator for read-once branching programs with known variable order. As a corollary of this construction, it follows that every RL algorithm that uses r random bits can be simulated by an NL algorithm that uses only $O(r/\log^c n)$ nondeterministic bits, where c is an arbitrarily large constant. Another consequence of the result is that any RL algorithm with small success probability ϵ can be simulated deterministically in space $O(\log^{3/2} n + \log n \log\log(1/\epsilon))$.

A hitting set is a set of instances such that every non-zero polynomial in the model has a non-root in the set. This would solve the Polynomial Identity Testing problem (PIT) in that model. *Ramprasad Saptharishi* showed that by barely improving the trivial $(s+1)^n$ size hitting set even for n-variate degree s, size s algebraic circuits, we could get an almost complete derandomization of PIT.

In a second talk, *William Hoza* talked about the possibility of derandomizing an algorithm by using randomness from the input itself. For a language L with a bounded-error randomized algorithm in space S and time $n \cdot \mathit{poly}(S)$ he gave a randomized algorithm for L with the same time and space resources but using only $O(S)$ random bits; the algorithm has a low failure probability on all but a negligible fraction of inputs of each length.

Andrej Bogdanov considered the problem of extracting true randomness from a set biased dice (Santha-Vazirani sources). He presented a recent result in which he completely classified all non-trivial randomness sources of this type into: non-extractable ones, extractable from polynomially many samples, and extractable from an logarithmically many samples (in the inverse of the error).

Coding Theory

Error-correcting codes and other kinds of codes, particularly those constructed from polynomials, i.e. Reed-Solomon codes or Reed-Muller codes, lie at the heart of many significant results in Computational Complexity. This is an area in which the relation between different areas of complexity, like the analysis of algebraic structures or derandomization becomes especially fruitful.

Greatly improving previously known constructions for an odd size alphabet, *Michal Koucký* presented a construction of quasi-Gray codes of dimension n and length 3^n over the ternary alphabet $\{0, 1, 2\}$ with worst-case read complexity $O(\log n)$ and write complexity 2. This generalizes to arbitrary odd-size alphabets. These results were obtained via a novel application of algebraic tools together with the principles of catalytic computation.

Noga Ron-Zewi presented a very recent result showing that Folded Reed-Solomon codes achieve list decoding capacity with constant list sizes, independent of the block length. She explained that multiplicity codes exhibit similar behavior, and used this to obtain capacity achieving locally list decodable codes with query complexity significantly lower than previous constructions.

Binary error correcting code with relative distance $(1-\epsilon)/2$ and relative rate $\epsilon^{2+o(1)}$ were explained in one of the talks given by *Amnon Ta-Shma*. Previous explicit constructions had rate about ϵ^3. The main tool used for this construction are *Parity Samplers*. He explained how to get better explicit parity samplers using a variant of the zig-zag product.

In his second talk, Amnon talked about $(1-\tau, L)$ erasure list-decodable codes. He presented a recent work where he constructed for the first time an explicit binary $(1-\tau, L)$ erasure list-decodable code having rate $\tau^{1+\gamma}$ (for any constant $\gamma > 0$ and τ small enough) and list-size $\mathrm{poly}(\log 1/\tau)$, exhibiting an explicit non-linear code that provably beats the best possible linear one. The main ingredient in his construction is a new (and almost-optimal) *unbalanced* two-source extractor.

Quantum Complexity

Complexity issues arising in the context of quantum computation are an important area in Complexity Theory since several decades. In this workshop we had one talk on this topic. *Sevag Gharibian* talked about quantum versions of the classical k-SAT problem. He talked about the problem of computing satisfying assignments to k-QSAT instances which have a "matching" or "dimer covering"; this is an NP problem whose decision variant is trivial, but whose search complexity remains open. He presented a parameterized algorithm for k-QSAT instances from a non-trivial class, which allows to obtain exponential speedups over brute force methods.

Conclusion

As is evident from the list above, the talks ranged over a broad assortment of subjects with the underlying theme of using algebraic and combinatorial techniques. It was a very fruitful meeting and has hopefully initiated new directions in research. Several participants specifically mentioned that they appreciated the particular focus on a common class of *techniques* (rather than end results) as a unifying theme of the workshop. We look forward to our next meeting!

2 Table of Contents

Executive Summary

Overview of Talks

3 Overview of Talks

3.1 The Non-Hardness of Approximating Circuit Size

Eric Allender (Rutgers University – Piscataway, US)

The Minimum Circuit Size Problem (MCSP) has been the focus of intense study recently; MCSP is hard for SZK under rather powerful reductions, and is provably not hard under "local" reductions computable in TIME($n^{0.49}$). The question of whether MCSP is NP hard (or indeed, hard even for small subclasses of P) under some of the more familiar notions of reducibility (such as many-one or Turing reductions computable in polynomial time or in AC^0) is closely related to many of the longstanding open questions in complexity theory.

All known hardness results for MCSP hold also for computing somewhat weak approximations to the circuit complexity of a function. Some of these results were proved by exploiting a connection to a notion of time-bounded Kolmogorov complexity (KT) and the corresponding decision problem (MKTP). More recently, a new approach for proving improved hardness results for MKTP was developed, but this approach establishes only hardness of extremely good approximations of the form $1 + o(1)$, and these improved hardness results are not yet known to hold for MCSP. In particular, it is known that MKTP is hard for the complexity class DET under nonuniform AC^0-many-one reductions, implying that MKTP is not in $\mathrm{AC}^0[p]$ for any prime p. It is still open if similar circuit lower bounds hold for MCSP. One possible avenue for proving a similar hardness result for MCSP would be to improve the hardness of approximation for MKTP beyond $1 + o(1)$ to $\omega(1)$. In this paper, we show that this is impossible.

More specifically, we prove that PARITY does not reduce to the problem of computing superlinear approximations to KT-complexity or circuit size via AC^0-Turing reductions that make $O(1)$ queries. This is significant, since it is known that just ONE query to a much worse approximation of circuit size or KT-complexity suffices, for an AC^0 reduction to compute an approximation to any set in P/poly. For weaker approximations, we also prove non-hardness results for more powerful reductions. Our non-hardness results are unconditional, in contrast to conditional results presented in earlier work of [Allender, Hirahara] (for more powerful reductions, but for much worse approximations). This also highlights obstacles that would have to be overcome by any proof that MKTP or MCSP is hard for NP under AC^0 reductions. It may also be a step toward confirming a conjecture of Murray and Williams, that MCSP is not NP-complete under logtime-uniform AC^0-many-one reductions.

3.2 Limits on All Known (and Some Unknown) Approaches to Matrix Multiplication

Josh Alman (MIT – Cambridge, US)

Joint work of Josh Alman, Virginia Vassilevska Williams
Main reference Josh Alman, Virginia Vassilevska Williams: "Further Limitations of the Known Approaches for Matrix Multiplication", in Proc. of the 9th Innovations in Theoretical Computer Science Conference, ITCS 2018, January 11-14, 2018, Cambridge, MA, USA, LIPIcs, Vol. 94, pp. 25:1–25:15, Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2018.
URL https://doi.org/10.4230/LIPIcs.ITCS.2018.25

We study the known techniques for designing Matrix Multiplication (MM) algorithms. The two main approaches are the Laser method of Strassen, and the Group theoretic approach of Cohn and Umans. We define a generalization based on zeroing outs which subsumes these two approaches, which we call the Solar method, and an even more general method based on monomial degenerations, which we call the Galactic method.

We then design a suite of techniques for proving lower bounds on the value of omega, the exponent of MM, which can be achieved by algorithms using many tensors T and the Galactic method. Some of our techniques exploit "local" properties of T, like finding a sub-tensor of T which is so "weak" that T itself couldn't be used to achieve a good bound on ω, while others exploit "global" properties, like T being a monomial degeneration of the structural tensor of a group algebra. Our main result is that there is a universal constant $c > 2$ such that a large class of tensors generalizing the Coppersmith-Winograd tensor CW_q cannot be used within the Galactic method to show a bound on ω better than c, for any q.

3.3 The Complexity of Computing Small Weight Graph Automorphisms

V. Arvind (Institute of Mathematical Sciences – Chennai, IN)

Joint work of V. Arvind, Johannes Köbler, Sebastian Kuhnert, Jacobo Torán
Main reference Vikraman Arvind, Johannes Köbler, Sebastian Kuhnert, Jacobo Torán: "Parameterized Complexity of Small Weight Automorphisms", in Proc. of the 34th Symposium on Theoretical Aspects of Computer Science, STACS 2017, March 8-11, 2017, Hannover, Germany, LIPIcs, Vol. 66, pp. 7:1–7:13, Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2017.
URL https://doi.org/10.4230/LIPIcs.STACS.2017.7

Given a graph (or hypergraph) G as input and a parameter k, does G have an automorphism of weight exactly k? We discuss the parameterized complexity of this and related problems, and also connections to the minimum weight codeword problem showing some cases in which the problems are fixed parameter tractable. As a building block for our algorithms, we generalize Schweitzer's FPT algorithm [ESA 2011] that, given two graphs on the same vertex set and a parameter k, decides whether there is an isomorphism between the two graphs that moves at most k vertices. We extend this result to hypergraphs, using the maximum hyperedge size as a second parameter. Another key component of our algorithm is an orbit-shrinking technique that preserves permutations that move few points and that may be of independent interest. Applying it to a suitable subgroup of the automorphism group allows us to switch from bounded hyperedge size to bounded color classes in the exactly-k case.

3.4 Optimal Extractors for Generalized Santha-Vazirani Sources

Andrej Bogdanov (The Chinese University of Hong Kong, HK)

Joint work of Salman Beigi, Andrej Bogdanov, Omid Etesami, Siyao Guo
Main reference Salman Beigi, Andrej Bogdanov, Omid Etesami, Siyao Guo: "Optimal Deterministic Extractors for Generalized Santha-Vazirani Sources", in Proc. of the Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2018, August 20-22, 2018 – Princeton, NJ, USA, LIPIcs, Vol. 116, pp. 30:1–30:15, Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2018.
URL https://doi.org/10.4230/LIPIcs.APPROX-RANDOM.2018.30

Take a finite set of biased dice that share some common faces. An adversary repeatedly tosses them, with each choice of die possibly depending on the previous outcomes. Can you extract true randomness? In 1986 Santha and Vazirani gave a negative answer when the dice are (two-sided) coins. In 2015 Beigi, Etesami, and Gohari showed how to obtain an almost-unbiased bit for other sets of dice. The sample complexity of their extractor is polynomial in the inverse of the error. We completely classify all non-trivial randomness sources of this type into: (1) non-extractable ones; (2) extractable from polynomially many samples; and (3) extractable from an logarithmically many samples (in the inverse of the error). The extraction algorithms are efficient and easy to describe. I will discuss the relevance to distributed and cryptographic computation from imperfect randomness and point out some open questions in this context.

3.5 Degree vs Sparsity of Flat Polynomials that Approximate Boolean Functions

Sourav Chakraborty (Indian Statistical Institute – Kolkata, IN)

Various conjectures and theorems about the Fourier spectrum of Boolean functions impose various constraints of what type of polynomials can approximate a Boolean function. One such conjecture is the Fourier Entropy Influence Conjecture (FEI). As an implication of the conjecture we can observe a relation between the degree and sparsity of any polynomial that approximates a Boolean function. Can we prove these implications directly without using the conjecture? This question is related to the B-H conjecture in mathematics, which can be thought of as a generalised balancing lights problem.

3.6 A PSPACE Construction of a Hitting Set for the Closure of Small Algebraic Circuits

Michael A. Forbes (University of Illinois – Urbana-Champaign, US)

Joint work of Michael A. Forbes, Amir Shpilka
Main reference Michael A. Forbes, Amir Shpilka: "A PSPACE Construction of a Hitting Set for the Closure of Small Algebraic Circuits", CoRR, Vol. abs/1712.09967, 2017.
URL https://arxiv.org/abs/1712.09967

In this paper we study the complexity of constructing a hitting set for the closure of VP, the class of polynomials that can be infinitesimally approximated by polynomials that are computed by polynomial sized algebraic circuits, over the real or complex numbers. Specifically, we show that there is a PSPACE algorithm that given n, s, r in unary outputs a set of n-tuples over the rationals of size $\mathrm{poly}(n, s, r)$, with $\mathrm{poly}(n, s, r)$ bit complexity, that hits all n-variate polynomials of degree-r that are the limit of size-s algebraic circuits. Previously it was known that a random set of this size is a hitting set, but a construction that is certified to work was only known in EXPSPACE (or EXPH assuming the generalized Riemann hypothesis). As a corollary we get that a host of other algebraic problems such as Noether Normalization Lemma, can also be solved in PSPACE deterministically, where earlier only randomized algorithms and EXPSPACE algorithms (or EXPH assuming the generalized Riemann hypothesis) were known. The proof relies on the new notion of a robust hitting set which is a set of inputs such that any nonzero polynomial that can be computed by a polynomial size algebraic circuit, evaluates to a not too small value on at least one element of the set. Proving the existence of such a robust hitting set is the main technical difficulty in the proof. Our proof uses anti-concentration results for polynomials, basic tools from algebraic geometry and the existential theory of the reals.

