Automated Reasoning and Ontology

Jeff Z. Pan, Jianfeng Du

Synonyms

Logical reasoning, inference, semantic computing, approximate reasoning

Definitions

Reasoning is the process of deriving conclusions in a logical way. Automatic reasoning is concerned with the construction of computing systems that automate this process over some knowledge bases.

Automated Reasoning is often considered as a sub-field of Artificial Intelligence. It is also studied in the fields of theoretical computer science and even philosophy.

Jianfeng Du
Guangdong University of Foreign Studies,
e-mail: jfdu@gdufs.edu.cn

Overview

The development of formal logic (Frege (1884)) played a big role in the field of automated reasoning, which itself led to the development of artificial intelligence.

Historically, automated reasoning is largely related to theorem proving, general problem solvers and expert systems (cf. the section of 'A Bit of History'). In the context of big data processing, automated reasoning is more relevant to modern knowledge representation languages, such as the W3C standard Web Ontology Language OWL (https://www.w3.org/TR/owl2-overview/), in which a knowledge base consists of a schema component (TBox) and a data component (ABox).

From the application perspective, perhaps the most well known modern knowledge representation mechanism is Knowledge Graph (Pan et al (2016b, 2017)). In 2012, Google popularised the term 'Knowledge Graph' by using it for improving its search engine. Knowledge graphs are then adopted by most leading search engines (such as Bing and Baidu) and many leading IT companies (such as IBM and Facebook). The basic idea of Knowledge Graph is based on the knowledge representation formalism called Semantic Networks. There is a modern W3C standard for semantic networks called RDF (Resource Description Framework, https://www.w3.org/TR/rdf11-concepts/). Thus RDF/OWL graphs can be seen as exchangeable knowledge graphs in the big data era.

While this entry will be mainly about automated reasoning techniques in the big data era, their classifications, key contributions, typical systems, as well as their applications, it starts with a brief introduction of the history.

A Bit of History

Many consider the Cornell Summer Meeting of 1957, which brought together many logicians and computer scientists, as the origin of automated reasoning.

The first automated reasoning systems were theorem provers, systems that represent axioms and statements in First Order Logic and then use rules of logic, such as modus ponens, to infer new statements. The first system of this kind is the implementation of Presburger's decision procedure (which proved that the sum of two even numbers is even) by Davis (1957).

Another early type of automated reasoning system were general problem solvers, which attemp to provide a generic planning engine that could represent and solve structured problems, by decomposing problems into smaller more manageable subproblems, solving each sub-problem and assembling the partial answers into one final answer. The first system of this kind is Logic Theorist from Newell et al (1957).

The first practical applications of automated reasoning were expert systems, which focused on much more well-defined domains than general problem solving, such as medical diagnosis or analysing faults in an aircraft, and on more limited implementations of First Order Logic, such as modus-ponens implemented via IF-THEN rules. One of the forerunners of these systems is MYCIN by Shortliffe (1974).

Since 1980s, there have been prosperous studies of practical subsets of First Order Logics as ontology languages, such as Description Logics (Baader et al (2003)) and Answer Set Programming (Lifschitz (2002)), as well as the standardisation of ontology language OWL (version 1 in 2004 and version 2 in 2009). The wide adoption of Ontology and Knowledge Graph (Pan et al (2016b, 2017)), including by Google and many other leading IT companies, confirms the status of ontology language in big data era.

In the rest of the entry, we will focus on automated reasoning with ontology languages.

Classification

There can be different ways of classifying research problems related to automated ontology reasoning.

From the *purpose* point of view, automatic ontology reasoning can be classified into (1) deductive ontology reasoning (Levesque and Brachman (1987)), which draws conclusions from given premises, (2) abductive ontology reasoning (Colucci et al (2003)), which finds explanations for observations that are not consequences of given premises, as well as (3) inductive ontology reasoning (Lisi and Malerba (2003)), which concludes that all instances of a class has a certain property if some instances of the class has the property.

