Rule Mining from Text: A Semantic-Relational Logic Approach

Jingbin Li[1], Xueli Liu[1] (✉), Mingyu Gao[1], Jian Yu[1], and Wenjun Wang[1,2,3]

[1] College of Intelligence and Computing, Tianjin University, China
{2023244193, xueli, 3021244167, yujian, wjwang}@tju.edu.cn
[2] Yazhou Bay Innovation Institute, Hainan Tropical Ocean University, China
[3] State Key Laboratory of Synthetic Biology, Tianjin University, China

Abstract. Logical rules are core tools for knowledge acquisition and decision support, providing clear explanations for reasoning tasks. This paper focuses on the problem of mining rules from text. Current methods primarily rely on knowledge graphs and statistical techniques, with two main limitations: (1) They struggle to effectively mine rules from rich text resources in the absence of structured data, and (2) statistical methods often miss semantically meaningful but infrequent rules. To address these challenges, we propose an application-driven framework to mine rules of interest to users from text, leveraging semantic-relational logic to ensure rule validity. This framework first applies a fine-tuned textual entailment model to select conditions based on the application aims. It then applies contrapositive reasoning to determine whether further rule expansion is necessary, ensuring the completeness of the conditions. We evaluated our framework on real-world textual datasets, applying the mined rules to knowledge graph inference tasks. Our approach surpassed existing rule-mining methods on Hit@N and MRR metrics, demonstrating its effectiveness.

Keywords: Textual Data · Rule Mining · Semantic-Relational Logic

1 Introduction

Human-readable logical rules [2] can generalize across reasoning tasks. Early research on logical rule mining focused on relational data, employing association rule algorithms to uncover patterns [10]. This gradually extended to knowledge graphs (KGs) [8], emphasizing the discovery of meaningful rules within graph structures. Traditional approaches typically generated candidate rules and evaluated them using weighted scoring mechanisms. Subsequent studies explored differentiable methods to simultaneously mine logical rules and their corresponding weights [20, 17], while scalability improvements introduced path-based extraction from knowledge bases [16, 4]. With the advancement of large language models (LLMs), ChatRule [14] pioneered leveraging the rich knowledge embedded in LLMs for logical rule mining on knowledge graphs.

Advances in pre-trained and large language models (LLMs) now enable rule mining from text, transforming unstructured data into structured knowledge for

KG completion [4], question answering [1], and decision support [12]. Current text-based rule mining methods either convert text into graphs/relational data [13, 5] for statistical analysis or use LLMs to directly extract rules. However, KG-based techniques face challenges in text applications due to the labor-intensive nature of KG construction and the limitations of statistical methods in capturing semantic coherence and low-frequency rules. LLMs, while powerful, often produce irrelevant or hallucinated rules. To address these issues, we propose a semantic-relational logic framework [6] for rule mining. First, LLMs decompose text into assertions. Next, a fine-tuned entailment model identifies implicit logical relationships to filter relevant conditions. Finally, contrapositive reasoning [3] ensures rule completeness by verifying condition sufficiency, maintaining logical rigor.

Our primary contributions are summarized as follows:

1. We propose a framework for directly extracting logical rules from textual data, overcoming the reliance on statistical methods in existing research and providing an effective approach for rule mining from text.

2. By integrating large language models with the textual entailment model, our approach enables rule mining from text based on semantic-relational logic, improving rule accuracy and interpretability.

3. We validate our method on textual datasets across multiple domains, demonstrating its superiority over existing methods on metrics such as Hit@N and MRR, confirming its effectiveness.

2 Methodology

2.1 Task Formulation

Horn rule, $X \Rightarrow H$, consists of a single head atom H and a conjunction of body atoms $(X_1 \wedge X_2 \wedge \cdots \wedge X_n)$. In this context, an atom represents a fact and serves as the fundamental, indivisible unit of a statement. Each atom includes a constant in the predicate position and at least one variable in either the subject or object position. Thus, an atom can be expressed in the form $\mathrm{Re}(A, B)$, where Re, A, and B denote the predicate, subject, and object, respectively.
Problem Definition. Given a text T, using an application-driven strategy, we select the user's interest as the conclusion Y. Our objective is to mine a collection of horn rules $S = \{r_1, r_2, \dots\}$ from the text, where each rule $r_i : X_i \Rightarrow Y_i$ corresponds to a specific application target Y_i.