3.7 Pseudorandom Generators for Read-Once Branching Programs, in any Order

Michael A. Forbes (University of Illinois – Urbana-Champaign, US)

Joint work of Michael A. Forbes, Zander Kelley
Main reference Michael A. Forbes, Zander Kelley: "Pseudorandom Generators for Read-Once Branching Programs, in any Order", CoRR, Vol. abs/1808.06265, 2018.
URL https://arxiv.org/abs/1808.06265

A central question in derandomization is whether randomized logspace (RL) equals deterministic logspace (L). To show that RL = L, it suffices to construct explicit pseudorandom generators (PRGs) that fool polynomial-size read-once (oblivious) branching programs (roBPs). Starting with the work of Nisan, pseudorandom generators with seed-length $O(\log^2 n)$ were constructed. Unfortunately, improving on this seed-length in general has proven challenging and seems to require new ideas. A recent line of inquiry has suggested focusing on a particular limitation of the existing PRGs, which is that they only fool roBPs when the variables are read in a particular known order, such as $x_1 < \cdots < x_n$. In comparison, existentially one can obtain logarithmic seed-length for fooling the set of polynomial-size roBPs that read the variables under any fixed unknown permutation $x_{\pi(1)} < \cdots < x_{\pi(n)}$. While recent works have established novel PRGs in this setting for subclasses of roBPs, there were no known $n^{o(1)}$

seed-length explicit PRGs for general polynomial-size roBPs in this setting. In this work, we follow the "bounded independence plus noise" paradigm of Haramaty, Lee and Viola, and give an improved analysis in the general roBP unknown-order setting. With this analysis we obtain an explicit PRG with seed-length $O(\log^3 n)$ for polynomial-size roBPs reading their bits in an unknown order. Plugging in a recent Fourier tail bound of Chattopadhyay, Hatami, Reingold, and Tal, we can obtain a $\widetilde{O}(\log^2 n)$ seed-length when the roBP is of constant width.

3.8 The Muffin Problem: Complexity Questions

William Gasarch (University of Maryland – College Park, US)

Joint work of Guangqi Cui, John Dickerson, Naveen Durvasula, William Gasarch, Erik Metz, Jacob Prinz, Naveen Raman, Daniel Smolyak, Sung Hyun Yoo
Main reference Guangqi Cui, John P. Dickerson, Naveen Durvasula, William Gasarch, Erik Metz, Jacob Prinz, Naveen Raman, Daniel Smolyak, Sung Hyun Yoo: "A Muffin-Theorem Generator", in Proc. of the 9th International Conference on Fun with Algorithms, FUN 2018, June 13-15, 2018, La Maddalena, Italy, LIPIcs, Vol. 100, pp. 15:1–15:19, Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2018.
URL https://doi.org/10.4230/LIPIcs.FUN.2018.15

Consider the following problem: You have m muffins and s students. You want to divide the muffins and give out pieces so that everyone gets m/s muffins. You can clearly divide each muffin in s pieces and give each person m s-sized pieces. Since students do not like crumbs we want to maximize the smallest piece. Let $f(m, s)$ be the size of the smallest piece in the procedure which maximizes the smallest piece.

We have proven many theorems and have many procedures to find $f(m, s)$. We have used these to obtain $f(m, s)$ for all $s \leq 60$ and $m \leq 70$. However, these procedures are somewhat ad-hoc.

- If s is fixed then, for $m \geq s^3$, $f(m, s)$ has an easy formula. So $f(m, s)$ is FPT.
- There is a Mixed Integer Program for $f(m, s)$ in $O(ms)$ variables. Note that the input is of size $\log m + \log s$.
- Is computing $f(m, s)$ in P? We do not know.
- Is computing $f(m, s)$ in NP (phrased as a set). We do not know.

3.9 On Efficiently Solvable Cases of Quantum k-SAT

Sevag Gharibian (Universität Paderborn, DE)

Joint work of Sevag Gharibian, Marco Aldi, Niel de Beaudrap, Seyran Saeedi
Main reference Marco Aldi, Niel de Beaudrap, Sevag Gharibian, Seyran Saeedi: "On Efficiently Solvable Cases of Quantum k-SAT", in Proc. of the 43rd International Symposium on Mathematical Foundations of Computer Science, MFCS 2018, August 27-31, 2018, Liverpool, UK, LIPIcs, Vol. 117, pp. 38:1–38:16, Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2018.
URL https://doi.org/10.4230/LIPIcs.MFCS.2018.38

The constraint satisfaction problems k-SAT and Quantum k-SAT (k-QSAT) are canonical NP-complete and QMA$_1$-complete problems (for $k \geq 3$), respectively, where QMA$_1$ is a quantum generalization of NP with one-sided error. Whereas k-SAT has been well-studied for special tractable cases, as well as from a parameterized complexity perspective, much less is known in similar settings for k-QSAT. Here, we study the open problem of computing satisfying assignments to k-QSAT instances which have a "matching" or "dimer covering"; this is an NP problem whose decision variant is trivial, but whose search complexity remains open.

Among other results, our main contribution is a parameterized algorithm for k-QSAT instances from a certain non-trivial class, which allows us to obtain exponential speedups over brute force methods in some cases. This is, to our knowledge, the first known such parameterized algorithm. The techniques behind our work stem from algebraic geometry, although no background in the topic is required for this presentation.

3.10 Number of near-shortest vectors in Lattices and Polynomial Identity Testing

Rohit Gurjar (Indian Institute of Technology – Mumbai, IN)

For a matrix A, consider the lattice $L(A)$ formed by all integral vectors v in the null-space of A. We ask for which matrices A, the lattice $L(A)$ has only polynomially many near-shortest vectors i.e., vectors whose length is close to the shortest length in $L(A)$. The motivation for this question comes from the fact that we can get a deterministic black-box polynomial identity testing algorithm for any polynomial whose newton polytope has faces described by matrices with the aforementioned property.

We show that when the matrix A is totally unimodular (all sub-determinants are 0, +1, or −1) then the lattice $L(A)$ has only polynomially many near-shortest vectors. The proof of this statement goes via a remarkable theorem of Seymour on a decomposition for totally unimodular matrices. The statement generalizes two earlier known results – the number of near-shortest cycles and the number of near-shorest cuts in a graph are poly-bounded. As a special case, we get PIT for any polynomial of the form $\det(\sum x_i A_i)$ for rank-1 matrices A_i.

3.11 Simple Optimal Hitting Sets for Small-Success RL

William Hoza (University of Texas – Austin, US)

Joint work of William Hoza, David Zuckerman

We give a simple explicit hitting set generator for read-once branching programs of width w and length r with known variable order. When $r = w$, our generator has seed length $O(\log^2 r + \log(1/\epsilon))$. When $r = \text{polylog } w$, our generator has optimal seed length $O(\log w + \log(1/\epsilon))$. For intermediate values of r, our generator's seed length smoothly interpolates between these two extremes.

Our generator's seed length improves on recent work by Braverman, Cohen, and Garg (STOC '18). In addition, our generator and its analysis are dramatically simpler than the work by Braverman et al. Our generator's seed length improves on all the classic generators for space-bounded computation (Nisan Combinatorica '92; Impagliazzo, Nisan, and Wigderson STOC '94; Nisan and Zuckerman JCSS '96) when ϵ is small.

As a corollary of our construction, we show that every RL algorithm that uses r random bits can be simulated by an NL algorithm that uses only $O(r/\log^c n)$ nondeterministic bits, where c is an arbitrarily large constant. Finally, we show that any RL algorithm with small success probability ϵ can be simulated deterministically in space $O(\log^{3/2} n+\log n \log\log(1/\epsilon))$. This improves on work by Saks and Zhou (JCSS '99), who gave an algorithm that runs in space $O(\log^{3/2} n + \sqrt{(\log n)\log(1/\epsilon)})$.

3.12 Typically-Correct Derandomization for Small Time and Space

William Hoza (University of Texas – Austin, US)

Main reference William M. Hoza: "Typically-Correct Derandomization for Small Time and Space", CoRR, Vol. abs/1711.00565, 2017.
URL http://arxiv.org/abs/1711.00565

Suppose a language L can be decided by a bounded-error randomized algorithm that runs in space S and time $n \cdot poly(S)$. We give a randomized algorithm for L that still runs in space $O(S)$ and time $n \cdot poly(S)$ that uses only $O(S)$ random bits; our algorithm has a low failure probability on all but a negligible fraction of inputs of each length. An immediate corollary is a deterministic algorithm for L that runs in space $O(S)$ and succeeds on all but a negligible fraction of inputs of each length. We also give several other complexity-theoretic applications of our technique.

3.13 Orbits of Monomials and Factorization into Products of Linear Forms

Pascal Koiran (ENS – Lyon, FR)

Joint work of Pascal Koiran, Nicolas Ressayre
Main reference Pascal Koiran, Nicolas Ressayre: "Orbits of monomials and factorization into products of linear forms", CoRR, Vol. abs/1807.03663, 2018.
URL http://arxiv.org/abs/1807.03663

This talk is devoted to the factorization of multivariate polynomials into products of linear forms, a problem which has applications to differential algebra, to the resolution of systems of polynomial equations and to Waring decomposition (i.e., decomposition in sums of d-th powers of linear forms; this problem is also known as symmetric tensor decomposition). We provide three black box algorithms for this problem. Our main contribution is an algorithm motivated by the application to Waring decomposition. This algorithm reduces the corresponding factorization problem to simultaenous matrix diagonalization, a standard task in linear algebra. The algorithm relies on ideas from invariant theory, and more specifically on Lie algebras. Our second algorithm reconstructs a factorization from several bi-variate projections. Our third algorithm reconstructs it from the determination of the zero set of the input polynomial, which is a union of hyperplanes.

3.14 Optimal Quasi-Gray Codes: The Alphabet Matters

Michal Koucký (Charles University – Prague, CZ)

Joint work of Diptarka Chakraborty, Debarati Das, Michal Koucký, Nitin Saurabh
Main reference Diptarka Chakraborty, Debarati Das, Michal Koucký, Nitin Saurabh: "Space-Optimal Quasi-Gray Codes with Logarithmic Read Complexity", in Proc. of the 26th Annual European Symposium on Algorithms, ESA 2018, August 20-22, 2018, Helsinki, Finland, LIPIcs, Vol. 112, pp. 12:1–12:15, Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2018.
URL http://dx.doi.org/10.4230/LIPIcs.ESA.2018.12

A quasi-Gray code of dimension n and length ℓ over an alphabet A is a sequence of distinct words $w_1, w_2, \dots, w_e$ from A^n such that any two consecutive words differ in at most c coordinates, for some fixed constant $c > 0$. In this talk we are interested in the read and write complexity of quasi-Gray codes in the bit-probe model, where we measure the number of symbols read and written in order to transform any word w_i into its successor w_{i+1}.

We present construction of quasi-Gray codes of dimension n and length 3^n over the ternary alphabet $\{0, 1, 2\}$ with worst-case read complexity $O(\log n)$ and write complexity 2. This generalizes to arbitrary odd-size alphabets. For the binary alphabet, we present quasi-Gray codes of dimension n and length at least $2^n - -20n$ with worst-case read complexity $6 + \log n$ and write complexity 2. This complements a recent result by Raskin (2017) who shows that any quasi-Gray code over binary alphabet of length 2^n has read complexity $\Omega(n)$.

Our results significantly improve on previously known constructions and for the odd-size alphabets we break the $\Omega(n)$ worst-case barrier for space-optimal (non-redundant) quasi-Gray codes with constant number of writes. We obtain our results via a novel application of algebraic tools together with the principles of catalytic computation [Buhrman et al. '14, Ben-Or and Cleve '92, Barrington '89, Coppersmith and Grossman '75].

3.15 A Near-Optimal Depth-Hierarchy Theorem for Small-Depth Multilinear Circuits

Nutan Limaye (Indian Institute of Technology – Mumbai, IN)

Joint work of Suryajith Chillara, Christian Engels, Nutan Limaye, Srikanth Srinivasan
Main reference Suryajith Chillara, Christian Engels, Nutan Limaye, Srikanth Srinivasan: "A Near-Optimal Depth-Hierarchy Theorem for Small-Depth Multilinear Circuits", CoRR, Vol. abs/1804.02520, 2018.
URL http://arxiv.org/abs/1804.02520

The field of Computational Complexity deals with the study of resources necessary and sufficient for computations. One classical theme well-studied in the literature deals with quantifying the additional power gained by a model of computation with extra resources. For instance one could ask: does a Turing machine that runs for T steps necessarily compute more functions than the machines that only run for $o(T)$ steps? In general, does more resources mean more power? A hierarchy theorem is exactly such a statement for a model of computation and a resource.

The Time Hierarchy Theorem, Space Hierarchy theorem and many more such theorems for the Turing machines are classical results in Computational Complexity theory. In this work the model of computation we focus on is arithmetic formulas. An arithmetic formula is a natural model of computation for polynomials. It uses + and × as operators for computing polynomials. The size of the formula is the number of such operators it uses. The depth of the formula is the longest input to output path in the formula. Here we provide a depth hierarchy theorem for multilinear arithmetic formulas, where a formula is said to be multilinear if each gate in it computes a multilinear polynomial.

Here we show that there is a polynomial computed by depth $D+1$ polynomial sized multilinear formula such that any depth D multilinear formula computing the polynomial must have exponential size. In particular, we show that for every $D \leq o(\log n / \log\log n)$, there is a polynomial P_D on n variables that can be computed by a multilinear formula of depth $D+1$ and size $O(n)$ but cannot be computed by any multilinear formula of depth D and size $\exp(n^{1/D})$. This strengthens the result of Raz and Yehudayoff (Computational Complexity 2009) who showed a quasipolynomial separation, and the result of Kayal, Nair and Saha (STACS 2016) who gave an exponential separation when $D=3$. Our separating examples may be viewed as algebraic analogues of variants of the Graph Reachability problem studied by Chen, Oliveira, Servedio and Tan (STOC 2016), who used them to prove lower bounds for constant-depth Boolean circuits.