From the *direction* point of view, automatic ontology reasoning can be classified into (1) forward reasoning (Baader et al (2005)), in which the inference starts with the premises, moves forward and ends with the conclusions, (2) backward reasoning (Grosof et al (2003)), in which the inference starts with the conclusions, moves backward and ends with the premises, as well as (3) bi-directional reasoning (MacGregor (1991)) in which the inference starts with both the premises and the conclusions, moves forward and backward simultaneously or interactively, until the intermediate conclusions obtained by forward steps include all intermediate premises required by backward steps.

From the *monotonicity* point of view, automatic ontology reasoning can be classified into (1) monotonic ontology reasoning in which no existing conclusions will be dropped when new premises are added, as well as (2) nonmonotonic ontology reasoning (Quantz and Suska (1994)) in which some existing conclusions can be dropped when new premises are added.

From the *scalability* point of view, automatic ontology reasoning can be classified into (1) parallel ontology reasoning (Bergmann and Quantz (1995)), in which reasoning algorithms can exploit multiple computation cores in a computation nodes and (2) distributed ontology reasoning (Borgida and Serafini (2003); L Serafini (2005)), in which reasoning algorithms can exploit a cluster of computation nodes. Scalable ontology reasoning is also often related to strategies of modularisation (Suntisrivaraporn et al (2008)) and approximation (Pan and Thomas (2007)).

From the *mobility* point of view, automated ontology reasoning can be classified into (1) reasoning with temporal ontologies (Artale and Franconi (1994)), in which the target ontologies contain temporal constructors for class and property descriptions, and (2) stream ontology reasoning (Stuckenschmidt et al (2010); Ren and Pan (2011)), which, given some continuous updates of the ontology, requires updating reasoning results without naively re-computing all results.

From the *certainty* point of view, automatic reasoning can be classified into (1) ontology reasoning with certainty in which both premises and conclusions are certain and either true or false, as well as (2) uncertainty ontology reasoning (Koller et al (1997)) in which either premises or conclusions are uncertain and often have truth values between 0/-1 and 1. There are different kind of uncertainties within ontologies, such as probabilistic ontologies (Koller et al (1997)), fuzzy ontologies (Straccia (2001)) and possibilistic ontologies (Qi et al (2011)).

Key Contributions

The highlight on contributions of automated ontology reasoning is the standardisation of the Web Ontology Language OWL.

The first version of OWL (or OWL 1) was standardised in 2004. It is based on the $\mathcal{SHOIQ}$ DL (Horrocks and Sattler (2005)). However, there are some limitations of OWL 1:

1. the datatype support is limited (Pan and Horrocks (2006));
2. the only sub-language, OWL-Lite, of OWL 1 is not tractable;
3. the semantics of OWL 1 and RDF are not fully compatible (Pan and Horrocks (2003)).

The second version of OWL (or OWL 2) was standardised in 2009. It is based on the $\mathcal{SROIQ}$ DL (Horrocks et al (2006)). On the one hand, OWL 2 has more expressive power, such as the stronger support of datatypes (Pan and Horrocks (2006); Motik and Horrocks (2008)) and rules (Krtzsch et al (2008)). On the other hand, OWL 2 has three tractable sub-languages, including OWL 2 EL (Baader et al (2005)), OWL 2 QL (Calvanese et al (2007)) and OWL 2 RL (Grosof et al (2003)).

This two-layer architecture of OWL 2 allows approximating OWL 2 ontologies to those in its tractable sub-languages, such as approximations towards OWL 2 QL (Pan and Thomas (2007)), towards OWL 2 EL (Ren et al (2010)) and towards OWL 2 RL (Zhou et al (2013)), so as to exploit efficient and scalable reasoners of the sub-languages. The motivation is based on the fact that real-world knowledge and data are hardly perfect or completely digitalised.

Typical Reasoning Systems

Below are descriptions of some well known OWL reasoners (in alphabetical order).

CEL

CEL (Baader et al (2006)) is a LISP-based reasoner for $\mathcal{EL}+$ (Baader et al (2008)), which covers the core part of OWL 2 EL. CEL is the first reasoner for the description logic $\mathcal{EL}+$, supporting as its main reasoning task the computation of the subsumption hierarchy induced by $\mathcal{EL}+$ ontologies.

