Analogy-based Knowledge Graph Completion Combined with Rule Mining

Y. Wang

Analogy-based Knowledge Graph Completion Combined with Rule Mining

by

Y. Wang

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Monday April 14, 2025, 4pm

Student number:	5763592	
Project duration:	July 10, 2024 – April 14, 2025	
Thesis committee:	Prof. Ujwal Gadiraju,	TU Delft, supervisor
	Prof. Pradeep Murukannaiah,	TU Delft
	Gaole He	TU Delft, daily co-supervisor

Preface

I am in the final stages of my Master's thesis and it also marks the end of my Master's career at Delft University of Technology. When I first contacted Prof Ujwal, I was overwhelmed by his help and guidance for my thesis project. I am even more thankful that he introduced me to gaole as my daily supervisor to assist me. From the initial selection of the thesis topic to the finalisation of the data mapping, the design and implementation of the experimental procedures, the deployment and application of the remote server, to the final analysis and improvement of the experimental results, gaole has been of great help to me.

Especially when I encountered difficulties, the patience and professionalism of both of them provided me with valuable advice and support in time to help me overcome the challenges. This experience has helped me to deeply understand the importance of teamwork and communication, and it has also helped me to be more determined in my academic direction.

In addition, I would like to thank my family and friends who have always given me unconditional support and encouragement during my time away from home.

With the completion of my master's thesis, I am about to embark on a new journey. Although the road ahead is full of unknowns and challenges, I believe that with the knowledge and experience I have gained at Delft University of Technology, I will be able to face all the difficulties bravely and continue to pursue higher goals in academia and life.

Y. Wang
Delft, March 2025

Abstract

Reasoning over large-scale knowledge graphs has long been dominated by embedding-based methods, which focus on representing entities and relationships in vector spaces to perform inference tasks. Despite advancements in knowledge graph completion (KGC), challenges such as data sparsity and the lack of interpretability persist. These issues are critical in biomedical areas, where prediction accuracy and explainability directly impact decision-making.

In light of these limitations, this paper proposes a KGC approach that allows it to leverage both globally structured patterns and locally analogical information. Aiming to enhance the reasoning capability and completeness of knowledge graphs, we develop a better joint inference approach based on the rules mining and knowledge graph embedding. Specifically, our approach employs knowledge graph embedding techniques to discover analogically similar triples, which are then used to construct more relevant and accurate inference paths. Subsequently, rule mining is integrated to extract structured knowledge patterns through analogical reasoning.

Experimental results demonstrate that our approach performs better than the traditional Knowledge Graph Embedding Model in some specified KGC tasks, which depend highly on the inner construction of the dataset and its relation types. The multi-hop nature of DRKG aligns well with rule-based approaches, where learned rules generalize over multiple instances, aiding in missing link prediction. Beyond improved performance, our method enhances explainability, facilitating transparent inference and enabling backtracking of key prediction results—an important feature for biomedical and high-stakes AI applications.

Contents

List of Figures

List of Tables

1 Introduction

With the explosion of AI and big data, Knowledge graphs (KGs) has become critical to properly organize and represent massive amounts of knowledge [Peng et al., 2023]. It is an essential tool for structuring and representing complex relationships between entities in various domains. Usually, KG is defined as a directed, multi-relationagraph in which edges and nodes are labeled [Kejriwal, 2022]. Nodes are treated as entities, ranging from common entities (e.g., musicians, movies, and locations) to highly domain-specific entities (e.g., proteins and viruses in the biological domain). Edges, also called predicates, represent relationships between entities (for example, the ‘treat’ relationship between compound and disease entities).

By interconnecting entities and organizing them into a network-based structure, KGs serve as a powerful framework for modeling real-world facts and facilitating relational reasoning [Tiwari et al., 2021]. Over the past decade, KGs have been widely employed in Artificial Intelligence (AI), playing a crucial role in tasks such as recommendation systems, information retrieval system, and question answering [Zou, 2020].

However, despite their extensive applications, the construction and maintenance of KGs remain formidable challenges [Hao et al., 2021]. Large-scale knowledge graphs, particularly those in specialized domains such as biomedicine, often suffer from incompleteness: a significant proportion of entities and relationships are missing due to the vastness and complexity of the data sources. This incompleteness impedes the ability of downstream AI models to perform accurate and reliable reasoning [Peng et al., 2023], as the sparsity of data and the absence of direct or indirect relational paths limit the effectiveness of existing inference techniques. Consequently, knowledge graph completion (KGC) has emerged as a critical area of research that aims to improve both the coverage and reliability of KGs.

1.1. Biomedical Knowledge Graphs

In recent years, AI4Science has driven a paradigm shift from purely data-driven methodologies to more knowledge-driven approaches in biomedical research. The research community is increasingly moving toward knowledge-enhanced learning, which incorporates structured biomedical knowledge into the learning process [Cong et al., 2018]. Biomedical Knowledge Graphs (BKGs) have emerged as a powerful tool in this context, enabling the integration of heterogeneous data sources into a unified and structured representation.