2.2 General Framework

We propose a framework for mining rules from text, as shown in Figure (1). The functions of each module are described as follows:

Text Assertion Generation: A large language model is employed to decompose the text into assertions. This decomposition helps us clarify the information structure and logical relationships within the text. Based on the definition

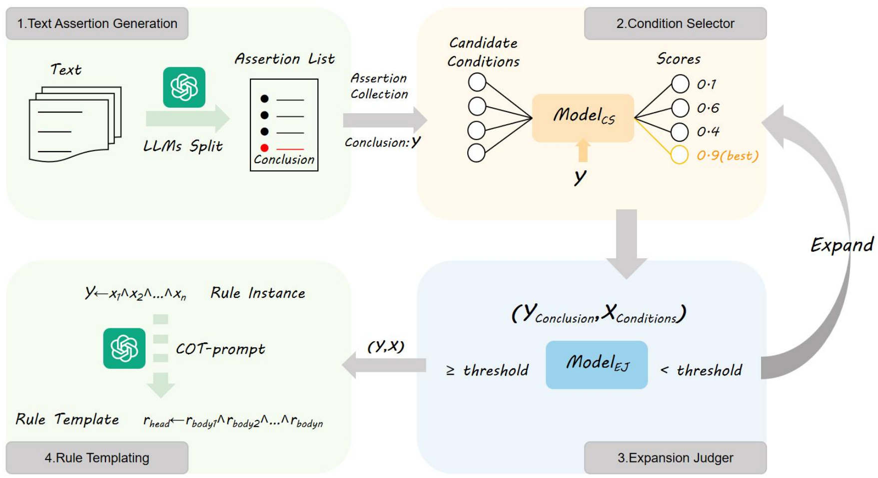


Fig. 1: An overview of the framework of Our Method.

of graph association rules [7], we define the following requirements for assertions: 1) An assertion must be a declarative sentence, typically containing a subject, predicate, and, if applicable, an object. 2) The assertion should express either a relationship between the subject and object or a property of a specific entity.

Condition Selector & Expansion Judger: The system includes two key components: the Condition Selector picks the highest-scoring assertion as the optimal condition for a given conclusion Y, starting with an empty condition set. Then, the Expansion Judger assesses if the current conditions and conclusion form a complete rule or need further conditions. If more conditions are needed, it loops back to the Condition Selector; otherwise, the rule mining is complete. Detailed mechanisms will be covered in sections 2.3 and 2.4.

Rule Templating: We utilize the Chain-of-Thought (COT) [18] method with LLMs to abstract the rule instance into a rule template. First, the rule instance is converted into a first-order logic rule, which is then further abstracted into a rule template. Relations and constants are mapped to specific semantic spaces to ensure that the extracted rules are generalized and applicable across similar contexts.

2.3 Condition Selector

To select conditions based on logical relationships between text elements, we use textual entailment (TE). Each candidate condition is assigned a relevance score for evaluation. We fine-tune a pre-trained TE model encoder to generate sentence embeddings, mapping assertions into the same semantic space. Additionally, to address the impact of connective words (e.g., "therefore," "thus") lost during decomposition, we assign an extra score to assertions with clear connectives in the original text. The scoring formula is as follows:

$$\text{Score}(C_i, H) = \cos\left(\mathbf{E}(C_i) + \sum_{k \in S} \mathbf{E}(C_k), \mathbf{E}(H)\right) + \lambda \cdot \mathbb{I}(\text{Conn}(C_i)) \quad (1)$$

where C_i is the i-th candidate condition, S is the set of selected conditions, $C_k \in S$, H is the conclusion, $\mathbf{E}(X)$ is the embedding of sentence X, $\cos(\cdot, \cdot)$ computes cosine similarity, λ is a weight, and $\mathbb{I}(\text{Conn}(C_i))$ indicates if C_i contains explicit connectives.

