Rulelog: Highly Expressive Semantic Rules with Scalable Deep Reasoning

Benjamin N. Grosof[1], Michael Kifer[23], and Paul Fodor[23]

[1] Accenture, USA
[2] Coherent Knowledge Systems, USA
[3] Stony Brook University, New York, USA
michael.kifer, paul.fodor@coherentknowledge.com

Abstract. In this half-day tutorial, we cover the fundamental concepts, key technologies, emerging applications, recent progress, and outstanding research issues in the area of Rulelog, a leading approach to fully semantic rule-based knowledge representation and reasoning (KRR). Rulelog matches well many of the requirements of cognitive computing. It combines deep logical/probabilistic reasoning tightly with natural language processing (NLP), and complements machine learning (ML). Rulelog interoperates and composes well with graph databases, relational databases, spreadsheets, XML, and expressively simpler rule/ontology systems and can orchestrate overall hybrid KRR. Developed mainly since 2005, Rulelog is much more expressively powerful than the previous state-of-the-art practical KRR approaches, yet is computationally affordable. It is fully semantic and has capable efficient implementations that leverage methods from logic programming and databases, including dependency-aware smart caching and a dynamic compilation stack architecture.

Keywords: knowledge representation and reasoning, declarative logic programs, semantic rules, cognitive computing, natural language processing

1 Rulelog in Ergo

Rulelog extends Datalog (database logic) with general classical-logic-like formulas, including existentials and disjunctions, and strong capabilities for meta knowledge and reasoning, including higher-order syntax, flexible defeasibility and probabilistic uncertainty, and restraint bounded rationality that ensures worst-case polynomial time for query answering[4]. A large subset of Rulelog is in draft as an industry standard. An exciting research frontier is that Rulelog can combine closely with NLP to both interpret and generate English, including potentially for conversational NL interaction.

The most complete system today for Rulelog is Ergo Suite (Ergo) from Coherent Knowledge[5]. A subset of Rulelog is also implemented in the open-source system Flora-2[6] (a.k.a. Ergo Lite) and an earlier SILK system from Vulcan. Using Ergo rule

[4] Benjamin Grosof's work done mainly while at Coherent Knowledge.
[5] http://coherentknowledge.com
[6] http://flora.sourceforge.net

inference engine, we will illustrate Rulelog's applications in deep reasoning and representing complex knowledge such as policies, regulations/contracts, science, and terminology mappings across a wide range of tasks and domains in business, government, and academe. Examples include: legal/policy compliance, e.g., in financial services; financial reporting/accounting; health care treatment guidance and insurance; education/tutoring; security/confidentiality policies; and e-commerce marketing.

In this tutorial on Rulelog as it is implemented in Ergo, we will cover some unique KRR features, such as:

- *frame-based* object-oriented frame syntax [5] and *higher-order* statements [3, 10] for practical logical knowledge representation;
- rule identifiers and provenance, argumentation-based *defeasible reasoning* [8, 9];
- *general quantification and general formulas* [4];
- *ErgoText*, an integration of logic with controlled natural language phrases (sprinkled with variables and other syntactic elements), which are translated to logic sentences;
- probabilistic reasoning and weighted uncertainty with restraint bounded rationality, including distribution semantics [7], and evidential probability [6];
- external querying and virtual data stores;
- *dynamically* evolving knowledge [2] and *hypothetical* reasoning, including integrity constraints and alarms;
- *explanations* that are fully detailed, interactively navigable, and presented in natural language – understandable by those who are not expert in logic or programming [1].

Much of this tutorial will be dedicated to applications of Ergo, both horizontally (e.g., policy-based decisions, info Integration, analytics, human-computer interaction (HCI), search, business intelligence, risk management) and vertically (e-commerce and marketing, financial services, personalized e-learning, security and defense, biomedical, insurance, Internet of Things (IoT), social media sharing policies).

Finally, we will discuss open research topics in Ergo, such as, authoring rules starting from NL, distributed reasoning, optimization of uncertainty reasoning, equality, aggregates, integration with ASP, constraint solving, and classical logic, hypotheticals, abduction and integration with ML.

The goal of the tutorial is for the audience to walk away with an understanding of Rulelogs key innovative logical and inferencing concepts, its broad applicability, its overall advantages and limitations, a sample of some specific application areas, and its open research topics.

The intended audience for this tutorial is the rules and reasoning community (all of the RuleML+RR audience) and the assumed background of the participants is only the basics of first-order-logic and relational databases. Knowledge of declarative logic programs, XML, RDF, and SPARQL will be helpful but not required.

References

1. Andersen, C., Benyo, B., Calejo, M., Dean, M., Fodor, P., Grosof, B.N., Kifer, M., Liang, S., Swift, T.: Advanced knowledge base debugging for rulelog. In: Joint Proceedings of the

7th International Rule Challenge, the Special Track on Human Language Technology and the 3rd RuleML Doctoral Consortium, Seattle, USA, July 11 -13, 2013 (2013), http://ceur-ws.org/Vol-1004/paper8.pdf
2. Bonner, A., Kifer, M.: Transaction logic: Unifying declarative and procedural knowledge (1993), manuscript
3. Chen, W., Kifer, M., Warren, D.: HiLog: A foundation for higher-order logic programming. Journal of Logic Programming 15(3), 187–230 (February 1993)
4. Grosof, B.: Rapid text-based authoring of defeasible higher-order logic formulas, via textual logic and rulelog. In: Morgenstern, L., Stefaneas, P., Lvy, F., Wyner, A., Paschke, A. (eds.) Theory, Practice, and Applications of Rules on the Web, Lecture Notes in Computer Science, vol. 8035, pp. 2–11. Springer Berlin Heidelberg (2013), http://dx.doi.org/10.1007/978-3-642-39617-5_2
5. Kifer, M., Lausen, G., Wu, J.: Logical foundations of object-oriented and frame-based languages. Journal of ACM 42, 741–843 (July 1995)
6. Kyburg, H., Teng, C.: Uncertain Inference. Uncertain Inference, Cambridge University Press (2001)
7. Riguzzi, F., Swift, T.: Well-definedness and efficient inference for probabilistic logic programming under the distribution semantics. Theory and Practice of Logic Programming 13(2), 279–302 (2013)
8. Wan, H., Grosof, B., Kifer, M., Fodor, P., Liang, S.: Logic programming with defaults and argumentation theories. In: Int'l Conference on Logic Programming (July 2009)
9. Wan, H., Kifer, M., Grosof, B.: Defeasibility in answer set programs with defaults and argumentation rules. Semantic Web Journal (2014)
10. Yang, G., Kifer, M.: Reasoning about anonymous resources and meta statements on the Semantic Web. Journal on Data Semantics, LNCS 2800 1, 69–98 (September 2003)