1.1.1. Multi-Hop Nature for Biomedical KGs

Unlike open-domain knowledge graphs (KGs), which often contain loosely defined relationships, biomedical knowledge graphs (BKGs) are constructed from scientifically validated and well-structured data sources such as DrugBank, Hetionet, and DisGeNET [Bonner et al., 2022]. These graphs encode relationships between biomedical entities—such as gene-disease associations, drug-target interactions, and protein-protein interactions—which often follow systematic patterns governed by biological, chemical, and functional principles.

Many biomedical relationships are inherently multi-hop, meaning that several intermediate steps connect an entity (e.g., a drug) to an outcome (e.g., disease treatment). Thus, a sequence of direct connections can lead to logically valid indirect conclusions [Lu et al., 2025]. For example, consider the following relational chain within a biomedical KG:

```
(Gene A, Affects, Protein B)
(Protein B, Involved in, Biological Pathway C)
(Biological Pathway C, Associated with, Disease D)
```

Here, multi-hop relations enable the inference that Gene A may be associated with Disease D, offering valuable insights into potential drug targets and disease mechanisms. Similarly, protein-protein interactions often exhibit structured patterns, as proteins involved in the same biological processes or signaling pathways frequently interact. Recognizing and leveraging these systematic relationships is essential for KGC, as it allows AI models to generalize from known facts to predict missing links.

1.1.2. Importance of Interpretability in Biomedical AI

Interpretability is crucial in medical AI systems because healthcare decisions directly impact patient safety, treatment effectiveness, and clinical outcomes. Unlike general knowledge graph tasks, where accuracy alone may be sufficient, biomedical applications demand transparent and explainable reasoning to ensure AI-generated insights are trustworthy, actionable, and clinically applicable [Nadkarni et al., 2021].

Multi-step explanation plays a key role in medical question-answering (QA) systems, where AI must navigate complex biomedical relationships to generate personalized treatment recommendations. For these applications, black-box predictions can be problematic, as they make it difficult for researchers and clinicians to validate AI-generated conclusions. Instead, transparent, human-readable inference pathways are essential to enhance trust, facilitate verification, and support informed clinical decision-making.

Beyond decision support, interpretability also helps mitigate inaccurate or risky predictions. In high-stakes areas such as drug discovery and precision medicine, blindly following AI-generated results can lead to misdiagnoses, ineffective treatments, or unforeseen adverse effects. By making AI predictions more interpretable, researchers and medical professionals can assess reliability, identify potential errors, and reduce costly trial-and-error processes before implementing AI-driven insights in real-world healthcare settings.

1.2. Problem Definition and Goal

Knowledge Graph Completion is an important task aimed at predicting missing triplets in KGs, which are inherently sparse and incomplete [Chen et al., 2020]. As large-scale knowledge bases continue to expand, KGC plays a crucial role in enabling a more comprehensive understanding of their structure and relationships.

Formally, a knowledge graph G is represented as a collection of triples:

$$\{(h, r, t)\} \subseteq E \times R \times E \tag{1.1}$$

where E is the set of entities and R is the set of relations [Xiong et al., 2018]. The goal of knowledge graph completion is to predict missing entities in incomplete triples, typically formulated as one of the following tasks:

$$Head\ prediction : (?, r, t) \tag{1.2}$$

$$Tail\ prediction : (h, r, ?) \tag{1.3}$$

where the missing entity (either head entity h or tail entity t) needs to be inferred based on existing knowledge within the graph.

Researchers have investigated a variety of methods for predicting missing information over the years. It is shown that embedding-based approaches have achieved remarkable success by learning low-dimensional vector representations [Ge et al., 2024]. However, despite their advanced predictive performance, embedding-based models suffer from a major limitation: lack of interpretability. These models operate in complex vector spaces, making it difficult for researchers to reason behind predictions, especially for the biomedical domain.

As an effective complementary method in KGC, rule-based approaches inspired us because they extract explicit, human-readable patterns from the knowledge graph that can explain the relationships between entities. This is consistent with our exception to build a transparent and interpretable framework for reasoning about missing entities. Specifically, we aim to develop a hybrid model that can leverage the power of vector embeddings for capturing complex relationships while simultaneously providing clear reasoning for validation and interpretability.

1.3. Motivations

The following subsections thoroughly explain the rationale behind integrating rule mining with analogy-based reasoning.

1.3.1. Why the rule mining method?

As discussed above, transparency in the reasoning process is a fundamental requirement for improving the performance of biomedical knowledge graph completion. However, KG embedding models, which encode entities and relationships as low-dimensional vectors, operate as black-box models. Their predictions rely on complex, non-intuitive mathematical transformations, making it difficult to trace how conclusions are derived. The lack of an explicit reasoning process limits their applicability in biomedical AI, where interpretability is crucial for ensuring the reliability of predictions in drug discovery, disease-gene association analysis, and other critical tasks.

In critical applications such as drug repurposing or disease diagnosis, blindly following AI-generated recommendations without insight into the underlying logic can lead to unnecessary trial-and-error costs, wasted resources, and even potential safety risks. For instance, if an embedding-based model suggests a drug-disease interaction, but the reasoning behind the prediction is unclear, researchers may need to conduct additional experiments to verify its validity, increasing both time and financial investment.