To improve the TE model encoder's ability to represent inference and conjunction relationships, we fine-tuned it using two datasets: 1. SNLI[4] (Stanford Natural Language Inference Dataset): This dataset contains sentence pairs labeled as entailment, contradiction, or neutral. 2. FOLIO[5] [11] (First-Order Logic Rules Dataset): This dataset includes textual rules with conditions and conclusions, assigning entailment labels to condition-conclusion pairs. The loss function is defined as the squared Euclidean distance between the two embeddings:

$$\text{Loss}_1 = \left\| \sum_{k \in S} \mathbf{E}(C_k) - \mathbf{E}(H) \right\|_2^2 \quad (2)$$

where S is the set of conditions, $C_k \in S$, H is the conclusion, $\mathbf{E}(X)$ is the embedding of X, and $\| \cdot \|_2^2$ is the squared ℓ_2-norm. Minimizing Loss_1 aligns the semantic space of conditions with the conclusion, capturing their logical relationship.

With the fine-tuned encoder in place, we use it to encode the sentences in the assertion collection and apply Equation (1) to calculate scores. Based on an application-driven strategy, we select the highest-scoring candidate condition from the assertion collection and add it to the existing condition collection.

2.4 Expansion Judger

The purpose of the Expansion Judger is to determine whether the current condition collection sufficiently supports the conclusion, i.e., whether the conditions are complete. To achieve this, we adopt contrapositive reasoning. For a given proposition, if the contrapositive holds, it signifies that the relationship between the conditions and the conclusion is both sufficient and necessary. If the contrapositive does not hold, the condition collection is incomplete. In such cases, the rule is expanded by revisiting the Condition Selector.

[4] https://nlp.stanford.edu/projects/snli/
[5] https://github.com/Yale-LILY/FOLIO

To ensure that the model can clearly distinguish between the embeddings of an assertion and its negated version, we further fine-tune it using a contrastive learning approach. Inspired by the work on sentence embeddings in SimCSE [9], the positive samples are generated by applying dropout-based random masking to the same sentence, while the negative samples consist of sentence pairs with contradictory labels from the SNLI dataset, along with manually created pairs of original and negated sentences. The cross-entropy loss function used is as follows:

$$\mathrm{Loss}_2 = -\log \frac{e^{\mathrm{sim}(\boldsymbol{h}_i, \boldsymbol{h}_i^+)/\tau}}{\sum_{j=1}^{N} \left(e^{\mathrm{sim}(\boldsymbol{h}_i, \boldsymbol{h}_j^+)/\tau} + e^{\mathrm{sim}(\boldsymbol{h}_i, \boldsymbol{h}_j^-)/\tau} \right)} \tag{3}$$

where $\boldsymbol{h}_i$ is the anchor embedding, $\boldsymbol{h}_i^+$ is the positive sample embedding, $\boldsymbol{h}_j^-$ is the negative sample embedding, $\mathrm{sim}(\cdot,\cdot)$ computes cosine similarity, τ is the temperature parameter, and N is the mini-batch size. This loss pulls semantically close embeddings together and pushes dissimilar ones apart, improving embedding quality.

3 Experiments

Given that most existing rule-mining methods focus on extracting rules from knowledge graphs for inference evaluation, we applied rules mined from text to a knowledge graph for inference to assess their performance. However, current knowledge graph datasets lack corresponding textual sources, and text datasets linked with knowledge graphs typically contain sentence-level rather than paragraph-level content. Consequently, we selected real-world textual data and leveraged large language models' capabilities, employing an ontology-based text-to-graph strategy [15] to construct our datasets.

3.1 Dataset

We validated our approach using texts from three sources: **SGAP** (Shenzhen Government Affairs Portal)[6]: Contains 2024 administrative reconsideration documents on transportation issues, detailing applications and outcomes. **CJW** (China Judgments Website)[7]: Focuses on economic crime cases, including case details, trial procedures, facts, laws, and judgments. **CGP** (Chinese Government Portal)[8]: Includes emergency education materials for public safety and disaster response.

These datasets provide diverse textual content for validation. We constructed knowledge graphs by defining ontologies and using LLMs to extract relational triples. See Table (1) for details.

[6] https://sf.sz.gov.cn/szsxzfywsfwpt/wsgk/
[7] https://wenshu.court.gov.cn/
[8] https://www.gov.cn/yjgl/yjzs.htm

Table 1: Statistics of Three Datasets

Dataset	Documents	Relations	Entities	Triples
SGAP	771	48	6572	103422
CJW	96	50	2434	20058
CGP	23	42	253	1983

3.2 Baselines

We selected three types of rule mining approaches for comparison.