ELK

ELK (Kazakov et al (2012)) is an OWL 2 EL reasoner. At its core, ELK uses a highly optimised parallel algorithm (Kazakov et al (2011)). It supports stream reasoning in OWL 2 EL (Kazakov and Klinov (2013)).

FaCT

FaCT (Horrocks (1998)) is a reasoner for the description logic $\mathcal{SHIF}$ (OWL-Lite). It is the first modern reasoner that demonstrates the feasibility of using optimised algorithms for subsumption checking in realistic applications.

FaCT++

FaCT++ (Tsarkov and Horrocks (2006)) is a reasoner for (partially) OWL 2. It is the new generation of the well-known FaCT reasoner is implemented using C++, with a different internal architecture and some new optimisations.

HermiT

HermiT (Glimm et al (2014)) is a reasoner for OWL 2. It is the first publicly available OWL 2 reasoner based on a hypertableau calculus (Motik et al (2009)), with a highly optimised algorithm for ontology classification (Glimm et al (2010)). HermiT can handle DL Safe rules (Motik et al (2005)) on top of OWL 2.

Konclude

Konclude (Steigmiller et al (2014b)) is a reasoner for OWL 2. It supports almost all datatypes in OWL 2. Konclude implements a highly optimised version of tableau calculus enhanced with tableau saturation (Steigmiller and Glimm (2015)). It supports parallel reasoning and nominal schemas (Steigmiller et al (2014a)) and DL-safe rules.

Mastro

Mastro (Calvanese et al (2011)) is an Ontology-Based Data Access (OBDA) management system for OWL 2 QL. It allows data to be managed external relational data management or data federation systems. It uses the Presto algorithm (Rosati and Almatelli (2010)) for query rewriting.

Ontop

Ontop (Calvanese et al (2016)) is an Ontology-Based Data Access (OBDA) management system for RDF and OWL 2 QL, as well as SWRL with limited forms of recursions. It also supports efficient SPARQL-to-SQL mappings via R2RML (Rodriguez-Muro and Rezk (2015)). Ontop has some optimisations on query rewriting based on database dependencies (Rodriguez-Muro et al (2013)).

Pellet

Pellet (Sirin et al (2007)) is a reasoner for OWL 2. It also has dedicated support for OWL 2 EL. It incorporates optimisations for nominals, conjunctive query answering, and incremental reasoning.

Racer

Racer (Haarslev and Mller (2001)) is a reasoner for OWL 1. It has a highly optimised version of tableau calculus for the description logic $\mathcal{SHIQ}(D)$ (Horrocks and Patel-Schneider (2003)).

RDFox

RDFox (Motik et al (2014)) is a highly scalable in-memory RDF triple store that supports shared memory parallel datalog (Ceri et al (1989)) reasoning. It supports stream reasoning (Motik et al (2015b)) and has optimisations for owl:sameAs (Motik et al (2015a)).

TrOWL

TrOWL (Thomas et al (2010)) is a highly optimised approximate reasoner (Pan et al (2016a)) for OWL 2. TrOWL outperforms some sound and complete reasoners in the time-constrained ORE (Ontology Reasoner Evaluation) competitions designed for sound and complete ontology reasoners. TrOWL has stream reasoning capabilities for both OWL 2 and OWL 2 EL (Ren and Pan (2011); Ren et al (2016)). It supports local closed world reasoning in NBox, or closed predicates (Lutz et al (2013)).

Applications

Automated ontology reasoning has been widely used in Web applications, such as for content management (BBC), travel planning and booking (Skyscanner), web search (Google, Bing, Baidu).

It is also being applied in a growing number of vertical domains. One typical example is life science. For instance, OBO foundry includes more than 100 biological and biomedical ontologies. The SNOMED-CT (Clinical Terminology) ontology is widely used in healthcare systems of over 15 countries, including US, UK, Australia, Canada, Denmark and Spain. It is also used by major US providers, such as Kaiser Permanete. Other vertical domains include, but not limited to, agriculture, astronomy, oceanography, defence, education, energy management, geography and geoscience.