To address this limitation, rule mining provides an alternative approach by explicitly identifying logical reasoning pathways between entities. Instead of relying on abstract vector space transformations, rule-based methods infer potential missing entities and relationships by extracting structured, explainable patterns from existing knowledge graphs. This not only enhances interpretability but also ensures that the inferred knowledge is backed by logically consistent and scientifically valid reasoning.

Adaptability to Dynamic Biomedical Knowledge Graphs. Another significant challenge in biomedical KGC is the dynamic nature of medical knowledge. Advances in biomedical research constantly introduce new discoveries, novel drug-target interactions, and evolving disease classifications. Consequently, knowledge graphs are frequently updated to incorporate emerging knowledge. However, traditional embedding-based KGC methods struggle to keep up with these changes due to their static nature.

- **High Computational Cost of Retraining:** Once trained, KG embedding models require complete retraining whenever new entities or relationships are added. For large-scale biomedical knowledge graphs, this retraining process is computationally expensive, memory-intensive, and time-consuming.

- **Scalability Issues:** As knowledge graphs grow in large size, embedding-based models become increasingly inefficient due to their reliance on large amounts of labeled training data and complex storage requirements.

Rule-based KGC methods offer a flexible and scalable alternative. Since they utilize generalizable knowledge structures rather than rely exclusively on large-scale data for training. In other words, they extract generalizable relational patterns from existing knowledge, new knowledge can be continuously integrated without requiring costly retraining. By applying mined rules to newly added entities and relationships, rule-based approaches seamlessly adapt to evolving knowledge graphs, making them well-suited for rapidly changing biomedical domains.

Effectiveness in Low-Data Scenarios. Another critical advantage of rule mining is its ability to generate new knowledge even in data-scarce environments. Unlike embedding-based models, which require large-scale labeled datasets to learn implicit representations, rule-based methods leverage structured patterns within the knowledge graph itself.

Rules extracted from a subset of biomedical data often remain valid even when the knowledge graph is incomplete or partially updated. This enables rule-based methods to infer missing links with relatively little data. Moreover, since rule mining does not rely on statistical learning from massive datasets, it can still recognize and apply valid relational patterns, making it particularly robust to data sparsity, such as in cases of rare disease-drug interactions

1.3.2. Why base it on the analogy context?

Traditional rule-based knowledge graph completion (KGC) methods typically mine rules indiscriminately from the entire knowledge graph. While this brute-force approach can discover valid logical patterns, it often leads to redundant, overly specific, or low-utility rules, which can negatively impact both prediction accuracy and computational efficiency. Moreover, rule mining without context sensitivity fails to distinguish between relevant and irrelevant relational patterns, making it harder for models to generalize effectively.

To address these challenges, we introduce analogy-based rule mining, which leverages structural and semantic similarities between entities and relations to enhance rule learning. Instead of treating all triples equally, our approach groups similar triples based on a pre-trained embedding model before performing rule mining. This method enables the model to focus on triples that are more likely to yield accurate predictions while filtering out noisy or less informative paths.

By grouping triples into analogy-based clusters, our approach also reduces the search space for rule mining, making it significantly more computationally efficient than traditional methods. Instead of exhaustively searching for patterns across the entire KG, the model focuses on structurally relevant subgraphs, reducing the number of rules required while retaining high predictive accuracy.

These considerations collectively motivate our research into not only enriching the knowledge graph with high-confidence inferences but also providing a transparent and scientifically valid framework for knowledge discovery and decision-making.

1.4. Research Questions

Our study investigates a hybrid method for knowledge graph completion tasks, particularly in the biomedical field. Given the critical role of explainability in medical AI applications, our approach aims to enhance not only predictive performance but also trustworthiness in inference results. By leveraging analogy-based rule mining, we strive to develop a method that is more applicable to medical AI systems, providing effective and reliable support for researchers, doctors, and patients in clinical and scientific decision-making.

To guide our investigation, we focus on the following key research questions:

RQ: How can the analogy-based rule mining impact both predictive accuracy and interpretability compared to embedding-based methods?

We seek to evaluate whether integrating analogy-based rule mining can yield more reliable and explainable inference results than traditional embedding-based models, which often function as black-box systems. By addressing the research question, our work moves forward to bridge the gap between AI-driven knowledge inference and real-world biomedical decision-making, ultimately contributing to more trustworthy and interpretable AI systems in the biomedical domain.

1.5. Contributions

The contributions of this paper are mainly in the following aspects:

- We perform analogy-based rule mining on well-known benchmark datasets and demonstrate that it works better than embedding-based models in certain KGC tasks.
- We develop a transparent KG reasoning framework, which supports tracing back the associated facts from KGs. This design makes analogy-based rule mining more suitable for domain-specific AI applications where interpretability is essential.

1.6. Outline

The remainder of this thesis is organized as follows: Chapter 2 reviews related work in the field of Knowledge Graph Completion (KGC), providing the theoretical foundation and highlighting key methods that inspire our research. Chapter 3 describes the proposed methodology, including the experimental design, setup details, and evaluation metrics used to assess model performance. Chapter 4 presents the experimental results on three different knowledge graphs and offers detailed analysis through case studies to interpret the model's predictions. Finally, Chapter 5 summarizes the main findings, discusses the limitations of the current work, and outlines potential directions for future research.