Traditional Knowledge Graph Rule Mining: Methods like **RNNLogic** [16] separate rule generation and weighting to enhance each other. **NCRL** [4] uses recursive merging to identify optimal rule structures, while **Ruleformer** [19] employs a Transformer with relational attention for efficient rule mining. **Rule Mining in Relation Extraction Frameworks**: **MILR** [5] combines rule mining with consistency regularization, and **BCBR** [13] introduces a bidirectional constraint strategy to handle pseudo-rules. **LLM-Based Rule Mining**: **ChatRule** [14] leverages LLMs to generate and filter rules, integrating semantic and structural knowledge for improved rule quality and scalability.

3.3 Evaluation

For the knowledge graph reasoning task, we mask either the tail or head entity of each test triple and use the rules generated by each method to make predictions. Following prior research [4], we use Mean Reciprocal Rank (MRR) and Hits@N as evaluation metrics, with N set to 1 and 10.

3.4 Results & Discussions

As shown in Table (2), the experimental results across the three datasets demonstrate that our method performs effectively across different domains. Applying the rules mined by our approach to knowledge graph inference yields outcomes that match or even surpass those of existing rule-mining methods, highlighting our method's ability to extract meaningful and practically valuable rules. Furthermore, the quality of the rules mined by our method is high—not only capturing high-frequency, logically consistent rules but also identifying infrequent yet logically sound rules. Additionally, evaluation metrics show that, while achieving or surpassing baseline methods in Hits@N, our method consistently outperforms baselines in MRR scores. This indicates that our approach mines fewer irrelevant rules.

3.5 Ablation Study

Can the score of the Condition Selector indicate the "inference" relationship between conditions and conclusions? To validate the effectiveness of our Condition Selector model, we compared the encoder's performance

Table 2: Comparison of Different Methods Across Three Datasets

Method	SGAP			CGP			CJW		
	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR
RNNLogic	0.1638	0.6327	0.3886	0.2688	0.9679	0.5519	0.1667	0.6315	0.4005
Ruleformer	0.1886	0.6436	0.3358	0.2476	0.9486	0.5361	0.1536	0.6203	0.3733
NCRL	0.1786	0.6250	0.3871	0.2609	**0.9783**	0.5648	0.1618	0.6324	0.3933
MILR	0.1745	0.6542	0.3906	0.2208	0.9283	0.3738	0.1548	0.6408	0.4012
BCBR	**0.1964**	0.6607	0.3945	0.2250	0.9375	0.3876	0.1765	**0.6471**	0.4077
ChatRule	0.1607	0.6250	0.3713	0.1750	0.9722	0.4026	0.1429	0.6429	0.3978
Ours	**0.1964**	**0.6786**	**0.4004**	**0.2826**	**0.9783**	**0.5756**	**0.2059**	**0.6471**	**0.4360**

on textual entailment tasks before and after fine-tuning. We conducted tests on the validation sets of the SNLI, MultiNLI[9] and FOLIO datasets. To align the first-order logic rule datasets with the textual entailment task, we labeled the relationship between the rule head and rule body as "entailment". The accuracy on the validation sets is shown in Table (3). The results demonstrate that the

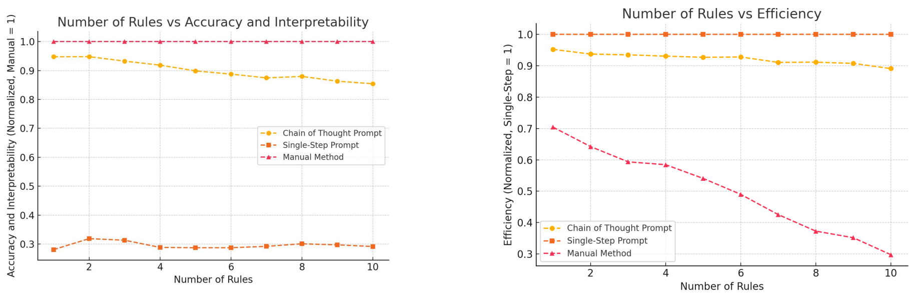
Table 3: Results on SNLI, MultiNLI and FOLIO Datasets

Model	**SNLI (Acc.)**	**MultiNLI (Acc.)**	**FOLIO (Acc.)**
CSE (Unfine-tuned)	0.42	0.45	0.29
CSE (Fine-tuned)	**0.69**	**0.75**	**0.82**

encoder can indeed capture the "inference" relationship between the rule head and rule body.