While ontologies are widely used as structured vocabularies, providing integrated and user-centric view of heterogeneous data sources in the big data era, benefits of using automated ontology reasoning include:

1. Reasoning support is critical for development and maintenance of ontologies, in particular on derivation of taxonomy from class definitions and descriptions.
2. Easy location of relevant terms within large structured vocabulary;
3. Query answers enhanced by exploiting schema and class hierarchy.

An example in the big data context is the use of ontology and automated ontology reasoning for data access in Statoil, where about 900 geologists and geophysicists use data from previous operations in nearby locations to develop stratigraphic models of unexplored areas, involving diverse schemata and TBs of relational data spread over 1000s of tables and multiple databases. Data analysis is the most important factor for drilling success. 30-70% of these geologists and geophysicists' time is spent on data gathering. The use of ontologies and automated ontology reasoning enable better use of experts' time, reducing turnaround for new queries significantly.

Outlook

Despite the current success of automated ontology reasoning, there are still some pressing challenges in the big data era, such as the following:

1. Declarative data analytics (Kaminski et al (2017)) based on automated ontology reasoning;
2. Effective approaches of producing high quality (Ren et al (2014); Konev et al (2014)) ontologies and knowledge graphs;
3. Integration of automated ontology reasoning with data mining (Lecue and Pan. (2015)) and machine learning (Chen et al (2017)) approaches.

References

Artale A, Franconi E (1994) A computational account for a description logic of time and action. In: KR1994, pp 3–14

Baader F, Calvanese D, McGuinness DL, Nardi D, Patel-Schneider PF (eds) (2003) The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press

Baader F, Brandt S, Lutz C (2005) Pushing the EL envelope. In: IJCAI2015

Baader F, Lutz C, Suntisrivaraporn B (2006) CEL—a polynomial-time reasoner for life science ontologies. In: IJCAR'06, pp 287–291

Baader F, Brandt S, Lutz C (2008) Pushing the EL envelope further. In: OWLED2008

Bergmann F, Quantz J (1995) Parallelizing Description Logics. In: KI1995, pp 137–148

Borgida A, Serafini L (2003) Distributed Description Logics. Journal on data semantics pp 153–184

Calvanese D, Giacomo GD, Lembo D, Lenzerini M, Rosati R (2007) Tractable reasoning and efficient query answering in description logics: The dl-lite family. J OF AUTOMATED REASONING 39:385–429

Calvanese D, De Giacomo G, Lembo D, Lenzerini M, Poggi A, Rodriguez-Muro M, Rosati R, Ruzzi M, Savo DF (2011) The MASTRO system for ontology-based data access. Journal of Web Semantics 2:43–53

Calvanese D, Cogrel B, Komla-Ebri S, Kontchakov R, Lanti D, Rezk M, Rodriguez-Muro M, Xiao G (2016) Ontop: Answering SPARQL Queries over Relational Databases. Semantic Web Journal

Ceri S, Gottlob G, Tancar L (1989) What you always wanted to know about Datalog (and never dared to ask). IEEE Transactions on Knowledge and Data Engineering 1:146–166

Chen J, Lecue F, Pan JZ, Chen H (2017) Learning from Ontology Streams with Semantic Concept Drift. In: IJCAI-2017, pp 957–963

Colucci S, Noia TD, Sciascio ED, Donini F (2003) Concept abduction and contraction in description logics. In: DL2003

Davis M (1957) A computer program for Presburgers' algorithm. In: Summaries of talks presented at the Summer Institute for Symbolic Logic, Cornell University, pp 215–233

Frege G (1884) Die Grundlagen der Arithmetik. Breslau: Wilhelm Kobner

Glimm B, Horrocks I, Motik B, Stoilos G (2010) Optimising Ontology Classification. In: ISWC2010, pp 225–240

Glimm B, Horrocks I, Motik B, Stoilos G, Wang Z (2014) HermiT: an OWL 2 reasoner. Journal of Automated Reasoning 53:245–269