2

Related work

2.1. KGC methods

In recent years, an increasing amount of research has been focused on the field of KGC. Numerous methods exist for completing knowledge graphs, broadly classified into embedding-based, neural network-based, rule-based, and mixed inference approaches [Shen et al., 2022]. The folloiwng four subsections will discuss each of them that leveraging different principles of inference.

2.1.1. Embedding-Based methods

Our knowledge graph completion (KGC) task aims to predict missing entities such that, given a relation r, we can infer whether $(h, r, t) \in G$ is true. Embedding-based methods form the backbone of KGC by learning vector representations of entities and relations and defining scoring functions that assess the plausibility of a candidate triple. Recently, researchers have developed increasingly sophisticated techniques to improve accuracy and capture richer relational semantics. These techniques contain translation-based, bilinear-based, and rotation-based approaches, each offering unique advantages in encoding relational patterns and improving reasoning capabilities.

Translation-based methods Generally, entities and relations are transformed into a multi-dimensional vector space, where mathematical operations are employed to capture relational semantics. TransE [Bordes et al., 2013a], a foundational model, embeds entities and relations into a shared vector space where relationships are modeled as translations. Their approach enables efficient knowledge inference but struggles with more complex relation types, such as 1-to-N, N-to-1, and N-to-N relations.

To address these limitations, TransH [Wang et al., 2014] extends TransE by introducing relation-specific hyperplanes that allow entities to be projected into a relation-dependent space before applying the translation operation [Sun et al., 2019]. This improves the ability to model complex relationships beyond simple 1-to-1 mappings.

Further refinements were introduced in TransR [Lin et al., 2015], which separates entity space and relation space, allowing entities to be projected into relation-specific vector spaces. This extra flexibility improves the understanding of specific relationships and makes it more accurate for different biomedical connections, such like how drugs interact with each other, and how diseases relate to genes.

Bilinear-based methods Unlike translation-based methods, bilinear models capture relational semantics using multiplicative interactions. In these models, relations act as bilinear transformations between entity embeddings.

DistMult [Yang et al., 2015] and ComplEx [Trouillon et al., 2016] interpret relations via similarity scores so as to capture the relational semantics. However, DistMult is limited to modeling only symmetric relations, which restricts its applicability in biomedical KGs where asymmetric interactions (e.g., drug-target or disease-gene interactions) are prevalent. ComplEx [Trouillon et al., 2016] extends DistMult by introducing complex-valued embeddings, enabling it to effectively handle a larger variety of binary relations including asymmetric relations.

Rotation-based methods Rotation-based models further refine the relational transformation process by representing relationships as rotations in complex space rather than translations. This enables more expressive reasoning over entity pairs.

To cover symmetry, asymmetry, inversion, and composition patterns, researchers Sun et al. achieve semantic transformation via a rotation operator on the complex plane. RotatE [Sun et al., 2019] defines relations as rotational transformations in the complex plane:

$$\mathbf{t} = \mathbf{h} \circ \mathbf{r} \tag{2.1}$$

where $\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{C}^k$ are the embeddings, and the module $|r_i| = 1$ ensures unit norm constraints for stability. The $\circ$ denotes the element-wise product. The model's scoring function $f_r(\mathrm{h}, \mathrm{t})$ is defined as:

$$f_r(\mathbf{h}, \mathbf{t}) = -\|\mathbf{h} \circ \mathbf{r} - \mathbf{t}\|^2 \tag{2.2}$$

Given a triplet (h, r, t), we expect that $\mathbf{t} = \mathbf{h} \circ \mathbf{r}$. Experimental results demonstrate that RotatE outperforms prior embedding methods on some widely used KGC benchmarks. Thus, it is a better choice in our experiment to model the enriched semantic information.

2.1.2. Advanced Neural Network-Based Methods

The above embedding-based approaches make full use of context-independent vectors from the KGs. They model triple plausibility with a well-designed score function and learn the entity and relation embeddings in a self-supervised manner [Zhang et al., 2024]. However, embedding-based methods treat each triple independently, lacking structural awareness within the global knowledge graph. Furthermore, their reliance on predefined scoring functions limits adaptability to complex relational patterns in real-world knowledge graphs, especially in some domains where relationships are often multi-hop and context-dependent.

To handle these problems, researchers have explored neural network-based methods, which leverage deep learning architectures to capture richer semantics and incorporate structural dependencies within KGs. By capturing textual knowledge in a base model, this type of method successfully generalizes traditional entity and relation embeddings to context-aware embeddings [Yao et al., 2019].

In recent years, researchers have integrated more powerful network structures to solve KGC tasks [Wei et al., 2023]. Our introductions fall into three categories:

- R-GCNs (Relational Graph Convolutional Networks) [Schlichtkrull et al., 2017] extend standard GNNs by incorporating relation-specific transformations into message passing. These models aggregate information from multi-hop neighbors, improving reasoning over long-range dependencies.

- ConvE (Convolutional KGE) [Dettmers et al., 2018] applies convolutional neural networks (CNNs) to knowledge graph embeddings. Their author Dettmers et al. shed light on building multi-layer models while maintaining high parameter efficiency. By transforming entity-relation pairs into 2D feature maps, ConvE captures complex interaction patterns with fewer parameters, achieving competitive performance compared to traditional embedding-based models.

