Logical Rule Learning from Knowledge Graphs

Aditya Jain [* 1] **Manish Reddy Gottimukkula** [* 1] **Nilay Pochhi** [* 1]

Abstract

Often logical rules represent the domain knowledge more accurately and forms the basis for widely used logical reasoning. Learning logical rules from Knowledge Graphs (KGs) would help in understanding the data better and improve the performance of upstream tasks. Many previous attempts to learn logical rules automatically from Knowledge Graphs (KGs) rely on the complete set of rule instances for rule evaluation and thus suffer from severe computational inefficiency. To evaluate rules in a more efficient way, a novel framework *RLogic* was recently developed which aims to learn the rules directly at the schema level. In addition, RLogic incorporates deductive nature into rule learning. Through experiments Rlogic is shown to be more efficient and effective than existing state-of-the-art algorithms. Despite the visible improvement, a closer look at RLogic framework would reveal the possibility of improving it further. In this project, we aim to develop multiple ideas to modify the existing RLogic architecture to improve the logical rule mining considerably better that other *sota* methods.

1. Introduction

Logical rule mining is the process to deduce logical rules from the given data. Logical Reasoning leverages those logical rules to derive missing knowledge and allows easy generalization to unobserved objects. Significant efforts are being made to integrate logical reasoning into neural network learning for various real-world applications.

Due to the benefits of logical rules, studying the underlying patterns in the data to learn logical rules automatically is very important. Note that logical rule is a schema level concept, while only instance level evidence can be directly observed from KGs. To bridge the gap between instance level observation and schema level abstraction, the frequency of *rule instances* is utilized to define the plausibility of the logical rules. Traditional methods are represented by association rule mining (Galárraga et al., 2013), where the score is defined as the ratio between the total number of rule instances and the total number of body instances for the corresponding rule. Then the rule mining process is to search over the entire rule space and select the rules with highest plausibility score. For example, in the KG given in Fig. 1, below two rules can be found with the high plausibility score:

$$\delta_1 := hasGrandma(x, y) \leftarrow hasMother(x, z) \wedge hasMother(z, y)$$
$$\delta_2 := hasUncle(x, y) \leftarrow hasMother(x, z_1) \wedge hasMother(z_1, z_2) \wedge hasSon(z_2, y)$$

Although many efforts have been made to learn logical rules automatically, there are two limitations for the existing studies. First, relying on all rule instances for rule evaluation is not scalable considering the huge sizes of KGs. Second, the existing methods are unable to mine rules that have no support from rule instances. For example, due to the lack of support evidence, the following rule cannot be learned from Fig. 1:

$$\delta_3 := hasUncle(x, y) \leftarrow hasGrandma(x, z) \wedge hasSon(z, y)$$

RLogic approach alleviates this problem by pushing *deductive reasoning* deeper into rule learning. Using deductive reasoning, RLogic will be able to mine δ_3. Below are the main contributions made by RLogic framework:

- Learn logical rules directly at the schema level via sampled rule instances.
- Pushing deductive reasoning deeper into rule learning to validate a rule when it lacks supporting evidence.

In this project, we would like propose a few methods to improve the existing RLogic framework. First, we would like to incorporate attention based mechanism over the predicates in the rule body. And then we will use transforms based model in the existing representation learning for better prediction of intermediate relations.

2. Related Work

There are different ways in which one can accomplish the goal of rule mining. The subsequent sections describe the related work utilizing different techniques.

2.1. Association Rule Mining

Association rule mining learns rules by searching over a large rule space. The frequency of rule instances is used to estimate the plausibility of a specific logic. AMIE (Galárraga et al., 2013) proposed PCA confidence as a criteria to prune rules under open world assumption. Efficiency is still a central challenge because association rule mining relies on the complete set of rule instances for rule evaluation.

2.2. Neural Logic Programming

Very recently, neural logic pro- gramming approaches are proposed to extend the rule mining prob- lem from counting to learning. Methods such as Neural-LP (Yang et al., 2017) enables rule instances to be softly counted via sequences of differ- entiable tensor multiplication. Because neural logic programming approaches involve large matrix multiplication and simultaneously learn logic rules and their weights, which is nontrivial in terms of optimization, they cannot handle large KGs.

2.3. Inductive Logic Programming

Mining Horn clauses has been extensively studied in the Inductive Logic Programming (ILP) (Muggleton & de Raedt, 1994) (Nienhuys-Cheng et al., 1997). Given a set of positive examples, as well as a set of negative examples, an ILP system aims to learn logic rules which are able to entail all the positive examples while exclude any of the negative examples. Scalability is a big issue for these methods.

3. Proposal

RLogic has a deductive nature and uses deductive reasoning further for confidence prediction. The idea is to decompose inference of long rules into steps. Given a long relation path $r_{b_1}, ...r_{b_n}$ each of the relation $r_{b<=i}$ is a relation path. $r_{b<=i}$ denotes all predicates with index less than or equal to i. The target relation is defined as r_I.

The score $p_{r_I^i|r_{b<=i}}$ at step i can be computed based on score computed at previous step $p_{r_I^{i-1}|r_{b<=i-1}}$. It has two substeps at each step i. 1 Target Relation Prediction :- Target relation prediction aims to predict target relation r_I^i based on the previous two-relation sequence $[r_I^{i-1}, r_{b_i}]$ as input. Both the embedding are concatenated to generate the representation of prefix $r_{b<=i}$. We use weighted sum learns embedding to represent target relation r_I^i by "softly" adding up all possible derivations at step i.

Our aim is to add attention mechanism to the predicates in the rule body such that contributions from multiple predicates would contribute to result in the unknown intermediate relations. And then we would like to use the transformers in place of current fully connected neural network layers to classify the relations.Combining above ideas should result in improved performance compared to the previous RLogic results.

4. Timeline

Following is the schedule we plan on following.

- **Week 5** - Basic experiments and getting familiarized with codebase and datasets.
- **Week 6-7** - Implement Transformer based method and other minor proposal additions for the current method and get the results for a single Family dataset.
- **Week 8** - Run the model on different datasets to get more thorough results.
- **Week 9-10** - Write the report, focus on collating the results and performing some analysis on the results.

References

Galárraga, L. A., Teflioudi, C., Hose, K., and Suchanek, F. Amie: Association rule mining under incomplete evidence in ontological knowledge bases. In *Proceedings of the 22nd International Conference on World Wide Web*, WWW '13, pp. 413–422, New York, NY, USA, 2013. Association for Computing Machinery. ISBN 9781450320351. doi: 10.1145/2488388.2488425. URL https://doi.org/10.1145/2488388.2488425.

Langley, P. Crafting papers on machine learning. In Langley, P. (ed.), *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)*, pp. 1207–1216, Stanford, CA, 2000. Morgan Kaufmann.

Muggleton, S. and de Raedt, L. Inductive logic programming: Theory and methods. *The Journal of Logic Programming*, 19-20:629–679, 1994. ISSN 0743-1066. doi: https://doi.org/10.1016/0743-1066(94)90035-3. URL https://www.sciencedirect.com/science/article/pii/0743106694900353. Special Issue: Ten Years of Logic Programming.

Nienhuys-Cheng, S.-H., Wolf, R. d., Siekmann, J., and Carbonell, J. G. *Foundations of Inductive Logic Programming*. Springer-Verlag, Berlin, Heidelberg, 1997. ISBN 3540629270.

Yang, F., Yang, Z., and Cohen, W. W. Differentiable learning of logical rules for knowledge base reasoning, 2017. URL https://arxiv.org/abs/1702.08367.