Grosof BN, Horrocks I, Volz R, Decker S (2003) Description logic programs: combining logic programs with description logic. In: WWW2003, pp 48–57

Haarslev V, Mller R (2001) RACER system description. In: IJCAR2001, pp 701–705

Horrocks I (1998) Using an expressive description logic: FaCT or fiction? In: KR1998

Horrocks I, Patel-Schneider P (2003) From SHIQ and RDF to OWL: The making of a web ontology language. Journal of Web Semantics 1:7–26

Horrocks I, Sattler U (2005) A tableaux decision procedure for shoiq. In: IJCAI2005, pp 448–453

Horrocks I, Kutz O, Sattler U (2006) The even more irresistible sroiq. In: KR2006, pp 57–67

Kaminski M, Grau BC, Kostylev EV, Motik B, Horrocks I (2017) Foundations of declarative data analysis using limit datalog programs. In: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17, pp 1123–1130, DOI 10.24963/ijcai.2017/156, URL https://doi.org/10.24963/ijcai.2017/156

Kazakov Y, Klinov P (2013) Incremental reasoning in OWL EL without bookkeeping. In: ISWC2013, pp 232–247

Kazakov Y, Krtzsch M, Simancik F (2011) Concurrent Classification of EL Ontologies. In: ISWC2011, pp 305–320

Kazakov Y, Krtzsch M, Simancik F (2012) ELK Reasoner: Architecture and Evaluation. In: ORE2012

Koller D, Levy A, Pfeffer A (1997) P-CLASSIC: A tractable probablistic description logic. In: AAAI1997

Konev B, Lutz C, Ozaki A, Wolter F (2014) Exact Learning of Lightweight Description Logic Ontologies. In: KR2014, pp 298–307

Krtzsch M, Rudolph S, Hitzler P (2008) Description logic rules. In: ECAI2008, pp 80–84

L Serafini AT (2005) Drago: Distributed reasoning architecture for the semantic web. In: ESWC2005, pp 361–376

Lecue F, Pan JZ (2015) Consistent Knowledge Discovery from Evolving Ontologies. In: AAAI-15

Levesque HJ, Brachman RJ (1987) Expressiveness and tractability in knowledge representation and reasoning. Computational Intelligence pp 78–93

Lifschitz V (2002) Answer set programming and plan generation. Artificial Intelligence 138:39–54

Lisi FA, Malerba D (2003) Ideal Refinement of Descriptions in AL-Log. In: ICILP2003

Lutz C, Seylan I, Wolter F (2013) Ontology-Based Data Access with Closed Predicates is Inherently Intractable (Sometimes). In: IJCAI2013, pp 1024–1030

MacGregor RM (1991) Inside the LOOM description classifier. ACM SIGART Bulletin - Special issue on implemented knowledge representation and reasoning systems 2:88–92

Motik B, Horrocks I (2008) Owl datatypes: Design and implementation. In: ISWC2008, pp 307–322

Motik B, Sattler U, Studer R (2005) Query answering for OWL-DL with rules. Journal of Web Semantics 3:41–60

Motik B, Shearer R, Horrocks I (2009) Hypertableau Reasoning for Description Logics. Journal of Artificial Intelligence Research 36:165–228

Motik B, Nenov Y, Piro R, Horrocks I, Olteanu D (2014) Parallel Materialisation of Datalog Programs in Centralised, Main-Memory RDF Systems. In: AAAI2014

Motik B, Nenov Y, Piro R, Horrocks I (2015a) Handling of owl:sameAs via Rewriting. In: AAAI2015

Motik B, Nenov Y, Robert Piro IH (2015b) Incremental Update of Datalog Materialisation: The Backward/Forward Algorithm. In: AAAI2015

Newell A, Shaw C, Simon H (1957) Empirical Explorations of the Logic Theory Machine. In: Proc. of the 1957 Western Joint Computer Conference

Pan JZ, Horrocks I (2003) RDFS(FA) and RDF MT: Two Semantics for RDFS. In: Fensel D, Sycara K, Mylopoulos J (eds) ISWC2003