3.16 Locating linear decision lists within TC^0

Meena Mahajan (Institute of Mathematical Sciences – Chennai, IN)

Joint work of Arkadev Chattopadhyay, Meena Mahajan, Nikhil Mande, Nitin Saurabh

Polynomial-size depth-2 circuits with linear threshold functions at each gate lie at the frontier of known circuit lower bounds. In this talk I will briefly survey the landscape below these circuits – the very-low-depth threshold hierarchy – and present one new result concerning decision lists, obtained jointly with Arkadev Chattopadhyay, Nikhil Mande and Nitin Saurabh. I will also describe a (somewhat related) question from proof complexity.

3.17 Improved List Decoding of Algebraic Codes

Noga Ron-Zewi (University of Haifa, IL)

Joint work of Swastik Kopparty, Noga Ron-Zewi, Shubhangi Saraf, Mary Wootters
Main reference To appear in FOCS 2018

We show that Folded Reed-Solomon codes achieve list decoding capacity with constant list sizes, independent of the block length. Prior work yielded list sizes that are polynomial in the block length, and relied on elaborate subspace evasive machinery to reduce the list sizes to constant.

We further show that multiplicity codes exhibit similar behavior, and use this to obtain capacity achieving locally list decodable codes with query complexity significantly lower than was known before.

3.18 Proper Learning of Non-degenerate Homogeneous Depth Three Arithmetic Circuits

Chandan Saha (Indian Institute of Science – Bangalore, IN)

Joint work of Neeraj Kayal, Chandan Saha

A homogeneous depth three circuit C computes a polynomial $f = T_1 + T_2 + \cdots + T_s$, where each T_i is a product of d linear forms in n variables. Given black-box access to f, can we efficiently reconstruct (i.e. proper learn) a homogeneous depth three circuit computing f? Learning homogeneous depth three circuits is stated as an open problem in a work by Klivans and Shpilka (COLT 2003).

We give a randomized poly(n, d, s) time algorithm to reconstruct non-degenerate homogeneous depth three circuits, if $n > (3d)^2$. The algorithm works over any field F, provided char$(F) = 0$ or greater than poly(nds). Loosely speaking, a circuit C is non-degenerate if the dimension of the partial derivative (similarly, shifted partial derivative) space of f equals the sum of the dimensions of the partial derivative (resp., shifted partial derivative) spaces of the terms $T_1, \ldots, T_s$; in this sense, the terms are “independent” of each other. A random homogeneous depth three circuit (chosen according to any reasonable distribution) is almost surely non-degenerate. Previous learning algorithms for homogeneous depth three circuits are either improper (with an exponential dependence on d), or they work for constant s (with a doubly exponential dependence on s).

Our algorithm hinges on simultaneous block-diagonalization of a basis of the shifted differential operator space that acts on the partials of f. The block-diagonalization yields a decomposition of the partial derivative space of f into subspaces which, in turn, leads to the terms of C via another application of shifts. To our knowledge, this is the first time shifted partial derivative has been used to make progress on reconstruction algorithms.

3.19 Near Optimal Bootstrapping for Algebraic Models

Ramprasad Saptharishi (TIFR Mumbai, IN)

Joint work of Mrinal Kumar, Ramprasad Saptharishi, Anamay Tengse
Main reference Mrinal Kumar, Ramprasad Saptharishi, Anamay Tengse: “Near-optimal Bootstrapping of Hitting Sets for Algebraic Circuits”, Electronic Colloquium on Computational Complexity (ECCC), Vol. 25, p. 132, 2018.
URL https://eccc.weizmann.ac.il/report/2018/132

The classical lemma of Ore-DeMillo-Lipton-Schwartz-Zippel states that any nonzero polynomial $f(x_1, \ldots, x_n)$ of degree at most s will evaluate to a nonzero value at some point on a grid S^n in F^n with $|S| > s$. Thus, there is an explicit hitting set for all n-variate degree s, size s algebraic circuits of size $(s+1)^n$.

In this paper, we prove the following results:

- Let $\epsilon > 0$ be a constant. For a sufficiently large constant n and all $s \geq n$, if we have an explicit hitting set of size $(s+1)^{n-\epsilon}$ for the class of n-variate degree s polynomials that are computable by algebraic circuits of size s, then for all s, we have an explicit hitting set of size $s^{\exp\exp(O(\log^* s))}$ for s-variate circuits of degree s and size s. That is, if we can obtain a barely non-trivial exponent compared to the trivial $(s+1)^n$ sized hitting set even for constant variate circuits, we can get an almost complete derandomization of PIT.

- The above result holds when “circuits” are replaced by “formulas” or “algebraic branching programs”.

This extends a recent surprising result of Agrawal, Ghosh and Saxena (STOC 2018) who proved the same conclusion for the class of algebraic circuits, if the hypothesis provided a hitting set of size at most $(s^{n^{0.5--\epsilon}})$ (where $\epsilon > 0$ is any constant). Hence, our work significantly weakens the hypothesis of Agrawal, Ghosh and Saxena to only require a slightly non-trivial saving over the trivial hitting set, and also presents the first such result for algebraic branching programs and formulas.

3.20 Algebraic Dependence is Not Hard

Nitin Saxena (Indian Institute of Technology Kanpur, IN)

Joint work of Zeyu Guo, Nitin Saxena, Amit Sinhababu
Main reference Zeyu Guo, Nitin Saxena, Amit Sinhababu: “Algebraic Dependencies and PSPACE Algorithms in Approximative Complexity”, in Proc. of the 33rd Computational Complexity Conference, CCC 2018, June 22-24, 2018, San Diego, CA, USA, LIPIcs, Vol. 102, pp. 10:1–10:21, Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2018.
URL https://doi.org/10.4230/LIPIcs.CCC.2018.10

Testing whether a set F of polynomials has an algebraic dependence is a basic problem with several applications. The polynomials are given as algebraic circuits. Algebraic independence testing question is wide open over finite fields (Dvir, Gabizon, Wigderson, FOCS’07). In this work we put the problem in AM ∩ coAM. In particular, dependence testing is unlikely to be NP-hard. Our proof method is algebro-geometric, estimating the size of the image/preimage of the polynomial map F over the finite field. A gap in this size is utilized in the AM protocols.

Next, we introduce a new problem called *approximate* polynomials satisfiability (APS). We show that APS is NP-hard and, using projective algebraic-geometry ideas, we put APS in PSPACE (prior best was EXPSPACE via Gröbner bases). This has many unexpected applications to approximative complexity theory. This solves an open problem posed in (Mulmuley, FOCS’12, J. AMS 2017); greatly mitigating the GCT Chasm (exponentially in terms of space complexity).

3.21 Indistinguishability by Adaptive Procedures with Advice, and Lower Bounds on Hardness Amplification Proofs

Ronen Shaltiel (University of Haifa, IL)

Joint work of Aryeh Grinberg, Ronen Shaltiel, Emanuele Viola
Main reference Aryeh Grinberg, Ronen Shaltiel, Emanuele Viola: "Indistinguishability by adaptive procedures with advice, and lower bounds on hardness amplification proofs", Electronic Colloquium on Computational Complexity (ECCC) Vol. 25 p. 61, 2018.
URL https://eccc.weizmann.ac.il/report/2018/061

We study how well can q-query decision trees distinguish between the following two distributions: (i) $R = (R_1, \dots, R_N)$ that are i.i.d. indicator random variables, (ii) $X = (R|R \in A)$ where A is an event s.t. $\Pr[R \in A] \geq 2^{-a}$. We prove two lemmas:

Forbidden-set lemma: There exists $B \subseteq [N]$ of size $\mathrm{poly}(a, q, \frac{1}{\eta})$ such that q-query trees that do not query variables in B cannot distinguish X from R with advantage η.

Fixed-set lemma: There exists $B \subseteq [N]$ of size $\mathrm{poly}(a, q, \frac{1}{\eta})$ and $v \in B^B$ such that q-query trees do not distinguish $(X|X_B = v)$ from $(R|R_B = v)$ with advantage η.

The first can be seen as an extension of past work by Edmonds, Impagliazzo, Rudich and Sgall (Computational Complexity 2001), Raz (SICOMP 1998), and Shaltiel and Viola (SICOMP 2010) to *adaptive* decision trees. It is independent of recent work by Meir and Wigderson (ECCC 2017) bounding the number of $i \in [N]$ for which there exists a q-query tree that predicts X_i from the other bits.

We use the second, fixed-set lemma to prove lower bounds on black-box proofs for hardness amplification that amplify hardness from δ to $\frac{1}{2} - \epsilon$. Specifically:

- Reductions must make $q = \Omega(\log(1/\delta)/\epsilon^2)$ queries, implying a "size loss factor" of q. We also prove the lower bound $q = \Omega(\log(1/\delta)/\epsilon)$ for "error-less" hardness amplification proofs, and for direct-product lemmas. These bounds are tight.
- Reductions can be used to compute Majority on $\Omega(1/\epsilon)$ bits, implying that black box proofs cannot amplify hardness of functions that are hard against constant depth circuits (unless they are allowed to use Majority gates).

Both items extend to pseudorandom-generator constructions.

These results prove 15-year-old conjectures by Viola, and improve on three incomparable previous works (Shaltiel and Viola, SICOMP 2010; Gutfreund and Rothblum, RANDOM 2008; Artemenko and Shaltiel, Computational Complexity 2014).

3.22 Memory Augmented Markovian Walks and Explicit Parity Samplers Giving Almost Optimal Binary Codes

Amnon Ta-Shma (Tel Aviv University, IL)

I will show an explicit construction of a binary error correcting code with relative distance $(1 - \epsilon)/2$ and relative rate $\epsilon^{2+o(1)}$. This comes close to the Gilbert-Varshamov bound that shows such codes with rate ϵ^2 exist, and the LP lower bound that shows rate $\epsilon^2/\log(1/\epsilon)$ is necessary. Previous explicit constructions had rate about ϵ^3, and this is the first explicit construction to get that close to the Gilbert-Varshamov bound.

The main tool we use are "Parity Samplers". A parity sampler is a collection of sets $S_i \subset \Lambda$ with the property that for every "test" $A \subset \Lambda$ of a given constant density ϵ_0, the probability a set S_i from the collection falls into the test set A an *even* number of times is about half. A *sparse* parity sampler immediately implies a good code with distance close to $1/2$. The complete t-complex of all sequences of cardinality t is a good parity sampler, but with too many sets in the collection. Rozenman and Wigderson, and independently Alon, used random walks over expanders to explicitly construct sparse parity samplers, and their construction implies explicit codes with relative rate ϵ^4.

In the last part of the talk I will explain how one can get better explicit parity samplers (and therefore also better explicit codes) using a variant of the zig-zag product. In the random walk sampler, there exist many sets with substantial overlap. One way to look at the zig-zag product is that it takes a sub-collection of the random walk sampler, and this sub-collection has a smaller overlap between sets in the collection. The zig-zag product achieves that by keeping a small internal state. I will show that by enlarging the internal state one can further reduce the overlap, and as a result improve the quality of the parity sampler. One may view this process as a memory augmented Markovian process.

3.23 Near-Optimal Strong Dispersers and Erasure List-Decodable Codes

Amnon Ta-Shma (Tel Aviv University, IL)

Joint work of Avraham Ben-Aroya, Dean Doron, Amnon Ta-Shma
Main reference Avraham Ben-Aroya, Dean Doron, Amnon Ta-Shma: "Near-Optimal Strong Dispersers, Erasure List-Decodable Codes and Friends", Electronic Colloquium on Computational Complexity (ECCC), Vol. 25, p. 65, 2018.
URL https://eccc.weizmann.ac.il/report/2018/065

A code C is $(1-\tau, L)$ erasure list-decodable if for every word w, after erasing any $1-\tau$ fraction of the symbols of w, the remaining tau-fraction of its symbols have at most L possible completions into codewords of C. Non-explicitly, there exist binary $(1-\tau, L)$ erasure list-decodable codes having rate $O(\tau)$ and tiny list-size $L = O(\log 1/\tau)$. Achieving either of these parameters explicitly is a natural open problem and was brought up in several prior works. While partial progress on the problem has been achieved, no explicit construction up to this work achieved rate better than $\Omega(\tau^2)$ or list-size smaller than $\Omega(1/\tau)$ (for τ small enough). Furthermore, Guruswami showed that no *linear* code can have list-size smaller than $\Omega(1/\tau)$. In this work we construct an explicit binary $(1-\tau, L)$ erasure list-decodable code having rate $\tau^{1+\gamma}$ (for any constant $\gamma > 0$ and τ small enough) and list-size $\mathrm{poly}(\log 1/\tau)$, answering simultaneously both questions, and exhibiting an explicit non-linear code that provably beats the best possible linear one.

The binary erasure list-decoding problem is equivalent to the construction of explicit, low-error, strong dispersers outputting one bit with minimal entropy-loss and seed-length. Specifically, such dispersers with error ϵ have an unavoidable entropy-loss of $\log\log 1/\epsilon$ and seed-length at least $\log 1/\epsilon$. Similarly to the situation with erasure list-decodable codes, no explicit construction achieved seed-length better than $2\log 1/\epsilon$ or entropy-loss smaller than $2\log 1/\epsilon$, which are the best possible parameters for extractors. For every constant $\gamma > 0$ and every small ϵ, we explicitly construct an ϵ-error one-bit strong disperser with near-optimal seed-length $(1+\gamma)\log 1/\epsilon$ and near-optimal entropy-loss $O(\log\log 1/\epsilon)$.