- CompGCN (Composition-based Multi-Relational Graph Convolutional Networks) Paper [Vashishth et al., 2020] further enhance graph convolutional networks by incorporating composition-based transformations, enabling more effective modeling of complex entity–relation interactions.

As a specific subset of neural network-based models, a great variety of Transformer-based methods were devised to cope with KGC tasks. Transformer-based models, particularly large language models (LLMs), have recently revolutionized knowledge graph reasoning by leveraging pre-trained linguistic knowledge to enhance entity and relation representations. KG-BERT [Yao et al., 2019] fine-tunes BERT [Devlin et al., 2019] for knowledge graph completion by treating triples as natural language sentences, which take the description of entities and relations as the input text for fine-tuning. Succeeding by the BERT model, multiple text-based models have shown great success in learning contextualized word embeddings and achieving state-of-the-art performance in many language understanding tasks.

To bridge the gap between structured reasoning (GNNs) and text-based inference, recent research has introduced subgraph-based and hybrid techniques, which combine explicit relational reasoning with neural inference models.

A more recent mining tool, Generative Subgraph-based KGC (GS-KGC) [Yang et al., 2025], fulfills a more substantial increase with their evaluation metrics as compared to the ordinary LLM-based model.

In the paper, Yang et al. discuss a subgraph partitioning algorithm to generate negatives and neighbors. On the one hand, it is advantageous to support LLMs to generate various answers by feeding the model with negative samples. On the other hand, taking into account the neighbors provides additional surrounding information for LLM inferring.

At the same time, Zheng et al. propose KG-CF (Knowledge Graph - Context Filtering), which is a principled framework tailored for ranking-based KGC tasks. With the support of LLMs' inference abilities, they train the sequence classifier to filter out irrelevant contexts and achieve superior results on real-world datasets for KGC tasks [Zheng et al., 2025]. However, they are hardly able to derive understandable symbolic rules that explicitly capture relational patterns. These methods serve as a subclass of deep learning techniques but have difficulty providing human-explainable reasons for their predictions.

2.1.3. Rule-Based methods

A complementary direction of KGC methods is to integrate useful messages beyond single triples, explicitly including neighbor-based (entity-relation pairs surrounding with the entity) and path-based information (a list of sample paths linking the head entity and the tail entity). Given a KG, learning logical rules from existing triples pertains to both neighbor-based and path-based information. By enumerating grounding relation paths, a great variety of logic rules extraction methods are designed for KGC and other applications where rules are beneficial. Researchers Nickel et al. show that models such as TransE [Bordes et al., 2013a] and RESCAL [Nickel et al., 2011] have problems in solving certain types of completion tasks that can be solved by a rule-based approach with high precision.

Most traditional methods, such as path ordering [Lao and Cohen, 2010] and Markov Logic Networks [Richardson and Domingos, 2006], capture multi-hop relational paths by randomly traveling through the graph, enumerate the relational paths on the graph as candidate logic rules, and then learn the weights of each rule as an assessment of the quality of the rules. Currently, researchers have developed several advanced rule mining and reasoning systems for KGC. The following two main reasoning mechanisms are included:

- **Hard Logic Rule Reasoning:** Inferring the new facts by forward chaining is commonly derived using rule-based inference, often in the form of $Horn\ Clauses$[1], namely a logical equation of a particular rule-likeformat. [Wikipedia, 2024]. Hard logic rules explicitly represent the statistical regularities and dependencies encoded in the KG. Methods like AMIE+ [Galárraga et al., 2015a], RuleN [Meilicke et al., 2018], and AnyBURL [Meilicke et al., 2019] are rooted in the symbolic space. They learn such rules and apply forward chaining to iteratively infer missing links in given knowledge graphs. The following part will further investigate these systems.

- **Probabilistic Soft Logic (PSL):** For indeterministic reasoning, Markov Logic Networks (MLNs) are constructed so as to assign probabilistic weights to the rules, which indicate their confidence level in the KG. Researchers Bach et al. present some algorithms for inferring the most likely variable assignments and enable Hingeloss Markov Random Fields to model a large amount of structured data at levels not formerly available. In addition, the novel Neural Probabilistic Soft Logic integrates deep learning with PSL by joint neuro-symbolic inference, achieving upwards of 30% improvement over independent neural network models [Pryor et al., 2023].

Below, we simply introduce the above-mentioned Hard Logic Rule Reasoning Method along with other related techniques.

- **AMIE+:** The development process of $AMIE$ [Galárraga et al., 2013] involved the discovery of Horn Rules through the use of Association Rule Mining methods. As a facilitation of $AMIE$, AMIE+ [Galárraga et al., 2015a] move forward to take a series of pruning and query rewriting techniques. Galárraga et al. build this heuristic rule mining system under open-world assumption [2], and measure the reliability of rules by partial completeness assumption (PCA) confidence.

