High-Power Logical Representation via Rulelog, for Neural-Symbolic

(position paper, extended abstract)

Benjamin N. Grosof

Coherent Knowledge
Mercer Island, Washington, USA
`http://coherentknowledge.com`
email: `firstname dot lastname at coherentknowledge dot com`

1 Introduction

Combining neural networks (NN) methods with symbolic methods is an area that is exciting in terms both of basic research and practical application. We discuss here the opportunities and challenges involved in combining NN ("neural") with logical knowledge representation and reasoning (KRR) that has high expressive power and fundamental scalability ("high-power" KRR). We describe how Rulelog KRR fits pretty well these challenges. (Note "logical" here includes probabilistic.)

2 Why to Combine KRR with NN and ML, generally

The domain-independent core of artificial intelligence (AI) consists of both KRR and machine learning (ML), as has been widely recognized within the AI research community since at least the 1980s.

There are a number of ways in which it is useful, or even required, to combine KRR methods with ML methods. The *prediction* step of ML requires reasoning. The *target* of ML is a representation. Getting business value from ML usually requires reasoning for analysis and decisions. KRR is required to effectively *combine* the results of ML from multiple ML episodes, sources, or methods. KRR is required to *accumulate* knowledge *coherently*, as knowledge ongoingly originates from ML (as well as from non-ML origins). KRR is required to *explain* knowledge understandably. Explanations are often part of required or desired analysis functionality for their own sake; they are also needed for humans to trust an automated system, check and debug knowledge/reasoning, and to help ask drill-down follow-up questions. ML-based technology today often – arguably, usually – has weaknesses in regard to such effective combination, coherent accumulation, and understandable explaining. Reasoning is useful to *supply derived facts* for ML to chew on. Reasoning is useful to *focus* ML's tasks and conjecture schemas: e.g., to provide sets of relevant features (perhaps weightedly) and/or important questions (reasoning "(sub)goals"), so as as to drive ML. Last but not least, *humans know stuff beyond what is available via ML training data*, and such knowledge is often pretty complex to state. Programming is expensive, so KRR methods are often a more cost-effective approach to entry/capture of such knowledge.

3 Symbolic Side's Representational Challenges and Requirements

For the (KRR-based) symbolic side to combine most effectively with the neural side, there are a number of representational challenges and requirements. Highly flexible expressiveness is needed; ideally any kind of knowledge can be ingested by the KRR. Thus it should be equipped feature-wise with *higher-order* syntax, logically *quantified formulas*, and strong *meta* (statements about statements). The KRR should represent *numeric weighting and uncertainty*, including (but not limited to) probabilistic and fuzzy. The KRR should treat the *evolving* character of knowledge and of the world; technically, it should have the *defeasibility* feature. The KRR should provide reasoning that is *deep*, including (but not limited to) multi-step in its logical chaining. The KRR should be *scalable* despite being expressive: not only computationally scalable to large amounts of asserted and concluded knowledge ("volume" and "velocity") but also "socially scalable" in regard to the diverse multiplicity of ML/other info sources, algorithmic methods, and underlying data samples ("variety").

4 Rulelog KRR Technology, its Advantages and Limitations

Rulelog methods meet the above set of neural-driven representational challenges pretty well overall: not only pareto-optimally among the set of available KRR approaches, but arguably better than any other KRR approach. Rulelog methods are especially strong on the meta and higher-order features, while providing scalability. They are also well suited to orchestrating / federating multiple knowledge sources and components so as to assemble and compose multiple analysis results.

Rulelog is a leading approach to semantic rules KRR. It is expressively powerful, computationally affordable, and has capable efficient implementations. A large subset of Rulelog is in draft as an industry standard[1] to be submitted to RuleML[2] and W3C[3] as a dialect of Rule Interchange Format (RIF) [1, 2].

Rulelog [4] extends database logic and well-founded declarative logic programs (LP) with:

- strong meta-reasoning, including higher-order syntax (Hilog) and rule ids (within the logical language);
- explanations of inferences;
- efficient higher-order defaults, including "argumentation theories";
- flexible probabilistic reasoning — including distribution semantics and evidential probability;
- bounded rationality, including restraint — a "control knob" to ensure that the computational complexity of inference is worst-case polynomial time;
- "omni-directional" disjunction and existential quantifiers in the rule heads;
- object-orientation and frame syntax, which subsumes RDF triples;

[1] `http://ruleml.org/rif/rulelog/rif/RIF-Rulelog.html`
[2] `http://www.ruleml.org`
[3] `http://www.w3.org`

- sound tight integration of first-order-logic ontologies including OWL; and several other lesser features, including aggregation operators and integrity constraints.

Probabilistic reasoning and tight integration with (inductive) ML is a key area of recent technology progress and ongoing R&D on Rulelog.