The main ingredient in our construction is a new (and almost-optimal) *unbalanced* two-source extractor. The extractor extracts one bit with constant error from two independent sources, where one source has length n and tiny min-entropy $O(\log \log n)$ and the other source has length $O(\log n)$ and arbitrarily small constant min-entropy rate. The construction incorporates recent components and ideas from extractor theory with a delicate and novel analysis needed in order to solve dependency and error issues.

3.24 A Conditional Information Inequality and its Combinatorial Applications

Nikolay K. Vereshchagin (NRU Higher School of Economics – Moscow, RU)

Joint work of Tarik Kaced, Nikolay Vereshchagin
Main reference Tarik Kaced, Andrei E. Romashchenko, Nikolai K. Vereshchagin: “A Conditional Information Inequality and Its Combinatorial Applications”, IEEE Trans. Information Theory, Vol. 64(5), pp. 3610–3615, 2018.
URL https://doi.org/10.1109/TIT.2018.2806486

We show that the inequality $H(A \mid B, X) + H(A \mid B, Y) \leq H(A \mid B)$ for jointly distributed random variables A, B, X, Y, which does not hold in general case, holds under some natural condition on the support of the probability distribution of A, B, X, Y. This result generalizes a version of the conditional Ingleton inequality: if for some distribution $I(X : Y \mid A) = H(A \mid X, Y) = 0$, then $I(A : B) \leq I(A : B \mid X) + I(A : B \mid Y) + I(X : Y)$.

We present the following applications of our result. The first one is the following easy-to-formulate theorem on edge colorings of bipartite graphs: assume that the edges of a bipartite graph are colored in K colors so that each two edges sharing a vertex have different colors and for each pair (left vertex x, right vertex y) there is at most one color a such both x and y are incident to edges with color a; assume further that the degree of each left vertex is at least L and the degree of each right vertex is at least R. Then $K \geq LR$.

Participants

- Farid Ablayev
Kazan State University, RU
- Eric Allender
Rutgers University – Piscataway, US
- Josh Alman
MIT – Cambridge, US
- Vikraman Arvind
Institute of Mathematical Sciences – Chennai, IN
- Nikhil Balaji
Universität Ulm, DE
- Markus Bläser
Universität des Saarlandes, DE
- Andrej Bogdanov
The Chinese University of Hong Kong, HK
- Sourav Chakraborty
Indian Statistical Institute – Kolkata, IN
- Stephen A. Fenner
University of South Carolina – Columbia, US
- Michael A. Forbes
University of Illinois – Urbana-Champaign, US
- Lance Fortnow
Georgia Institute of Technology – Atlanta, US
- Anna Gál
University of Texas – Austin, US
- William Gasarch
University of Maryland – College Park, US
- Sevag Gharibian
Universität Paderborn, DE
- Frederic Green
Clark University – Worcester, US
- Rohit Gurjar
Indian Institute of Technology – Mumbai, IN
- William Hoza
University of Texas – Austin, US
- Christian Ikenmeyer
MPI für Informatik – Saarbrücken, DE
- Valentine Kabanets
Simon Fraser University – Burnaby, CA
- Neeraj Kayal
Microsoft Research India – Bangalore, IN
- Pascal Koiran
ENS – Lyon, FR
- Antonina Kolokolova
Memorial University of Newfoundland – St. John's, CA
- Arpita Korwar
University Paris-Diderot, FR
- Michal Koucký
Charles University – Prague, CZ
- Nutan Limaye
Indian Institute of Technology – Mumbai, IN
- Zhenjian Lu
Simon Fraser University – Burnaby, CA
- Vladimir Lysikov
Universität des Saarlandes, DE
- Meena Mahajan
Institute of Mathematical Sciences – Chennai, IN
- David A. Mix Barrington
University of Massachusetts – Amherst, US
- Anurag Pandey
Universität des Saarlandes, DE
- Natacha Portier
ENS – Lyon, FR
- Noga Ron-Zewi
University of Haifa, IL
- Chandan Saha
Indian Institute of Science – Bangalore, IN
- Ramprasad Saptharishi
TIFR Mumbai, IN
- Nitin Saxena
Indian Institute of Technology Kanpur, IN
- Uwe Schöning
Universität Ulm, DE
- Ronen Shaltiel
University of Haifa, IL
- Amnon Ta-Shma
Tel Aviv University, IL
- Thomas Thierauf
Hochschule Aalen, DE
- Jacobo Torán
Universität Ulm, DE
- Christopher Umans
Caltech – Pasadena, US
- Nikolay K. Vereshchagin
NRU Higher School of Economics – Moscow, RU

Report from Dagstuhl Seminar 18401

Automating Data Science

Edited by

Tijl De Bie[1], Luc De Raedt[2], Holger H. Hoos[3], and Padhraic Smyth[4]

1 **Ghent University, BE,** `tijl.debie@ugent.be`
2 **KU Leuven, BE,** `luc.deraedt@cs.kuleuven.be`
3 **Leiden University, NL,** `hh@liacs.nl`
4 **University of California – Irvine, US,** `smyth@ics.uci.edu`

Abstract

Data science is concerned with the extraction of knowledge and insight, and ultimately societal or economic value, from data. It complements traditional statistics in that its object is data as it presents itself in the wild (often complex and heterogeneous, noisy, loosely structured, biased, etc.), rather than well-structured data sampled in carefully designed studies. It also has a strong computer science focus, and is related to popular areas such as big data, machine learning, data mining and knowledge discovery.

Data science is becoming increasingly important with the abundance of big data, while the number of skilled data scientists is lagging. This has raised the question as to whether it is possible to automate data science in several contexts. First, from an artificial intelligence perspective, it is interesting to investigate whether (data) science (or portions of it) can be automated, as it is an activity currently requiring high levels of human expertise. Second, the field of machine learning has a long-standing interest in applying machine learning at the meta-level, in order to obtain better machine learning algorithms, yielding recent successes in automated parameter tuning, algorithm configuration and algorithm selection. Third, there is an interest in automating not only the model building process itself (cf. the Automated Statistician) but also in automating the preprocessing steps (data wrangling).

This Dagstuhl seminar brought together researchers from all areas concerned with data science in order to study whether, to what extent, and how data science can be automated.

Seminar September 30–October 5, 2018 – http://www.dagstuhl.de/18401
2012 ACM Subject Classification Computing methodologies → Artificial intelligence, Computing methodologies → Machine learning
Keywords and phrases artificial intelligence, automated machine learning, automated scientific discovery, data science, inductive programming
Digital Object Identifier 10.4230/DagRep.8.9.154
Edited in cooperation with Jefrey Lijffijt

Automating Data Science, *Dagstuhl Reports*, Vol. 8, Issue 09, pp. 154–181
Editors: Tijl De Bie, Luc De Raedt, Holger H. Hoos, and Padhraic Smyth
Dagstuhl Reports
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Executive Summary

Tijl De Bie (Ghent University, BE)
Luc De Raedt (KU Leuven, BE)
Holger H. Hoos (Leiden University, NL)
Padhraic Smyth (University of California – Irvine, US)

Introduction

Data science is concerned with the extraction of knowledge and insight, and ultimately societal or economic value, from data. It complements traditional statistics in that its object is data as it presents itself *in the wild* (often complex and heterogeneous, noisy, loosely structured, biased, etc.), rather than data well-structured data sampled in carefully designed studies.

Such 'Big Data' is increasingly abundant, while the number of skilled data scientists is lagging. This has raised the question as to whether it is possible to automate data science in several contexts. First, from an artificial intelligence perspective, it is related to the issue of "robot scientists", which are concerned with the automation of scientific processes and which have so far largely focused on the life sciences. It is interesting to investigate whether principles of robot scientists can be applied to data science.

Second, there exist many results in the machine learning community, which has since the early 1980s been applying machine learning at a meta-level, in order to learn which machine learning algorithms, variants and (hyper-)parameter settings should be used on which types of data sets.

In recent years, there have been breakthroughs in this domain, and there now exist effective systems (such as Auto-WEKA and auto-sklearn) that automatically select machine learning methods and configure their hyperparameters in order to maximize the predictive performance on particular datasets.

Third, there are projects such as the Automated Statistician that want to fully automate the process of statistical modeling. Such systems could dramatically simplify *scientific data modeling* tasks, empowering scientists from data-rich scientific disciplines such as bioinformatics, climate data analysis, computational social science, and so on. To ensure success, important challenges not only from a purely modelling perspective, but also in terms of interpretability and the human-computer interface, need to be tackled. For example, the input to the Automated Statistician is a dataset, and the system produces not only a complex statistical model by means of a search process, but also explains it in natural language.

Fourth, there is an interest in not only automating the model building step in data science, but also various steps that precede it. It is well known in data science that 80% of the effort goes into preprocessing the data, putting it in the right format, and selecting the right features, whereas the model-building step typically only takes 20% of the effort. This has motivated researchers to focus on automated techniques for data wrangling, which is precisely concerned with transforming the given dataset into a format that can be handled by the data analysis component. Here, there are strong connections with inductive programming techniques.

Fifth, as it is often easier for non-expert users to interpret and understand visualisations of data rather than statistical models, work on automatic visualisation of data sets is also very relevant to this Dagstuhl seminar.

Finally, an interesting and challenging research question is whether it is possible to develop an integrated solution that tackles all these issues (as is the topic of the ERC AdG SYNTH).

Overview of the seminar

Structure of the seminar

The seminar was structured as follows. The mornings were generally dedicated to presentations (short tutorials on day one), whereas the afternoons were generally dedicated to discussions such as plenary discussions, smaller-group breakout sessions, and flex time that was kept open prior to the seminar. The flex time ended up being dedicated to a mix of presentations and breakout sessions.

Challenges in automating data science

On day one, a range of challenges for research on automating data science were identified, which can be clustered around the following six themes:

1. **Automating Machine Learning (AutoML)**
   Main challenges: computational efficiency; ensuring generalization also for small data; make AutoML faster and more data-efficient using meta-learning; extending ideas from AutoML to exploratory analysis / unsupervised learning.
2. **Exploratory data analysis and visualization**
   Main challenges: the fact that there is there is no single or clearly defined objective; help the user make progress towards an ill-defined goal; (subjective) interestingness of an analysis, a pattern, or a visualization; integrate machine learning and interaction in exploration; exploration of data types beyond simply tabular; veracity of visualizations; how to quantify progress and measure success; the need for benchmarks.
3. **Data wrangling**
   Main challenges: extend the scope of AutoML to include data wrangling tasks; user interfaces to provide intuitive input in data wrangling tasks; how to quantify progress and measure success; the need for benchmarks.
4. **Automation and human-centric data science (explainability, privacy, fairness, trust, interaction)**
   Main challenges: build-in privacy and fairness constraints in automatic data science systems; the dangers of ignorant usage of automated data science systems; different levels of expertise benefit from different degrees of automation; optimizing the performance of the combined human/machine ‘team’; determine when and where the human must be involved; definition or criteria for explainability; risk that automation will reduce explainability and transparency; explainability to whom – a data scientist or layperson?
5. **Facilitating data science by novel querying and programming paradigms**
   Main challenges: interactive data models to help users gain intuitive understanding; declarative approaches for data analysis, querying, and visualization; a query language for automated data science.
6. **Evaluation**
   Main challenges: robust objective measures for data science processes beyond predictive modelling; subjective measures: measures that depend on the user background and goals; evaluation of the entire data science pipeline versus individual steps; reproducibility in the presence of user interactions.

Topics discussed in depth

These identified challenges were then used to determine the program of the rest of the seminar. Talks were held on partial solutions to a range of these challenges. In addition, breakout discussions were held on the following topics:

1. The relation between data-driven techniques and knowledge-based reasoning.
2. Data wrangling.
3. Beyond the black-box: explainability.
4. Automation of exploratory / unsupervised data science tasks, and visualization.
5. Automating data science for human users.

Along with abstracts of the talks, detailed discussions of the main ideas and conclusions of each of these breakout sessions are included in this Dagstuhl report.

Discussion and outlook

Automating data science is an area of research that is understudied as such. AutoML, as a subarea of automating data science, is arguably the first subarea where some remarkable successes have been achieved. This seminar identified the main challenges for the field in translating these successes into advances in other subareas of automating data science, most notably in automating exploratory data analysis, data wrangling and related tasks, integrating data and knowledge-driven approaches, and ultimately the data science process as a whole, from data gathering to the creation of insights and value.

Further developing automated data science raises several challenges. A first challenge concerns the evaluation of automated data science methods. Indeed, the possibility to automate is preconditioned on the availability of criteria to optimize. A second key one is how to ensure that automated data science systems remain Human-Centric, viewing humans as useful allies and ultimate beneficiaries. This can be achieved by designing effective user-interaction techniques, by ensuring explainability, and by ensuring privacy is respected and individuals are treated fairly. These are basic requirements for ensuring justified trust in automated data science systems, and thus key drivers to success.