Pan JZ, Horrocks I (2006) OWL-Eu: Adding Customised Datatypes into OWL. Journal of Web Semantics pp 29–39

Pan JZ, Thomas E (2007) Approximating OWL-DL Ontologies. In: the Proc. of the 22nd National Conference on Artificial Intelligence (AAAI-07), pp 1434–1439

Pan JZ, Ren Y, Zhao Y (2016a) Tractable approximate deduction for OWL. Artificial Intelligence 235:95–155

Pan JZ, Vetere G, Gomez-Perez J, Wu H (2016b) Exploiting Linked Data and Knowledge Graphs for Large Organisations. Springer

Pan JZ, Calvanese D, Eiter T, Horrocks I, Kifer M, Lin F, Zhao Y (2017) Reasoning Web: Logical Foundation of Knowledge Graph Construction and Querying Answering. Springer

Qi G, Ji Q, Pan JZ, Du J (2011) Extending Description Logics with Uncertainty Reasoning in Possibilistic Logic. In: International Journal of Intelligent Systems

Quantz JJ, Suska S (1994) Weighted defaults in description logics: Formal properties and proof theory. In: Annual Conference on Artificial Intelligence, pp 178–189

Ren Y, Pan JZ (2011) Optimising Ontology Stream Reasoning with Truth Maintenance System. In: CIKM 2011

Ren Y, Pan JZ, Zhao Y (2010) Soundness Preserving Approximation for TBox Reasoning. In: AAAI2010

Ren Y, Parvizi A, Mellish C, Pan JZ, van Deemter K, Stevens R (2014) Towards Competency Question-driven Ontology Authoring. In: Proc. of 11th Conference on Extended Semantic Web Conference (ESWC 2014).

Ren Y, Pan JZ, Guclu I, Kollingbaum M (2016) A Combined approach to Incremental Reasoning for EL Ontologies. In: RR2016

Rodriguez-Muro M, Rezk M (2015) Efficient SPARQL-to-SQL with R2RML mappings. Journal of Web Semantics

Rodriguez-Muro M, Kontchakov R, Zakharyaschev M (2013) Query Rewriting and Optimisation with Database Dependencies in Ontop. In: DL2013

Rosati R, Almatelli A (2010) Improving Query Answering over DL-Lite Ontologies. In: KR2010, pp 290–300

Shortliffe EH (1974) MYCIN: A rule-based computer program from advising physicians regarding antimicrobial therapy selection. PhD thesis, Stanford University

Sirin E, Parsia B, Grau B, Kalyanpur A, Katz Y (2007) Pellet: A practical owl-dl reasoner. Journal of Web Semantics 5:51–53

Steigmiller A, Glimm B (2015) Pay-As-You-Go Description Logic Reasoning by Coupling Tableau and Saturation Procedure. Journal of Artificial Intelligence Research 54:535–592

Steigmiller A, Glimm B, Liebig T (2014a) Reasoning with Nominal Schemas through Absorption. Journal of Automated Reasoning 53:351–405

Steigmiller A, Liebig T, Glimm B (2014b) Konclude: System Description. Journal of Web Semantics 27

Straccia U (2001) Reasoning within fuzzy description logics. Journal of Artificial Intelligence Research 14:147–176

Stuckenschmidt H, Ceri S, Valle ED, van Harmelen F (2010) Towards expressive stream reasoning. In: Semantic Challenges in Sensor Networks, no. 10042 in Dagstuhl Seminar Proceedings

Suntisrivaraporn B, Qi G, Ji Q, Haase P (2008) A modularization-based approach to finding all justifications for OWL DL entailments. In: ASWC2008, pp 1–15

Thomas E, Pan JZ, Ren Y (2010) TrOWL: Tractable OWL 2 Reasoning Infrastructure. In: ESWC2010

Tsarkov D, Horrocks I (2006) FaCT++ description logic reasoner: System description. In: IJCAR2006, pp 292–297

Zhou Y, Grau BC, Horrocks I, Wu Z, Banerjee J (2013) Making the most of your triple store: query answering in OWL 2 using an RL reasoner. In: WWW2013, pp 1569–1580