Is Chain-of-Thought (COT) prompting effective for rule standardization? Rule standardization aims to template rule instances for broader applicability. We compared single-step prompting, COT prompting, and manual methods for rule quality and time efficiency. Rule quality, assessed by accuracy and interpretability via expert review, used manual methods as the benchmark (score of 1). Time efficiency, benchmarked against single-step prompting (score of 1), measured relative performance. As shown in Figure (2), single-step prompting is the fastest but produces lower-quality templates. Manual methods achieve high quality but suffer from reduced efficiency with increasing rule complexity and domain diversity. COT prompting matches manual quality while significantly improving time efficiency, making it a balanced approach for rule standardization.

[9] https://paperswithcode.com/dataset/multinli



(a) As the number of rules increases, the accuracy and interpretability of different methods change.

(b) As the number of rules increases, the time efficiency of different methods changes.

Fig. 2: Comparison of Rule Templating Methods.

3.6 Case Study

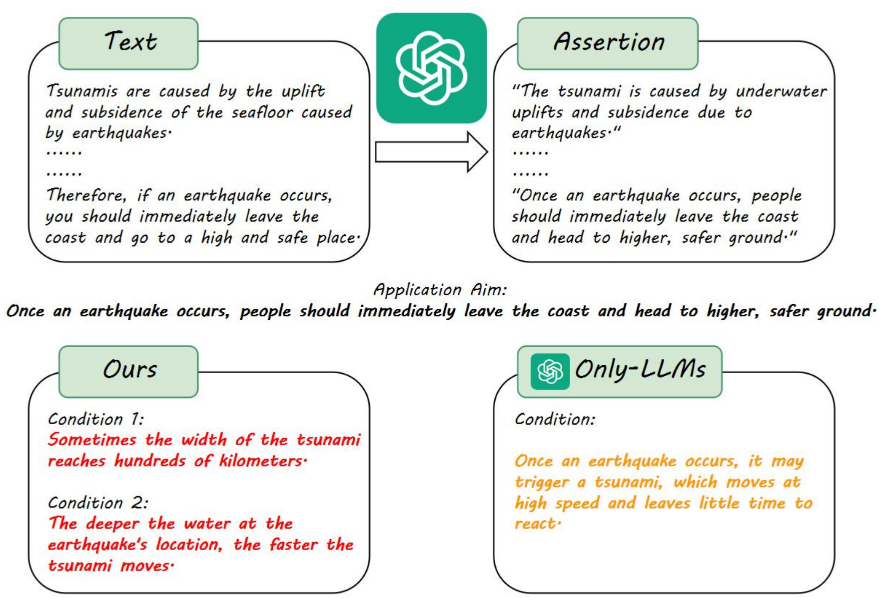


Fig. 3: Comparison of Rules Extracted by the LLMs and Our Method.

We selected a text from CGP dataset to compare the advantages of our method over the llm-based approach. The selected text discusses tsunami evacuation, aiming to extract rules that explain why people should flee the coast and move to higher ground. As shown in Figure (3), the llm-based approach only captured the idea that tsunamis are fast and require immediate evacuation, but overlooked other critical information in the text, specifically why it is necessary to leave the coast and head to higher ground. In contrast, our method accurately identified the conditional sentences needed to support the conclusion, extracting two key conditions. The case study illustrates that our proposed method can more accurately identify relevant conditions in the text, resulting in rules that offer greater practical value.

4 Conclusion

This paper introduces an application-driven framework for mining rules from textual data using semantic-relational logic, addressing the limitations of structured-data-dependent methods. Unlike traditional approaches relying on knowledge graphs and statistical techniques, our framework leverages a fine-tuned textual entailment model to select conditions based on application goals. It employs contrapositive reasoning to assess the need for rule expansion, ensuring condition completeness. Experiments show that our method outperforms existing rule-mining approaches across diverse datasets, achieving superior results in metrics like Hit@N and MRR. This framework enables the direct extraction of interpretable logical rules from text, opening new avenues for research in text-based rule mining.