2 Table of Contents

Executive Summary

Overview of Talks

Working groups

3 Overview of Talks

3.1 Automated Machine Learning from Spatio-temporal Data

Mitra Baratchi (Leiden University, NL)

Main reference Mitra Baratchi, Geert Heijenk, Maarten van Steen: "Spaceprint: a Mobility-based Fingerprinting Scheme for Public Spaces", CoRR, Vol. abs/1703.09962, 2017.
URL https://arxiv.org/abs/1703.09962

Spatio-temporal mobility datasets are generated abundantly as a result of prevalent use of location-aware technologies. Incorporating unprecedented information about moving entities such as people, animals, and vehicles, automating the process of learning from such data opens the door towards many applications in ecology, transportation, and urban planning. However, due to having a non-propositional representation, automated machine learning from raw mobility data is still an open challenge. Current machine-learning-based approaches using such data still rely on an extensive manual pre-processing phase. In this talk, I presented two examples of automated pre-processing tasks based on mobility data in the context of space classification and map segmentation. Both these examples can achieve automation through defining an unsupervised learning problem on the original representation of data.

3.2 Towards Automated Clustering

Hendrik Blockeel (KU Leuven, BE)

Joint work of Hendrik Blockeel, Toon Van Craenendonck, Wannes Meert, Sebastijan Dumancic, Elia Van Wolputte
URL https://dtai.cs.kuleuven.be/software/cobras/

This talk provides an overview of the PhD research of Toon Van Craenendonck on semi-automated, interactive clustering. The main conclusions of this research are as follows. The choice of the clustering algorithm strongly affects the results of clustering. Choosing the most suitable algorithm cannot be done with internal quality measures, but it can be done using a small number of must-link and cannot-link constraints. A novel algorithm called COBRAS is proposed that makes use of an intermediate layer between clusters and instances, called super-instances, and that automatically determines the appropriate granularity of the super-instances. COBRAS is the first clustering algorithm that is truly interactive in the sense that it combines three desirable properties: it is anytime, query-efficient, and time-efficient.

References

1 T Van Craenendonck, H Blockeel (2015). Using internal validity measures to compare clustering algorithms. Benelearn 2015 Poster presentations (online), 1-8
2 T Van Craenendonck, H Blockeel (2017). Constraint-based clustering selection. Machine Learning 106(9-10): 1497-1521.
3 T Van Craenendonck, S Dumancic, H Blockeel (2017). COBRA: A Fast and Simple Method for Active Clustering with Pairwise Constraints. IJCAI 2017: 2871-2877

4 T Van Craenendonck, S Dumančić, E Van Wolputte, H Blockeel (2018). COBRAS: Fast, Iterative, Active Clustering with Pairwise Constraints. In Proc. of 17th International Symposium on Intelligent Data Analysis, 2018. Springer, to appear. Preprint arXiv:1803.11060

5 T Van Craenendonck, W Meert, S Dumancic, H Blockeel (2018). COBRAS-TS: A new approach to Semi-Supervised Clustering of Time Series. In Proc. of 20th International Conference on Discovery Science. Springer, to appear. Preprint arXiv:1805.00779

3.3 AutoDiscovery : Intelligent Automated Exploratory Data Analysis for Biomedical Research

Ray G. Butler (Butler Scientifics – Barcelona, ES)

Joint work of Ray G. Butler, Joan Guàrdia-Olmos, Javier Hernández-Losa

According to NIH and SciMago Journal & Country Rank estimations, there are more than 400,000 principal investigators worldwide actively running biomedical research projects in the form of clinical studies, collaborations with pharma companies and basic biological research, among others.

The datasets being produced through these projects are distinguished primarily by their complexity in terms of multidimensionality and sample stratification. John W. Tukey's exploratory data analysis (EDA) techniques are rising in response to this particular scenario. However, both open-source and commercial EDA software packages typically require a broad range of data science skills and knowledge including data integration and visualization, software programming and statistical methodologies which makes it difficult for principal investigators to become actively involved in the exploratory phase.

AutoDiscovery is an intelligent automated exploratory data analysis software that helps biomedical principal investigators integrating and exploring their complex datasets to unveil associations with high statistical significance and clinical relevance hidden in the data files of scientific experiments and clinical trials.

3.4 Elements of an Automated Data Scientist

Luc De Raedt (KU Leuven, BE)

Joint work of Luc De Raedt, Hendrik Blockeel, Samuel Kolb, Stefano Teso, Gust Verbruggen
Main reference Luc De Raedt, Hendrik Blockeel, Samuel Kolb, Stefano Teso, Gust Verbruggen: "Elements of an Automatic Data Scientist", in Proc. of the Advances in Intelligent Data Analysis XVII – 17th International Symposium, IDA 2018, 's-Hertogenbosch, The Netherlands, October 24-26, 2018, Proceedings, Lecture Notes in Computer Science, Vol. 11191, pp. 3–14, Springer, 2018.
URL https://doi.org/10.1007/978-3-030-01768-2_1

We provide a simple but non-trivial setting for automating data science. Given are a set of worksheets in a spreadsheet and the goal is to automatically complete some values. We also outline elements of the SYNTH framework that tackles this task: SYNTH-A-SIZER, an automated data wrangling system for automatically transforming the problem into attribute-value format; TACLE, an inductive constraint learning system for inducing formula's in spreadsheets; MERCS, a versatile predictive learning system; as well as the autocompletion component that integrates these systems.

3.5 Towards a Measurement Theory for Data Science

Peter Flach (University of Bristol, GB)

Performance evaluation is of clear importance in machine learning and data science, and arguably even more so for *automated* data science. Our understanding of performance evaluation measures for machine-learned classifiers has improved considerably over the last twenty years. In this short talk I highlighted a range of areas where understanding is still lagging behind our algorithmic advances, sometimes leading to ill-advised practices in classifier evaluation. I argued that in order to make further progress we need to develop a proper *measurement theory* for data science. I gave some examples what such a measurement theory might look like and what kinds of new results it would entail. In future work I will explore the idea that key properties such as classification ability and data set difficulty are unlikely to be directly observable, taking inspiration from the kind of latent-variable models developed in psychometrics. I will also explore the value of causal explanations of observed performance of machine learning models and algorithms.

3.6 MagicWrangler Demo: Tool and Data

Jose Hernandez-Orallo (Technical University of Valencia, ES)

Joint work of Jose Hernandez-Orallo, Lidia Contreras-Ochando, Cèsar Ferri, Susumu Katayama, Fernando Martínez-Plumed, M. José Ramírez-Quintana
URL http://dmip.webs.upv.es/datawrangling/

MagicWrangler is a data wrangling tool that makes domain identification and extracts patterns using MagicHaskeller. We made a presentation of how the domain is identified and used to reduce the search space for MagicHaskeller. We presented a new data repository including 123 data wrangling datasets.

References

1 Contreras-Ochando, L.; Martínez-Plumed, F.; Ferri, C.; Hernández-Orallo, J.; and Ramírez-Quintana, M. J. “General-purpose inductive programming for data wrangling automation”, AI4DataSci @ NIPS, 2016.

2 Contreras-Ochando, L.; Ferri, C.; Hernández-Orallo, J.; Martínez-Plumed, F.; Ramírez-Quintana, M. J.; and Katayama, S. “Domain specific induction for data wrangling automation (system demonstration)”, AutoML @ ICML 2017.

3 Contreras-Ochando, L. “Domain specific induction for data wrangling automation”, in Approaches and Applications of Inductive Programming (Dagstuhl Seminar 17382), Schmid, U. ; Muggleton, S. H. ; Singh, R. (eds.) Dagstuhl Reports, Volume 7, Issue 9. 2018, DOI: 10.4230/DagRep.7.9.86

4 Contreras-Ochando, L.; Ferri, C.; Hernández-Orallo, J.; Martínez-Plumed, F.; Ramírez-Quintana, M. J.; and Katayama, S. “General-purpose declarative inductive programming with domain specific background knowledge for data wrangling automation”, https://arxiv.org/abs/1809.10054

3.7 Mapping the Skills in Data Science with Those in AI/ML

Jose Hernandez-Orallo (Technical University of Valencia, ES) and Lidia Contreras-Ochando

Joint work of Jose Hernandez-Orallo, Cèsar Ferri, Flach, P., Kull, M., Lachiche, N., Fernando Martínez-Plumed, M. José Ramírez-Quintana

Many skills, knowledge, abilities and competences have been identified as necessary for data scientists, including both technical and non-technical skills. Some standards, such as CRISP-DM (and extensions) recognise processes, but not competences or skills. In this talk, under the more general context of automation in the workplace, we explored how we can identify and map the skills in data science with the capabilities of AI and ML, in order to know when tasks can be effectively semi-automated in data science, according to the skills they need and the foreseeable progress in AI/ML.

We touched upon issues such as the value of semi-automation in front of whole automation, the problems of task 'atomising' instead of task automating, the issues of generality and autonomy in automation [3], the new needs for further supervision whenever automation takes place, and the overall view of automating data science as an intrinsic part of it, which actually makes data science evolve and require new tasks, skills and abilities.

We analysed what data science needs by first looking at the traditional data mining process (e.g., CRISP-DM), with a view of optimisation by considering a precise data mining goal, with then moving to more open data science trajectories [2], where data scientists require many non-technical skills and attitudes, and especially knowledge about the domain. We saw that more efforts for competence frameworks are needed, as what we have it is still too preliminary to characterise what data science requires in terms of skills.

On the other hand, we covered what AI/ML is providing, to see whether it is ready (now or in the near future) to cover partial automation in data science. We saw that apart from many other techniques in computer science, AI/ML provide solvers and learners, using Geffner's terminology [4]. If we restrict to ML only, there are some recent proposal of rubrics to see whether a task is automatable according to them, such as Brynjolfsson and Mitchell's rubric [1], which could be used for data science too. If we focus on the automatic handling of domain knowledge, there has been significant progress (e.g., NELL, Watson, etc.), but this is still poorly integrated with some other techniques.

Overall, we see that performing a well-informed mapping between what data science requires and what AI/ML can provide in the years to come is extremely challenging, perhaps more challenging than other areas where automation is playing an increasing role (e.g., transportation). This may be caused by the exploratory character of data science, which in the end is associated with the scientific methodology (data scientists are a kind of scientists). Nevertheless, we claimed that performing an analysis in terms of abilities or skills [5] will be more powerful and predictive in terms of degrees and opportunities for automation than just performing the analysis in terms of tasks.

References

1 Brynjolfsson, E. and Mitchell, T. "What can machine learning do? Workforce implications". Science, 358(6370):1530–1534, 2017.

2 Contreras, L., Ferri, C., Flach, P., Kull, M., Hernandez-Orallo, J. Lachiche, N., Martínez-Plumed, F. Ramírez-Quintana, M. J., "CRISP-DM Twenty Years Later: From Data Mining Processes to Data Science Trajectories" in preparation.

3 Fernández-Macías, E., Gómez, E., Hernández-Orallo, J., Loe, B.S., Martens, B., Martínez-Plumed, F. and Tolan, S., 2018. A multidisciplinary task-based perspective for evaluating the impact of AI autonomy and generality on the future of work. arXiv preprint arXiv:1807.02416, AEGAP@IJCAI 2018.

4 Geffner, H. "Model-free, Model-based, and General Intelligence", ht-tps://arxiv.org/abs/1806.02308, 2018.

5 Hernández-Orallo, J. "Evaluation in artificial intelligence: from task-oriented to ability-oriented measurement", Artificial Intelligence Review 48 (3), 397-447, 2017.

3.8 A Taxomomy of Methods for Explainable Machine Learning Models

Tobias Jacobs (NEC Laboratories Europe – Heidelberg, DE)

In a short talk at Dagstuhl I have presented a taxonomy to categorize methods that contribute to explainable (including interpretable) machine learning models. A first fundamental dimension of the taxonomy is the purpose for which the method is applicable. Potential purposes range from technical and scientific benefits (e.g. model debugging, sanity checking, generating new insights) to societal requirements (e.g. validation of fairness, granting of of legal rights). The second dimension distinguishes between explainability as a model requirement or constraint during model construction on the one hand, and methods that help to explain existing complex models on the other. The latter kind of methods can be further categorized as methods which explain black-box models in terms of generic properties of the black box, or as methods to open the box and analyze what is happening inside. The final dimension distinguishes between methods to explain a model as a whole (also known as interpretability or global explainability) and methods to explain specific results of the model (also known as local explainability, or explainability in the narrow sense).

3.9 Counterfactual Prediction with Instrumental Variables and Deep Learning

Kevin Leyton-Brown (University of British Columbia – Vancouver, CA)

Joint work of Jason Hartford, Greg Lewis, Kevin Leyton-Brown, Matt Taddy
Main reference Jason S. Hartford, Greg Lewis, Kevin Leyton-Brown, Matt Taddy: "Deep IV: A Flexible Approach for Counterfactual Prediction", in Proc. of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017, Proceedings of Machine Learning Research, Vol. 70, pp. 1414–1423, PMLR, 2017.
URL http://proceedings.mlr.press/v70/hartford17a.html

Counterfactual prediction requires understanding causal relationships between so-called treatment and outcome variables. This paper provides a recipe for augmenting deep learning methods to accurately characterize such relationships in the presence of instrument variables (IVs) – sources of treatment randomization that are conditionally independent from the outcomes. Our IV specification resolves into two prediction tasks that can be solved with deep neural nets: a first-stage network for treatment prediction and a second-stage network whose loss function involves integration over the conditional treatment distribution. This Deep

IV framework allows us to take advantage of off-the-shelf supervised learning techniques to estimate causal effects by adapting the loss function. Experiments show that it outperforms existing machine learning approaches.