- **RuleN:** Followed by Galárraga et al., Meilicke et al. has developed their own rule-based system $RuleN$ based on $AMIE$. Their KGC tasks are focused on solving certain types of complementary

[1] `https://en.wikipedia.org/wiki/Horn_clause`

[2] `https://en.wikipedia.org/w/index.php?title=Open-world_assumption&redirect=no`

	Core Mechanism	Apply to Large KG	Apply to Sparse KG	Efficiency
AMIE+	Association-based	×	×	Lower
RuleN	Path-based	×	✓	Medium
AnyBURL	Statistical-based	✓	✓	Higher

Table 2.1: Comparison among existing rule-based systems (Efficiency here indicates Computation Efficiency).

tasks with a rule-based approach targeted at high precision, where embedding models such as $TransE$ [Bordes et al., 2013a] are problematic. Researchers Meilicke et al. constructed a model consisting of $RuleN$ and $AMIE$ in one line as the embedding-based method, also containing $TransE$ [Bordes et al., 2013a], $HolE$ [Nickel et al., 2015], and $RESCAL$ [Nickel et al., 2011] as a rule-based method. Combining these models using linear blending, the objectives for their research involved binding the strengths of each model at the relational level.

- **AnyBURL:** As the most prominent work among all rule-based mining systems, AnyBURL (Anytime Bottom Up Rule Learning) [Meilicke et al., 2019] developed a stop-in-time algorithm to extract rules from knowledge graphs, which has been progressively improved so far. Instead of traversing the entire knowledge graph, such as the AMIE and AMIE+ did, it searches high-quality strict rules through sampling and heuristic evaluation. AnyBURL iteratively samples random paths of length n from a KG **G**, where n is the number of edges and starts from n = 1. These paths are then assembled into ground-logical rules where the first left edge is treated as the head and the remaining as the body [Wu et al., 2022]. This approach is verified to be competitive with the current state-of-the-art method and outperforms other rule-based approaches [Galárraga et al., 2015b; Meilicke et al., 2018].

Table 2.1 explicitly compares the features of AMIE+, RuleN, and AnyBURL. The learning mechanisms for each method differ by their design thought. It is noteworthy that AnyBURL stands out from others because of its better applicability for large-scale KG with sparse data and higher computational efficiency. These are vital reasons why we select AnyBURL as the basis of the rule learning technique.

Rule-based approaches directly extract symbolic logical rules, making their predictions transparent and interpretable. Moreover, the rule-based KGC addresses the sparsity issue in knowledge graphs and enhances the quality of embedding by utilizing symbolic features. Rule-based approaches specifically transfer existing rules to new instances without requiring retraining when new entities appear. This kind of inductive reasoning for unseen entities allows KGC to be effectively applied in various real-world instances, including drug discovery and drug repurposing, where new entities are expected to accelerate the discovery of new drugs or therapeutic functionalities.

2.2. Neuro-Symbolic system

Apparently, embedding-based and rule-based approaches use different reasoning strategies, which lead to varying levels of effectiveness across different knowledge graph completion tasks. Compared with each other, embedding-based approaches excel at handling semantic-rich relations by learning latent representations in continuous vector spaces, while rule-based methods are particularly effective in discovering structural graph patterns and deriving understandable and interpretable inferences through symbolic logic.

This fundamental distinction highlights a promising research direction: neuro-symbolic integration. By combining the expressive power of neural embeddings with the transparency and logical structure of symbolic rule reasoning, hybrid neuro-symbolic models have the potential to leverage the strengths of both paradigms. Such integration not only improves predictive performance but also enhances interpretability, making it especially valuable for domains like biomedical knowledge graphs where explainability and reliability are critical. Several recent studies have attempted to bridge this dichotomy to generate models that facilitate interpretability and maintain competitive performance [DeLong et al., 2024].

Hybrid neuro-symbolic approaches have emerged as a promising direction for KGC. One of the pioneering mixed methods, embedding and rule-based reasoning, [An et al., 2020] is created to adopt relational contextual knowledge incorporated into the rules rather than solely the vector representation of the word in the embedding model. An et al. explores the relationship between two entities in a deeper context and demonstrates great performance on FB15k and WN18. Researchers Werner construct a Neuro-symbolic method allow the joint use of the symbolic nature of ontologies and the vector representation in knowledge graph embeddings [Werner, 2024]. Modeling analogical structures in multi-relational embeddings is another classical approach that achieves state-of-the-art results on two widely used benchmark datasets [Liu et al., 2017].

An up-to-date method, $Renn$ [Zong et al., 2024], is deeply devoted to improving temporal KGC through the embedding of rules. Researchers Zong et al. merge the embedding of temporal logic rules in deep neural networks and test the efficacy on temporal datasets. Another hybrid model, called $DegreEmbed$, also proved to outperform the previous advanced models with a clear margin, as well as produce high-quality rules with great interpretability. Moreover, Alam et al. turn into temporal knowledge graphs and provide an insight into neurosymbolic methods for dynamic (temporal or non-temporal) KG completion. These approaches have the potential to improve the resolution of knowledge graph completion tasks in terms of reliability, interpretability, and data efficiency [DeLong et al., 2024].

2.3. Interpretability in AI4Science

As artificial intelligence continues to play a transformative role in accelerating scientific discovery, interpretability has emerged as a fundamental requirement—particularly in disciplines where trustworthiness, transparency, reproducibility, and scientific hypothesis generation are fundamental [Loyola-Gonzalez, 2019]. Different from conventional black-box models whose decisions may be opaque, AI systems in scientific domains are expected to not only deliver high-accuracy predictions but also to provide clear, traceable, and domain-aligned explanations. This is especially critical in areas such as biomedicine, drug discovery, and molecular biology, where decisions informed by AI may guide experiments, clinical trials, or therapeutic strategies.