Rulelog also combines closely with natural language processing (NLP), in the Textual Rulelog approach [5], so as to support: human authoring of knowledge; mapping between different info schemas and terminologies; and explaining conclusions. This is another key area of ongoing R&D on Rulelog.

Implementation techniques for Rulelog inferencing include transformational compilations and extensions of *tabling* algorithms from logic programming. "Tabling" here means smart caching of subgoals and conclusions together with incremental revision of the cached conclusions when facts or rules are dynamically added or deleted [7, 8]. "Tabling" is thus a mixture of backward-direction and forward-direction inferencing.

There are both open-source and commercial tools for Rulelog that vary in their range of expressive completeness and of user convenience. They are interoperable with databases and spreadsheets, and complement inductive machine learning and natural language processing techniques. The most complete system today for Rulelog is Ergo[4], a commercial platform suite from Coherent Knowledge[5]. Ergo Lite, a.k.a. Flora-2[6], is an open source system that implements a significant subset of Rulelog reasoning. Ergo's ErgoText feature is a commercial realization of Textual Rulelog.

Rulelog has some important limitations. One is that it lacks "reasoning-by-cases", a.k.a. it is "intuitionistic"; it only concludes a disjunction if it concludes one of the disjuncts. Another limitation is that Rulelog methods are not yet optimized for probabilistic reasoning.

As has been extensively discussed in the ML and KRR literature, classical logic (e.g., first-order logic but also higher-order logic) has reasoning-by-cases but is brittle in the face of conflicting/evolving knowledge, and lacks computational scalability. Markov Logic Networks [6] are attractively flexible and principled in their ability to represent probabilistic/weighted knowledge, but are much less computationally scalable than Rulelog. Answer Set Programs [3] have reasoning-by-cases and are much less brittle than classical logic, but lack computational scalability and many of Rulelog's expressive features. For reasons of focus, we refrain from giving here additional comparison to other KRR approaches.

Applications to date of Rulelog technology include a wide range of tasks and domains in business, government, and science. Examples include: legal/policy compliance, particularly in financial services; personalized tutoring about science; and e-commerce marketing. Rulelog shines especially on representing, and deeply reasoning with, complex commonly-arising kinds of knowledge such as: mappings among terminologies, ontologies, and data schemas; policies, regulations, and contracts; and causal pathways (e.g., in science).

---

[4] `http://coherentknowledge.com/ergo-suite-platform-technology/`
[5] `http://coherentknowledge.com`
[6] `http://http://flora.sourceforge.net`

5 Future Research Directions, including Applications

An immediate direction for future R&D is to hook up Rulelog implementations to NN systems. There are many potential applications for this combination of Rulelog KRR with NN – and/or with other ML. One realm is compliance and fraud. Another realm is NL understanding in intelligence analysis and in search.

There are a number of other interesting future research directions in terms of core technology and experiments, per the overall discussion in section 1. Next, we highlight a few of these directions. One is to *feed derived data* from Rulelog to NN. Another is to combine the results of neural with other ML and structured info, including human-authored complex knowledge, e.g., that started life as English sentences. Terminology mappings and source trustworthiness are two interesting kinds of such complex knowledge represented in Rulelog. A third direction is to combine NN *word-vector* distributed representations with Textual Rulelog.

Related future directions for research on Rulelog KRR itself include to optimize its reasoning with probabilistic/weighted knowledge, and to extend Rulelog's expressiveness to *selective* reasoning-by-cases.

References

1. Boley, H., Kifer, M.: RIF Basic Logic Dialect (February 2013), `http://www.w3.org/TR/rif-bld/`, W3C Recommendation. `http://www.w3.org/TR/rif-bld/`
2. Boley, H., Kifer, M.: RIF Framework for Logic Dialects (February 2013), `http://www.w3.org/TR/rif-fld/`, W3C Recommendation. `http://www.w3.org/TR/rif-fld/`
3. Gelfond, M.: Answer Sets. In: van Harmelen, F., Lifschitz, V., , Porter, B. (eds.) Handbook of Knowledge Representation, pp. 285–316. Elsevier, Amsterdam (2008)
4. Grosof, B.N., Kifer, M., Fodor, P.: Powerful Practical Semantic Rules in Rulelog: Fundamentals and Recent Progress (July 2015), `http://2015.ruleml.org/tutorials.html`, Conference Tutorial (1.5 hours). Extended abstract is in the Symposium's Proceedings (Springer-Verlag), and contains numerous references. Also available at `http://coherentknowledge.com/publications/`.
5. 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`
6. Richardson, M., Domingos, P.: Markov Logic Networks. Machine Learning 62, 107–136 (2006)
7. Swift, T., Warren, D.: XSB: Extending the Power of Prolog using Tabling. Theory and Practice of Logic Programming 12(1-2), 157–187 (September 2012)
8. Swift, T.: Incremental Tabling in Support of Knowledge Representation and Reasoning. TPLP 14(4-5), 553–567 (2014), `http://dx.doi.org/10.1017/S1471068414000209`