3.10 Subjective Interestingness in Data Mining

Jefrey Lijffijt (Ghent University, BE)

Joint work of Jefrey Lijffijt, Tijl De Bie
URL http://users.ugent.be/ jlijffij/presentations/DAGSTUHL2018.pdf

I present a brief introduction to the topic of subjective interestingness, particularly an information-theoretic view that enables ranking of any type of patterns that we may want to extract from data. After this introduction, we review two 'instances' of this approach, for relational patterns (a generalisation of itemsets/tiles), as well as automatically finding informative views of data by seeing visualisations as patterns. Finally, I conclude with a question regarding to the topic of the workshop: are there fundamental or important differences between the topics of explainability and interpretability of machine learning models versus deriving insights from data.

Most of the talk covers material from the tutorial that we recently presented on this topic: http://www.interesting-patterns.net/forsied/tutorial/

References

1 Tias Guns, Achille Aknin, Jefrey Lijffijt, Tijl De Bie. "Direct mining of subjectively interesting relational patterns". In Proceedings of the IEEE International Conference on Data Mining (ICDM), pp 913 – 918, 2016.

2 Bo Kang, Jefrey Lijffijt, Raúl Santos-Rodríguez, Tijl De Bie. "SICA: Subjectively Interesting Component Analysis". Data Mining and Knowledge Discovery 32(4): 949-987, 2018.

3 Bo Kang, Kai Puolamäki, Jefrey Lijffijt, Tijl De Bie. "A tool for subjective and interactive visual data exploration". In Proceedings of the European Conference of Machine Learning and Principles and Practices of Knowledge Discovery in Databases (ECML-PKDD) – Part III, pp. 3 – 7, 2016.

4 Jefrey Lijffijt, Eirini Spyropoulou, Bo Kang, Tijl De Bie. "P-N-RMiner: A generic framework for mining interesting structured relational patterns". International Journal of Data Science and Analytics, 1(1): 61-76, 2016.

5 Kai Puolamäki, Bo Kang, Jefrey Lijffijt, Tijl De Bie. "Interactive visual data exploration with subjective feedback". In Proceedings of the European Conference of Machine Learning and Principles and Practices of Knowledge Discovery in Databases (ECML-PKDD) – Part II, pp. 214 – 229, 2016.

3.11 AutoML Challenges 2015-2018: Review and Call for Action

Zhengying Liu (University of Paris Sud – Orsay, FR)

Joint work of Zhengying Liu, Isabelle Guyon, Hugo Jair Escalante, Michèle Sebag, Wei-Wei Tu, Andre Elisseeff, and numerous, other collaborators
Main reference Isabelle Guyon, Kristin Bennett, Gavin C. Cawley, Hugo Jair Escalante, Sergio Escalera, Tin Kam Ho, Núria Macià, Bisakha Ray, Mehreen Saeed, Alexander R. Statnikov, Evelyne Viegas: "Design of the 2015 ChaLearn AutoML challenge", in Proc. of the 2015 International Joint Conference on Neural Networks, IJCNN 2015, Killarney, Ireland, July 12-17, 2015, pp. 1–8, IEEE, 2015.
URL https://doi.org/10.1109/IJCNN.2015.7280767

We introduce a series of data challenges in the research field of AutoML, organized by ChaLearn and many other organizations, with the support of numerous collaborators. These challenges are:

- AutoML challenge (2015-2016), collocated with NIPS, ICML, IJCNN;
- AutoML2 (2017-2018), collocated with PAKDD18;
- AutoML3: AutoML for Lifelong Machine Learning (on-going), collocated with NIPS18;
- AutoDL challenge (coming soon), more details to be announced.

References

1 Isabelle Guyon, Kristin Bennett, Gavin Cawley, Hugo Jair Escalante, Sergio Escalera, Tin Kam Ho, Núria Macia, Bisakha Ray, Mehreen Saeed, Alexander Statnikov, et al. Design of the 2015 ChaLearn AutoML challenge. In *Neural Networks (IJCNN), 2015 International Joint Conference on*, pages 1–8. IEEE, 2015.

2 Isabelle Guyon, Lisheng Sun-Hosoya, et al. Analysis of the AutoML challenge series 2015-2018. In https://www.automl.org/book/

3.12 Bilevel Programming for Hyperparameter Optimization and Meta-Learning

Paolo Frasconi (University of Florence, IT)

Joint work of Luca Franceschi, Massimiliano Pontil, Paolo Frasconi, Riccardo Grazzi, Saverio Salzo
Main reference Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, Massimiliano Pontil: "Bilevel Programming for Hyperparameter Optimization and Meta-Learning". In Proc. of the 35th International Conference on Machine Learning ICML 2018, Stockholm, Sweden, July 10-15, 2018, pp. 1563–1572.
URL http://proceedings.mlr.press/v80/franceschi18a.html

We introduce a framework based on bilevel programming that unifies gradient-based hyperparameter optimization and meta-learning. We show that an approximate version of the bilevel problem can be solved by taking into explicit account the optimization dynamics for the inner objective. Depending on the specific setting, the outer variables take either the meaning of hyperparameters in a supervised learning problem or parameters of a meta-learner. We provide sufficient conditions under which solutions of the approximate problem converge to those of the exact problem. We instantiate our approach for meta-learning in the case of deep learning where representation layers are treated as hyperparameters shared across a set of training episodes. In experiments, we confirm our theoretical findings, present encouraging results for few-shot learning and contrast the bilevel approach against classical approaches for learning-to-learn.

3.13 Pyconstruct: A Library for Declarative, Constructive Machine Learning

Andrea Passerini (University of Trento, IT)

Joint work of Paolo Dragone, Stefano Teso, Andrea Passerini
Main reference Paolo Dragone, Stefano Teso, Andrea Passerini: "Pyconstruct: Constraint Programming Meets Structured Prediction", in Proc. of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden., pp. 5823–5825, ijcai.org, 2018.
URL https://doi.org/10.24963/ijcai.2018/850

Constructive learning is the task of learning to synthesize structured objects from data. Examples range from classical sequence labeling to layout synthesis and drug design. Learning in these scenarios involves repeatedly synthesizing candidates subject to feasibility constraints and adapting the model based on the observed loss. Many synthesis problems of interest are non-standard: they involve discrete and continuous variables as well as arbitrary constraints among them. In these cases, widespread formalisms (like linear programming) can not be applied, and the developer is left with writing her own ad-hoc solver. This can be very time consuming and error prone. I will describe Pyconstruct [1], a Python library tailored for solving real-world constructive problems with minimal effort. The library leverages max-margin approaches to decouple learning from synthesis and constraint programming as a generic framework for synthesis. Pyconstruct enables easy prototyping of working solutions, allowing developers to write complex synthesis problems in a declarative fashion in few lines of code. The library is available at: https://goo.gl/U1PaKF

References

1 Paolo Dragone, Stefano Teso and Andrea Passerini. Pyconstruct: Constraint Programming Meets Structured Prediction. In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18, pages 5823–5825, 7 2018.

3.14 Pairwise Meta Rules, Full Model Selection, and Some Speculative Ideas

Bernhard Pfahringer (University of Waikato, NZ)

Joint work of Quan Sun, Mike Mayo, Bernhard Pfahringer
Main reference Sun, Q. (2014). Meta-Learning and the Full Model Selection Problem (Thesis, Doctor of Philosophy (PhD)). University of Waikato, Hamilton, New Zealand. Retrieved from https://hdl.handle.net/10289/8520
URL https://hdl.handle.net/10289/8520

In this talk I present pairwise meta rules for meta-learning, as well as a more scalable hierarchical version of them. They show good potential, especially when used with nearest neighbour, or random ranking forests. I also quickly cover an early attempt at an evolutionary system that jointly optimises combinations of up to four preprocessing methods and one learning algorithm. System Fantail combines ideas from genetic algorithms with particle swarm optimisation. Finally I speculate about approaches beyond black box optimisation.

3.15 Interactive / Visual Data Exploration Tutorial

Kai Puolamäki (University of Helsinki, FI) and Remco Chang (Tufts University – Medford, US)

We give an overview of interactive and visual data exploration. First, we discuss the scope of the talk, which include the analytic problems that benefit from both automation and human feedback, and in which the interaction happens at time scales of c. 1-10 seconds. We argue that practitioners would benefit from studying visualization techniques developed in vis community, such as graph techniques and techniques for multivariate data. Finally, we give an high level overview of dimensionality reduction and how to incorporate interaction there, plus a brief tutorial of how to get started on interactive and visual data exploration using R and Shiny library.

3.16 Tell Me Something I Don't Already Know: Tools for Human-guided Data Analysis

Kai Puolamäki (University of Helsinki, FI)

The outcome of the explorative data analysis (EDA) phase is vital for successful data analysis. EDA is more effective when the user interacts with the system used to carry out the exploration. A good EDA system has three requirements: (i) it must be able to model the information already known by the user and the information learned by the user, (ii) the user must be able to formulate the objectives, and (iii) the system must be able to show the user views that are maximally informative about desired features data that are not already know for the user. Furthermore, the system should be fast if used in interactive system. We present the Human Guided Data Exploration framework which satisfies these requirements and generalises previous research. This framework allows the user to incorporate existing knowledge into the exploration process, focus on exploring a subset of the data, and compare different complex hypotheses concerning relations in the data. The framework utilises a computationally efficient constrained randomisation scheme. To showcase the framework, we developed a free open-source tool, using which the empirical evaluation on real-world datasets was carried out. Our evaluation shows that the ability to focus on particular subsets and being able to compare hypotheses are important additions to the interactive iterative data mining process.

In this talk we present some tools for human-guided data analysis that utilise maximum entropy and/or constrained randomisation methods, such as sideR, available at http://www.iki.fi/kaip/sider.html, and tiler available at https://github.com/aheneliu/tiler

References

1 Puolamäki, Oikarinen, Atli, Henelius. Human-guided data exploration using randomization. arXiv:1805.07725 [stat.ML]
2 Henelius, Oikarinen, Puolamäki. Tiler: Software for Human-Guided Data Exploration. In Proc ECML-PKDD 2018, to appear. https://youtu.be/fqKLjMwJHnk

3 Puolamäki, Oikarinen, Kang, Lijffijt, De Bie. Interactive Visual Data Exploration with Subjective Feedback: An Information-Theoretic Approach. In Proc ICDE 2018, to appear. arXiv:1710.08167 [stat.ML]

3.17 Explaining Learned Models – Towards Relating Verbal Explanations to Visual Domains for Interactive Learning with Mutual Explanations

Ute Schmid (Universität Bamberg, DE)

Main reference Johannes Rabold, Michael Siebers, Ute Schmid: "Explaining Black-Box Classifiers with ILP – Empowering LIME with Aleph to Approximate Non-linear Decisions with Relational Rules", in Proc. of the Inductive Logic Programming – 28th International Conference, ILP 2018, Ferrara, Italy, September 2-4, 2018, Proceedings, Lecture Notes in Computer Science, Vol. 11105, pp. 105–117, Springer, 2018.
URL https://doi.org/10.1007/978-3-319-99960-9_7

With the ever growing interest in machine learning in application domains such as autonomous driving, medical diagnosis, connected industry, or education, it has been recognized that machine learned models need to be transparent and comprehensible. For instance, a medical expert has to understand why a machine classifies some health state as critical given a computer tomographic image before s/he decides on the diagnosis. This is especially the case when the expert opinion deviates from the classification. In this case, the expert might (maybe wrongly) suspect a false alarm, mistrust the system, and consequently follow his/her own opinion. In the context of the collaborative project 'Transparent Medical Expert Companion' we are developing an approach to explanation generation for medical image data. Explanation generation is realized by a template based transformation of Prolog rules into natural language text. The Prolog rules has been learned with an inductive programming approach (Aleph, Metagol). Current topics of research in our project are: Combining black-box machine learning, especially Convolutionary Networks with ILP to address the trade-off between predicitive accuracy and interpretability; investigating incremental learning to allow the human experts to correct classifications which results in an adaptation of the classification rules; relating verbal explanations with the original image data, especially for explaining rules involving binary relations or negation.

3.18 Monte Carlo Tree Search for Algorithm Configuration: MOSAIC

Michele Sebag (CNRS, FR)

Joint work of Michele Sebag, Herilalaina Rakotoarison

The sensitivity of machine learning algorithms w.r.t. their hyper-parameters and the difficulty of finding the ML algorithm and its hyper-parameter setting best suited to the dataset at hand has led to the rapidly developing field of automated machine learning (AutoML), at the crossroad of meta-learning and structured optimization. Several international AutoML challenges have been organized since 2015, motivating the development of the Bayesian optimization-based approach Auto-Sklearn (Feurer et al. 15) and the randomized search approach Hyperband (Li et al. 16). In this paper, a new approach, called Monte Carlo Tree Search for Algorithm Configuration (MOSAIC), is presented, fully exploiting the tree structure of the algorithm portfolio-hyperparameter search space, with competitive results on the AutoML challenge 2015.

3.19 Statistical Thinking and Data Science: Observations

Padhraic Smyth (University of California – Irvine, US)

In the context of the seminar topic "automating data science" it is worth visiting potentially relevant ideas from the field of statistics, given the historical experience with the broad canvas of data analysis in that field. There has long been an acknowledgement in statistics of the importance of statistical thinking and statistical strategy in terms of providing a view of data analytic activities at a more general level than the details of specific methods and techniques. From a computer science perspective there is value in understanding how statisticians have addressed the question of how to think about data analysis in a systematic manner. This systematic perspective has been approached from the point of view of the practice of data analysis (see for example position papers by Tukey, Mallows, Cox, Huber, Chatfield, and many others), to the development of general theories and models for the process of data analysis (e.g., see Wilde 1994; Grolemund and Wickham, 2014), to education (e.g., Breiman, 1984 (US Berkeley Tech Report); Horton and colleagues, 2014 onwards), to the development of software systems to provide guidance to data analysts along the path of data analysis (e.g., see Oldford and Peters, 1986; Lubinsky and Pregibon, 1988). In this talk we briefly discussed these threads of work and their potential relevance to current endeavours in data science. In particular, one important message from this prior work is the notion that the human analyst is central to the process of data analysis and, as a consequence, tools that support the human analyst (i.e., semi-automation) are more likely to be successful than tools that seek to fully automate data analysis without a human in the loop.