In the context of AI4Science, interpretability bridges the gap between complex machine learning outputs and human scientific reasoning. It allows researchers to validate predictions against prior knowledge, discover hidden mechanistic relationships, and derive new scientific insights.

Recent works have highlighted the importance of interpretable AI in biological and biomedical applications. For example, Preuer et al. explore interpretable models for drug discovery using graph neural networks, enabling domain experts to trace how molecular substructures contribute to predictions [Preuer et al., 2022]. Similarly, Wellawatte et al. demonstrat that explainable AI methods, such as chemical counterfactuals and descriptor-based explanations, not only clarify deep learning predictions but also offer valuable insights into structure–property relationships [Wellawatte et al., 2023].

Previous studies have primarily concentrated on post-hoc methods to interpret models, particularly graph neural networks. However, these methods often lack stability in their interpretations and may capture features that are only spuriously correlated with the target task [Miao et al., 2022]. Furthermore, interpretability has also been explored through attention mechanisms in transformer-based models. These mechanisms offer a degree of transparency in otherwise black-box architectures such as RNNs and GNNs. Specifically, attention weights—scalar values assigned to input tokens or nodes—can be interpreted as indicators of importance, thereby providing a straightforward avenue for interpretability [Tutek and Šnajder, 2022]. Building on this idea, Miao et al. proposed Graph Stochastic Attention (GSAT), which injects stochasticity into attention weights to improve robustness and interpretability. Their method demonstrated a 20% improvement in explanation AUC over state-of-the-art approaches, highlighting its effectiveness in generating more reliable and interpretable attention-based explanations.

In knowledge graph reasoning, rule-based inference provides a highly interpretable alternative to purely neural approaches. Li et al. propose a method named Entity–Attribute–Relation–Rule (EARR), which separates attributes from entities and uses logic rules to extend the dataset. Additionally, Zhang et al. provide a comprehensive survey on the evolution of symbolic, neural, and hybrid reasoning approaches over knowledge graphs. They focus on two key reasoning tasks — knowledge graph completion and question answering — and present a unified framework that explains these tasks within a cohesive reasoning paradigm. Recent studies have demonstrated that combining rule-based reasoning with embedding-based models can further enhance performance while maintaining interpretability. Tools like AnyBURL and SAFRAN exemplify this direction by mining interpretable logical rules that are automatically applicable across large-scale scientific knowledge graphs.

Motivated by prior studies, we contribute to this line of research by seamlessly attaching rule-based mining with the embedding techniques. Placing analogical context over rule discovery, we leverage pre-trained KG embeddings to guide rule application so as to provide reliable recommendations for completing missing facts. Our approach allows users to trace back predictions supported by multi-hop evidence chains.

3
Method and Experimental Setup

The key idea behind our project is to predict potential entities by exploiting logical rules within an analogy context. Discovering missing nodes from the subgraph pattern shared between other node pairs with the same type of nodes and relations.

In the first part, we describe in detail the overall structure of the study and each of the steps involved; in the second part, we show the specific configuration and setup of the experimental process system and model, with the purpose of generating reproducible and reliable predictive suggestions accordingly.

3.1. Our method

Our approach leverages global structural patterns and locally analogical information to enhance KGC. In other words, combining generalized rule reasoning with context awareness is the key thought behind our method. This involves the following two views:

- **Global structural patterns:** refer to high-level relational structures captured by rules that emerge across the knowledge graph. By analyzing large numbers of paths—sequences of connected entities and relations—these patterns are generalized into formalized rules. Although rule learning relies on extensive training data, it allows for inference even in sparsely connected areas of the graph where direct evidence is absent.

- **Local proximity information:** captures the proximity between triples by identifying entities that are closely embedded in the vector space. By grouping such entities, our approach enriches the inference process with more contextually relevant paths.

By integrating two views, our research process in this paper is divided into three steps: Analogy Mapping, Rule Mining, and Analogy Reasoning.

Figure 3.1 shows the overall process of predicting missing entities for any query established from the testing triple dataset. First, in the Analogy Mapping stage, we establish mappings from each testing triple (h, r, t) to the analogy triple set defined (H, r, T) in the subgraph. The core of this step is to identify triples with similar semantic and structural information so as to provide the context for subsequent rule mining and reasoning.

Second, in the rule mining phase, we employ the AnyBURL-based rule mining algorithm to extract potential rules from the knowledge graph over all identified analogy sets. The red arrows highlight three triangular structures that share a common multi-hop path pattern. From these recurring patterns, a rule is summarized using a strict logical form, capturing the consistent relational structure observed across different entity pairs within the analogy set. The rules can not only reveal the implicit relationships in the subgraph but also provide guidance for the subsequent reasoning process.