Acknowledgments. This work is supported by National Key R&D Program of China (No. 2021YFC2600501)

References

1. Atif, F., El Khatib, O., Difallah, D.: Beamqa: Multi-hop knowledge graph question answering with sequence-to-sequence prediction and beam search. In: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 781–790 (2023)
2. Barwise, J.: An introduction to first-order logic. In: Studies in Logic and the Foundations of Mathematics, vol. 90, pp. 5–46. Elsevier (1977)
3. Bonnefon, J.F., Villejoubert, G.: Modus tollens, modus shmollens: Contrapositive reasoning and the pragmatics of negation. Thinking & reasoning **13**(2), 207–222 (2007)
4. Cheng, K., Ahmed, N.K., Sun, Y.: Neural compositional rule learning for knowledge graph reasoning. arXiv preprint arXiv:2303.03581 (2023)
5. Fan, S., Mo, S., Niu, J.: Boosting document-level relation extraction by mining and injecting logical rules. In: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. pp. 10311–10323 (2022)
6. Fan, W., Fu, W., Jin, R., Lu, P., Tian, C.: Discovering association rules from big graphs. Proceedings of the VLDB Endowment **15**(7), 1479–1492 (2022)

7. Fan, W., Jin, R., Liu, M., Lu, P., Tian, C., Zhou, J.: Capturing associations in graphs. Proceedings of the VLDB Endowment **13**(12), 1863–1876 (2020)
8. Galárraga, L.A., Teflioudi, C., Hose, K., Suchanek, F.: Amie: association rule mining under incomplete evidence in ontological knowledge bases. In: Proceedings of the 22nd international conference on World Wide Web. pp. 413–422 (2013)
9. Gao, T., Yao, X., Chen, D.: Simcse: Simple contrastive learning of sentence embeddings. arXiv preprint arXiv:2104.08821 (2021)
10. Getoor, L., Taskar, B.: Introduction to statistical relational learning. MIT press (2007)
11. Han, S., Schoelkopf, H., Zhao, Y., Qi, Z., Riddell, M., Zhou, W., Coady, J., Peng, D., Qiao, Y., Benson, L., et al.: Folio: Natural language reasoning with first-order logic. arXiv preprint arXiv:2209.00840 (2022)
12. Kong, G., Xu, D.L., Yang, J.B., Wang, T., Jiang, B.: Evidential reasoning rule-based decision support system for predicting icu admission and in-hospital death of trauma. IEEE Transactions on Systems, Man, and Cybernetics: Systems **51**(11), 7131–7142 (2020)
13. Liu, Y., Zhu, Z., Zhang, X., Feng, Z., Chen, D., Li, Y.: Document-level relationship extraction by bidirectional constraints of beta rules. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. pp. 2256–2266 (2023)
14. Luo, L., Ju, J., Xiong, B., Li, Y.F., Haffari, G., Pan, S.: Chatrule: Mining logical rules with large language models for knowledge graph reasoning. arXiv preprint arXiv:2309.01538 (2023)
15. Mihindukulasooriya, N., Tiwari, S., Enguix, C.F., Lata, K.: Text2kgbench: A benchmark for ontology-driven knowledge graph generation from text. In: International Semantic Web Conference. pp. 247–265. Springer (2023)
16. Qu, M., Chen, J., Xhonneux, L.P., Bengio, Y., Tang, J.: Rnnlogic: Learning logic rules for reasoning on knowledge graphs. arXiv preprint arXiv:2010.04029 (2020)
17. Sadeghian, A., Armandpour, M., Ding, P., Wang, D.Z.: Drum: End-to-end differentiable rule mining on knowledge graphs. Advances in Neural Information Processing Systems **32** (2019)
18. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q.V., Zhou, D., et al.: Chain-of-thought prompting elicits reasoning in large language models. Advances in neural information processing systems **35**, 24824–24837 (2022)
19. Xu, Z., Ye, P., Chen, H., Zhao, M., Chen, H., Zhang, W.: Ruleformer: Context-aware rule mining over knowledge graph. In: Proceedings of the 29th International Conference on Computational Linguistics. pp. 2551–2560 (2022)
20. Yang, F., Yang, Z., Cohen, W.W.: Differentiable learning of logical rules for knowledge base reasoning. Advances in neural information processing systems **30** (2017)