3.20 Explainable Interactive Learning

Stefano Teso (KU Leuven, BE)

Joint work of Stefano Teso, Kristian Kersting
Main reference Stefano Teso, Kristian Kersting: ""Why Should I Trust Interactive Learners?" Explaining Interactive Queries of Classifiers to Users", CoRR, Vol. abs/1805.08578, 2018.
URL https://arxiv.org/abs/1805.08578

Although interactive learning puts the user into the loop, the learner remains mostly a black box for the user. Understanding the reasons behind queries and predictions is important when assessing how the learner works and, in turn, justifiably establishing or revoking trust. This talk covers some recent work on integrating active learning with explainable machine learning, where the queries to the user are augmented with predictions and explanations thereof, and the user provides both labels and explanation corrections for improved directability and control.

3.21 Automatic Configuration of Stream Clustering Algorithms?

Heike Trautmann (Universität Münster, DE)

Joint work of Heike Trautmann, Matthias Carnein

Analysing data streams has received considerable attention over the past decades due to the widespread usage of sensors, social media and other streaming data sources. A core research area in this field is stream clustering which aims to recognize patterns in an unordered, infinite and evolving stream of observations. Clustering can be a crucial support in decision making, since it aims for an optimized aggregated representation of a continuous data stream over time and allows to identify patterns in large and high-dimensional data. A multitude of algorithms and approaches has been developed that are able to find and maintain clusters over time in the challenging streaming scenario.

However, benchmarking stream clustering algorithms is a nontrivial task although first studies already exist. Besides the well-known problem of choosing an appropriate performance measure in unsupervised learning, the most crucial challenge when applying stream clustering algorithms is the correct choice of parameter settings. Stream clustering algorithms usually have a multitude of interdependent parameters, both for the micro-clustering step as well as for the macro-clustering phase and are highly sensitive to these settings. Automated algorithm configuration techniques would require an appropriate learning phase on the one hand and in our scenario moreover would have to be able to deal with drifts or structural changes of the stream. The talk aims at initiating a fruitful discussion on the topic paving the way to automated algorithm configuration and selection approaches.

References

1 Carnein, M., & Trautmann, H. (2018). Optimizing Data Stream Representation: An Extensive Survey on Stream Clustering Algorithms. Business and Information Systems Engineering (BISE), 2018. (Accepted)

2 Carnein, M., Assenmacher, D., & Trautmann, H. (2017). An Empirical Comparison of Stream Clustering Algorithms. In Proceedings of the ACM International Conference on Computing Frontiers (CF'17), Siena, Italy, 361–365.

3.22 Automatic Bayesian Density Analysis

Isabel Valera (MPI für Intelligente Systeme – Tübingen, DE)

Making sense of a dataset in an automatic and unsupervised fashion is a challenging problem in statistics and AI. Classical approaches for density estimation are usually not flexible enough to deal with the uncertainty inherent to real-world data: they are often restricted to fixed latent interaction models and homogeneous likelihoods; they are sensitive to missing, corrupt and anomalous data; moreover, their expressiveness generally comes at the price of intractable inference. As a result, supervision from statisticians is usually needed to find the right model for the data. However, as domain experts do not necessarily have to be experts in statistics, we propose Automatic Bayesian Density Analysis (ABDA) to make density estimation accessible at large. ABDA automates the selection of adequate likelihood models from arbitrarily rich dictionaries while modeling their interactions via a deep latent structure adaptively learned from data as a sum-product network. ABDA casts uncertainty estimation at these local and global levels into a joint Bayesian inference problem, providing robust and yet tractable inference. Extensive empirical evidence shows that ABDA is a suitable tool for automatic exploratory analysis of heterogeneous tabular data, allowing for missing value estimation, statistical data type and likelihood discovery, anomaly detection and dependency structure mining, on top of providing accurate density estimation.

3.23 Making Smart Data Analytics available for SMEs

Andreas Wierse (SICOS BW GmbH – Stuttgart, DE)

One of the main tasks of SICOS BW is to support small and medium sized enterprises (SMEs) in the uptake of data analytics technology (smart data with emphasis on the generation of value for the company). Since SMEs usually do not have a lot of expertise in data analytics (mostly none at all), we developed so called “potential analysis”-projects, where researchers of KIT (Karlsruhe Institute of Technology) work with the company’s data for a few weeks in order to find interesting patterns but also to let them experience the process that is necessary to perform successful data analytics. If we find interesting patterns in the end the company is often interested to apply analytics in their every day business.

On one side this creates a need for training in order to provide the employees with the necessary knowledge; this is usually a fairly difficult and time consuming process. On the other side it would be very helpful for the SMEs, if the technology they want to use were highly automated. This could ease the burden for the SMEs significantly, since more employees would be able to use the technology and the need for training could be released. In general high usability is crucial for data analytics methods and tools to be successful in the SME context.

The companies that SICOS BW supports come from different sectors, including machinery, production, plant engineering and construction, manufacturing as well as trade an information technology. Application cases are for example predictive maintenance, sales forecast, early identification of production anomalies (chemical industry), energy consumption estimates or text classification.

References

1 Andreas Wierse, Till Riedel. *Praxishandbuch Smart Data Analytics.* De Gruyter Oldenburg, Berlin, Germany, 2017.

3.24 Tutorial on Data Preparation and Cleaning

Christopher Williams (University of Edinburgh, GB)

Joint work of Christopher Williams, Taha Ceritli, James Geddes, Zoubin Ghahramani, Ernesto Jimenez-Ruiz, Ian Horrocks, Alfredo Nazabal, Tomas Petricek, Charles Sutton, and Gerrit Van Den Burg of the Artificial Intelligence for Data Analytics (AIDA) project at the Alan Turing
URL https://www.turing.ac.uk/research/research-projects/artificial-intelligence-data-analytics

A common view is that up to 80% of work on a data mining project is involved in data understanding, cleaning and preparation, yet machine learning has not focused very much on these topics. I will describe the issues around data parsing, obtaining (or inferring) a data dictionary, data integration, entity resolution, addressing format variability, structural variability, identifying and repairing missing data, and anomaly detection and repair.

4 Working groups

The following sections contain summaries of breakout discussion session that were organized during the seminar. The authors for each section led the discussion of one of these breakout sessions. They made an attempt to summarize these discussions in an accurate, comprehensive, and intelligible manner, fairly reflecting all points of view. Although it is not practical to attribute all input to individual participants, note that these summaries are thus the result of input by several, often many of the workshop participants. As such, the summaries also do not always reflect either the author's or the full breakout session group's opinion.

The workshop and the discussion organizers sincerely appreciate the input from all the participants in the breakout sessions and would like to thank everyone for their contributions.

4.1 Breakout session on "Beyond the Black Box – Explainability of Machine Learning Models"

Ray G. Butler (Butler Scientifics – Barcelona, ES)

The breakout session "Beyond the Black Box – Explainability of Machine Learning Models" discussed issues related to being able to explain and interpret the models that machine learning produces. We made a distinction between explainability – the ability of a model to justify decisions/predictions in individual cases – and interpretability – understanding a model as a whole and in general, perhaps through a series of decisions/predictions (a "conversation" with the model). In particular, explainability is relatively easy to achieve and already present for most models (for example each prediction can be traced through a series of neurons that represent certain features in a deep neural network), while interpretability is poorly understood even for simple models. An interpretable model would allow to judge any and all biases that were present in the training data and the model to assess whether the model will work as expected in every case.

An explanation or interpretation of a model is subjective in that one human may be satisfied with/convinced by a particular explanation that another human finds unsatisfactory.

There is literature in the social sciences on what explanations are and how humans perceive them that should be helpful in this context. A caveat to watch out for when providing explanations is that humans are very good at matching patterns and convincing themselves that something makes sense, and an incorrect or misleading explanation that is not supported by the data may sound convincing.

Finally, we discussed the issue of performance in this context. While in many cases only complex and hard-to-interpret models deliver good performance, the use of AutoML methods can help in that it allows to train even relatively simple models with good performance. Explainability can also help to increase performance, as it allows human experts to inspect the model, identify areas where performance is bad and the reasons for that, and remedy those issues.

4.2 Breakout Session on "Automation of data exploration tasks"

Tijl De Bie (Ghent University, BE)

Many important data science tasks are ill-specified: there is some amount of data (increasingly an abundance of it, and often heterogeneous and with complex structure), and the data analyst wishes to gain new insights or make discoveries driven by this data. The main related challenges stem from the fact that such tasks are purposefully ill-defined and open-ended – i.e. there is no clear objective function like in more traditional machine learning tasks where the goal is typically to build a predictive model for a specific target variable.

The key challenge in automating data exploration is thus how to specify and pursue an inherently ill-defined objective, and how to measure progress towards such an objective.

It was mentioned that data exploration is often an intermediate step in a data science pipeline, the end goal of which is well-defined but too complex to tackle without a good understanding of the data. An example was given of investment companies gathering large amounts of data, which is first analyzed in an exploratory manner by data profilers, before it is used for predicting stock price fluctuations – the ultimate goal which is predictive. A more elementary example is the well-known use of of feature selection or dimensionality reduction as a regularization strategy for a subsequent machine learning model.

However, several participants raised examples where data exploration is arguably the end-point. A first example given is data-driven research such as (increasingly) biological research. The end goal of biological research is arguably to understand how life works. This is a scientific discovery problem for which researchers increasingly rely on high-throughput data – i.e. it is increasingly addressed using data science techniques. Of course, it is inevitable to break this task down into smaller subproblems (e.g. "does this particular subsequence in the DNA code for a gene or not?"), but these are merely steps towards the larger goal of using data in order to gain an understanding of life. Other examples that were raised pertain to astronomy research (e.g. the Sloan Digital Sky Survey initiative), as well as to industry (e.g. the placement of a multitude of sensors throughout an industrial plant, which may be used to find anomalies in the process and understand the process better).

Thus, a distinction can be made between tasks where a clear objective exists although it is too complex in practice to pursue without an initial exploratory analysis of the data, and tasks where such an objective function does not exist at all and the goal is simply exploring the data in order to make data-driven discoveries or gain new insights. These different settings may have different needs in practice, and thus have been studied in different communities (e.g. the databases, the statistics and exploratory data analysis, and the KDD communities) using different kinds of techniques. Arguably, the roots are to be found in statistics, and in the seminal work of John Tukey on Exploratory Data Analysis (EDA), who suggested EDA as a way of hypothesis generation, to be subsequently subjected to confirmatory analyses.

A point of discussion was whether the automation of data exploration can be approached in a similar manner as AutoML techniques for supervised learning. To an extent this seems to be the case:

- Where the exploratory analysis is merely an intermediate step, automating this step can at least in principle be driven by the quantifiable objective of the entire data science process (which could be a predictive modeling task). Yet, in practice this is typically infeasible.
- There are also similar issues of model and/or algorithm selection to be done in prototypical data exploration tasks such as outlier detection, clustering, community detection in networks, dimensionality reduction, and density estimation. In those tasks, some notion of model fit (e.g. the likelihood of a model, a clustering cost function, etc.) is an obvious criterion. However, such a notion may not always align with the needs of the human data analyst (e.g. it may explain irrelevant aspects of the data very well, at the expense of the more relevant aspects in the data). Thus, user interaction seems fundamentally inevitable to this process, whereas in AutoML user interaction is arguably needed only due to imperfections in AutoML that at least in principle could be remedied with more research progress.

Some other orthogonal issues were raised in the discussion, and briefly summarized here.

The first remaining issue is the need for methods that can extract insights from and make discoveries in structured data. Many techniques still expect the data to be formatted in a basic data table, often causing significant loss of information.

The second remaining issue is the need to take into account background knowledge, in two ways. The first way is in ensuring that the findings in the data add new value to that prior background knowledge, i.e. they should not be implied by it as the analyst would not gain new insights. An approach like this would ensure the findings are subjectively interesting to the data analyst. The second way is almost the opposite: ensuring that the findings in part corroborate the prior knowledge of the data analyst, in order to create trust in the analysis. Hence, there is a trade-off between trust and information to be made.

The third remaining issue concerns visualization, often a crucial part of data exploration. Quantifying the quality of a visualization is an open challenge. A particular aspect not often studied is the veracity of visualizations, or lack thereof: the risk that a visualization can make the data analyst see patterns that are not there (e.g. t-SNE is known to exhibit spurious cluster structure if the parameters are not tuned right). To conclude, the exploration of data was acknowledged as an important data science component, sometimes as a stepping stone towards a clearly defined goal, and sometimes as an ill-specified but important goal on itself. Such tasks are challenging in practice, and considerable amounts of research are needed to automate it further – but inevitably with the user tightly in the loop.

4.3 Breakout Session on "Data Wrangling"

Jose Hernandez-Orallo (Technical University of Valencia, ES)

We had a breakout session on "Data Wrangling" on Tuesday afternoon. We understood the term as covering all elements in data preparation, such as data integration, cleansing and transformation, e.g., following Chris William's taxonomy, presented in his tutorial. We wanted to make this clear as some definitions of data wrangling are too narrow (e.g., https://en.wikipedia.org/wiki/Data_wrangling).

During the discussion we covered some of the data wrangling topics that were considered most relevant during the previous days. In particular we covered:

- How to instruct the system on what we want to do with data? ('domain specific wrangling'): Given a dataset and a task (for which we are given, in the best cases, a performance metric) can we clean (apply operators to) the dataset to maximise the performance metric? We understand cleaning here as going beyond feature extraction. Or, given a dataset and a target algorithm, can we put the data in the right format for the particular target algorithm? An issue is whether transformation are reversible or traceable (need to be able to come back to the original representation). One question that was raised is whether data wrangling can be done just to increase "data quality" without having a particular task in mind. For instance, domain inference can be done and is useful even if we do not know the task yet. When talking about "domain specific wrangling", we saw the important distinction between the format (more task-independent) and the semantic part (more task-dependent).
- Automatic data integration: this item, which is related to the previous one, considers the integration of different sources of data, in different formats. Can we have a sequence of data wrangling operations that lead to a desirable format? Again, we are here very interested in ensuring provenance (an issue not only but mostly for integration).
- How can we make tools more general where users can add their knowledge instead of being preprogrammed? We mentioned some possibilities, such as the use of inductive logic programming, for which this is natural. As an example, Pyconstruct is a library for declarative, constructive machine learning (Andrea Passerini) where background knowledge is expressed as constraints. One question that arose is whether we need domain-specific ontologies or domain adaptation (use knowledge/models from different but related domains). Finally, the interfaces for interaction are most relevant here, and learning can be transformed into teaching: using "teaching by example" followed by program synthesis (e.g., FlashFill).
- Quantifiable measurable progress and metrics for data wrangling. We all agreed that we need benchmarks, for exploratory settings (with no predetermined task) and for settings where there is precise goal. We need metrics that estimate how close the data wrangling process gets us to some format that could be used in a data analysis tool (weka, sk-learn, Knime, ...). For the evaluation in supervised settings, it is important not to apply the semantic transformations to the test data (e.g., the test data should not use instance selection motivated by missing values).
- Non-tabular data (text, sensor, video, etc.). Most of the data wrangling literature thinks in terms of converting data into one single integrated minable view, in a tabular way, with columns (features) and rows (examples). But many problems do not fit this setting. Some may use embeddings to convert between representations, especially in NLP, but in other

cases the representations are created (possibly automatically by learning representations techniques), and both the format and the domain may change with time and must require adaptation.

- Feedback from the users: finally, and related to many of the items above, it is very important for semi-automated approaches or for improving and validating the fully-automated ones to have feedback from the users. For instance, Flashfill is an example of simple feedback, but there are other ways of feedback (reinforcement learning, preferences, such as collaborative filtering, etc.). Another point that was made is that intervening early in the process might be easier and more effective.

As possible actions from this workshop we suggested the collection of related papers, the analysis of what ETL / DS tools provide nowadays, having a look at sites such as "frictionless data" https://frictionlessdata.io/ including csv files and code for wrangling, the derivation of new metrics and collection of datasets. As a more long-term action we talked about a possible challenge/competition.

4.4 Breakout session on "Data-driven + / vs knowledge-based techniques (learning + / vs automated reasoning)"

Andrea Passerini (University of Trento, IT)

The discussion started by pointing out that automated reasoning is being largely overlooked nowadays because of the hype on machine learning and especially deep learning. However, there is a strong potential in the combination of machine learning techniques and automated reasoning / knowledge-based approaches, which is definitely worth exploring.

Three main ways of combining these fields were identified:

- using machine learning to improve reasoning systems
- using reasoning techniques to improve machine learning systems, e.g. by adding a reasoning layer on top of machine learning models
- learning to reason

One non-trivial aspect which had to be addressed was what is meant by knowledge. Generally speaking, any model made out of data could be considered knowledge. The consensus was that we talk about knowledge meaning explicit knowledge, see also the distinction between explicit and implicit knowledge in psychology.

A second relevant aspect was that the distinction between learning and reasoning does not imply a distinction between data-driven and knowledge-driven approaches, as learning itself can be both data-driven and knowledge-driven, and the same holds for reasoning. The distinction between data-driven and knowledge-driven approaches should thus be complemented with the distinction between induction and deduction.

In terms of usefulness of the combination, these are the main opportunities we identified:

- More accurate models
- More efficient models (one-shot learning)
- Reliable models
- Interpretable models
- Augmenting humans (fix reasoning biases, improve reasoning capabilities)

When talking about more developing accurate models, the field of statistical relational learning is an obvious candidate. However, many questions remain unanswered:

- Do we need statistical relational learning, when plenty of data is available?
- The computational cost of these systems is substantially higher that the one of deep learning systems (also thanks to advances in hardware), can this gap be filled in some way?

When talking about interpretable models and augmenting humans, the need for understanding the principles and limitations of human reasoning clearly emerged as a necessary and poorly explored aspect (see e.g. Thinking, fast and slow by Daniel Kahneman). From this perspective, the fact that existing approaches have mostly focused on propositional and first order logic seems suboptimal or at least incomplete. Other modalities should be better explored, like:

- argumentation
- confirmation theory
- causality
- counterfactual reasoning
- conversational reasoning

When talking about reliability of learning systems, while explainability clearly helps in building trust, it is not always required:

- Technologies get accepted and used, even if we do not fully understand them.
- Unsat proofs from SAT solvers can be huge (but they can be verified by simple proof checkers)
- We trust black boxes all the times, they are called humans!
- There are two different kinds of trust – the one that leads us to trust in people and the one that lets us to trust engineered systems (planes, etc.)

4.5 Breakout Session on "Human-in-the-Loop Automatic Data Science, and How to Avoid Ignorant Use"

Joaquin Vanschoren (TU Eindhoven, NL) and Holger H. Hoos (Leiden University, NL)

An important question arising in the context of automating data science concerns the degree of automation that is feasible or desirable. This breakout session started from a broad agreement that complete automation is currently infeasible as well as conceptually problematic. Two groups explored and discussed the concept of human-in-the-loop AutoDS, i.e., automated data science that supports human users with a certain level of expertise rather than aiming to replace them, and how to avoid ignorant use of human-in-the-loop AutoDS systems.

4.5.1 Human-in-the-loop AutoDS

Often, a user does not know precisely what she wants. To effectively aid such users, it seems inevitable to include user modelling in the system: the AutoDS system should learn to understand the intentions and limitations of specific users. This leads to 'personalized data science': AutoDS systems should figure out what matters to their users, and how to best support them in achieving their objectives. Realising such capabilities also requires expertise in human-computer interaction. The objectives and goals of human users of an AutoDS system (or any data science tool) will be substantially affected by their level of knowledge. AutoDS systems should assess this, for example, by running experiments designed to assess a user's expertise and tendency to blindly trust the system. Especially for inexperienced users, advice such as 'people who used this model/technique also found that model/technique useful' may be helpful.

The existing research area of preference learning is highly relevant in this context. One useful approach would be for the AutoDS system to start processing given data and ask the user for input at carefully chosen points in time. Interesting research questions arising in this context include the following: Where in a data science pipeline can the user meaningfully / effectively add knowledge, and where is it most needed? Which kind of user feedback should be elicited if there is a limit on the number of interactions or the amount of overall time allotted to them? Specific questions to elicit user preferences are "Can you understand this model?", or "I have removed these outliers, was that OK?" There is a limit on the amount of feedback a user is willing to give, so the AutoDS system should be conscientious in requesting it. Considering that humans often work very effectively with visualisations, these should likely play a key role in such interactions.

Another mode of interaction that may be interesting to consider is that found in so-called centaur teams in chess playing, where a human player suggests a move to chess program (before actually making the move), and the program analyses and demonstrates what will likely happen. This type of interaction has been shown to lead to powerful interactions and strength of play beyond that of the best chess programs.

Overall, it seems that finding compelling answers to the questions arising in the context of human-in-the-loop systems requires a significant shift in research focus for many AutoDS researchers; for example, there is currently little interest in user studies. An open question is how to incentivise researchers to work on these questions.

4.5.2 How to avoid ignorant use of AutoML/AutoDS?

Ignorant use (in the sense of use without necessary understanding of key characteristics and limitations) of machine learning and data science methods can lead to poor performance, misleading results and ultimately, incorrect or harmful decisions. Considering the degree to which machine learning and data science techniques are starting to be used within organisations without access to the prerequisite expertise, serious problems are bound to arise, especially as increasing automation promises to make data science techniques more accessible to non-experts. A crucial question therefore is how to prevent or alleviate the problems arising from ignorant use as much as possible.

First, we need better education and training. Many online courses now promise to make anyone taking them into a data scientist with minimal effort and time investment. Worse, there is a misperception that AutoDS will soon completely eliminate the need for data scientists. It is important for experts in AutoDS to actively warn against this view. One idea is to generate counterexamples where AutoDS does not work yet, and share war stories to make people more aware of current pitfalls. It may also be useful to start a blog on how not to automate data science.

Second, we need to provide guardrails against ignorant use of AutoDS. For instance, users could be prevented from or warned against performing multiple comparisons without proper statistical correction. Yet, distributed reuse is difficult to control. Other preventive measures could include that learned models should refuse to make predictions if the inputs are too different from the training data. However, if the model is poor, this may be difficult to recognise. Overall, the idea of guardrails against ignorant use raises interesting research questions: What are conditions that should be met in order to safely use a method? Can we learn those? How can we best support users in meeting these conditions?

Generally, the automation of data science is likely subject to a generalised form of Wiener's laws of aviation and human error, specifically: Digital devices tune out small errors while creating opportunities for large errors.

With this in mind, and related to the second point above, it seems useful to distinguish two types of automation. The first is to eliminate drudgery, as in a washing machine that (for most purposes) eliminates the need for manual washing. The second is to provide high-level oversight to help detect, avoid or compensate for human error, as in an fly-by-wire system that would not accept control inputs from a pilot that would stall the aircraft. It appears that good progress is being made in the area of drudgery automation, i.e., in automating tasks that data scientists have to do routinely, such as hyperparameter optimisation. Less attention has been paid so far to the second type of automation, which may hold the key to avoiding many of the pitfalls of ignorant use of AutoDS systems.

Overall, it seems useful to bracket human expertise by the two types of automation, which we may call a "human in the centre" approach. To avoid complacency, it may also be useful to occasionally let humans perform tasks that can be automated (this has, in fact, be proposed in aviation to counteract the detrimental effects of autopilot use on human pilots' skills). Finally, it may be interesting to investigate to which degree an AutoDS system could recognise when it can automate tasks for a specific user, and how to do so safely, without eliminating or reducing important elements of human judgement.

It seems clear that, taking a the "human in the centre" approach, increasing automation from both sides will gradually reduce the role of the human user or operator. Whether this will ultimately make it possible to completely eliminate the need for a human expert is an open question; even if it were possible, it is unclear whether this would a desirable goal to achieve. As long as human experts play an important role in the data science process, it seems crucial to ensure that they have an appropriate level of knowledge and preparation that allows them to safely use the system – perhaps in the form of a meaningfully defined 'data science license'.

Participants

- Leman Akoglu
Carnegie Mellon University – Pittsburgh, US
- Mitra Baratchi
Leiden University, NL
- Michael R. Berthold
Universität Konstanz, DE
- Hendrik Blockeel
KU Leuven, BE
- Pavel Brazdil
University of Porto, PT
- Ray G. Butler
Butler Scientifics – Barcelona, ES
- Remco Chang
Tufts University – Medford, US
- Felipe Leno da Silva
University of São Paulo, BR
- Tijl De Bie
Ghent University, BE
- Luc De Raedt
KU Leuven, BE
- Peter Flach
University of Bristol, GB
- Paolo Frasconi
University of Florence, IT
- Elisa Fromont
University of Rennes, FR
- Jose Hernandez-Orallo
Technical University of Valencia, ES
- Holger H. Hoos
Leiden University, NL
- Frank Hutter
Universität Freiburg, DE
- Tobias Jacobs
NEC Laboratories Europe – Heidelberg, DE
- Lars Kotthoff
University of Wyoming – Laramie, US
- Nada Lavrac
Jozef Stefan Institute – Ljubljana, SI
- Kevin Leyton-Brown
University of British Columbia – Vancouver, CA
- Jefrey Lijffijt
Ghent University, BE
- Zhengying Liu
University of Paris Sud – Orsay, FR
- Siegfried Nijssen
UC Louvain, BE
- Andrea Passerini
University of Trento, IT
- María Pérez-Ortiz
University of Cambridge, GB
- Bernhard Pfahringer
University of Waikato, NZ
- Kai Puolamäki
University of Helsinki, FI
- Matteo Riondato
Two Sigma Investments LP – New York, US
- Ute Schmid
Universität Bamberg, DE
- Marc Schoenauer
INRIA Saclay, FR
- Michele Sebag
CNRS, FR
- Padhraic Smyth
University of California – Irvine, US
- Alexandre Termier
University Rennes, FR
- Stefano Teso
KU Leuven, BE
- Heike Trautmann
Universität Münster, DE
- Isabel Valera
MPI für Intelligente Systeme – Tübingen, DE
- Matthijs van Leeuwen
Leiden University, NL
- Joaquin Vanschoren
TU Eindhoven, NL
- Jilles Vreeken
Universität des Saarlandes, DE
- Andreas Wierse
SICOS BW GmbH – Stuttgart, DE
- Christopher Williams
University of Edinburgh, GB