Finally, in the analogy reasoning phase, we predict the missing entities by performing strict logical reasoning on the query $(h, r, ?)$ or $(?, r, t)$ given the learned rules. In this way, we are able to achieve efficient KGC without relying on a large amount of labeled data. Since the triples retrieved from the analogy set have already excluded irrelevant paths after Analogy Mapping, the triples used to build the sample path during rule mining were reduced by 75.77% and 55.52% for $Fb15k-237$ and $DRKG$,

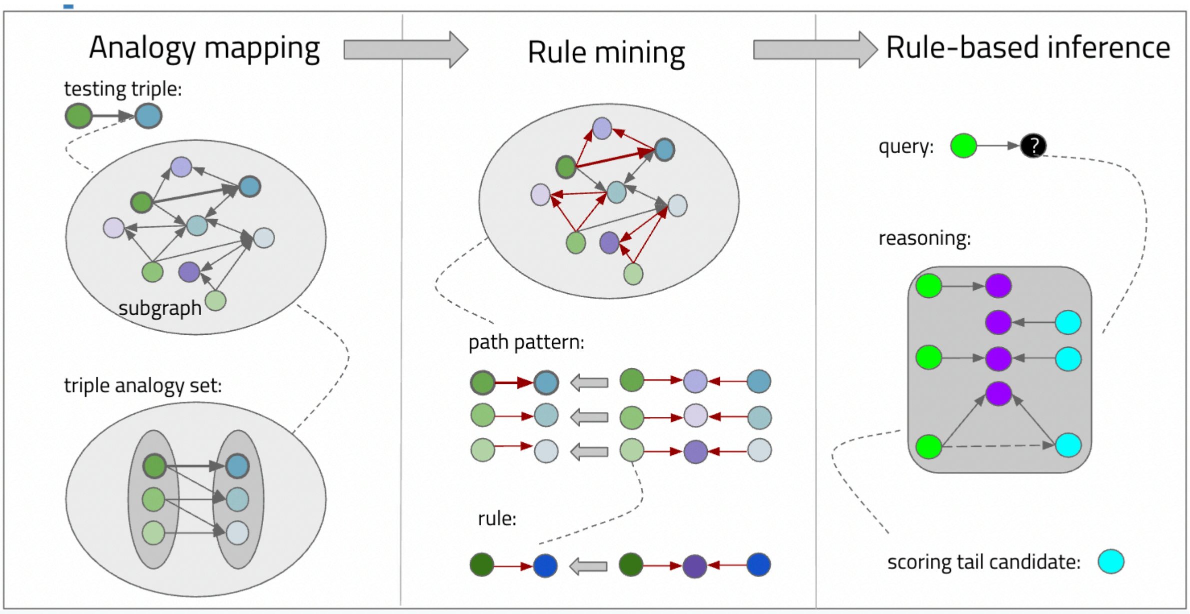


Figure 3.1: The flow of analogy-based rule mining for an individual test triple (each node represents a specific entity, and similarly colored nodes indicate close proximity in the embedding space).

respectively. To score the predicted candidates, we adopt an efficient ranking strategy by assigning each candidate the highest confidence score among all rules that generated it. In cases where multiple candidates share the same maximum score, we break the tie by comparing the second-highest confidence rule for each candidate, continuing this process iteratively until a difference is found.

Given a random triple (h_1, r_1, t_1), our goal is to identify a subgraph that captures structurally and semantically related head or tail entities. This process maps an individual triple to a broader analogy-based subgraph. As illustrated in Figure 3.2, which begins from the seed triple (h_1, r_1, t_1), different colors denote different entity types. To enable analogy-based mapping, we cluster entities into two groups: a head analogy set H and a tail analogy set T. These sets are constructed by computing proximity between entity embeddings in the knowledge graph, identifying entities that are close in the embedding space. As a result, we derive an analogy set: $\{(h_1, r_1, t_1),\ (h_3, r_1, t_1),\ (h_3, r_1, t_3)\}$, which contains triples that are analogous to the original (h_1, r_1, t_1) and share similar structural and semantic contexts. It is worth noting that although (h_2, r_1, t_2) shares the same relation r_1 with them, it is excluded from the analogy set due to the larger embedding distance, which reflects weaker proximity in complex vector space. Additionally, this exclusion may be attributed to the presence of fewer analogous structural patterns surrounding (h_2, r_1, t_2) compared to those included in the analogy set. The resulting analogy set provides the contextual foundation for subsequent multi-hop path discovery and rule mining, as discussed in the following subsection.

We build the above analogy mapping based on capturing the textual embeddings of triples. Considering the difficulty of embedding multi-relational data in vector spaces, we train the domain-specific rotate embedding model as the foundation model on different KGs. The training implementation relies on the latest stable version of the PyKEEN[1] package, which provides a high-level entry into the extensible functionality [Ali et al., 2021]. We move on from the pipeline function and use the Stochastic Local Closed World Assumption (SLCWA) to pick a subset of negative triples at random instead of assuming that all unobserved triples are wrong. Contrastive learning is thus built to assign higher scores for positive triples. The score for a triple (h, r, t) in our Rotate model is calculated as:

$$Score(h, r, t) = -\|e_h \circ e_r - e_t\| \tag{3.1}$$

where $h, r, t \in \mathbb{C}^d$ are the complex embeddings of the head entity h, relation r, and tail entity t. Rotating h by the phase of r, $h \circ r$ computed according to the element-wise product. The above scoring function is designed to minimize the distance between the transformed head entity $h \circ r$ and the tail entity t

[1] https://github.com/pykeen/pykeen

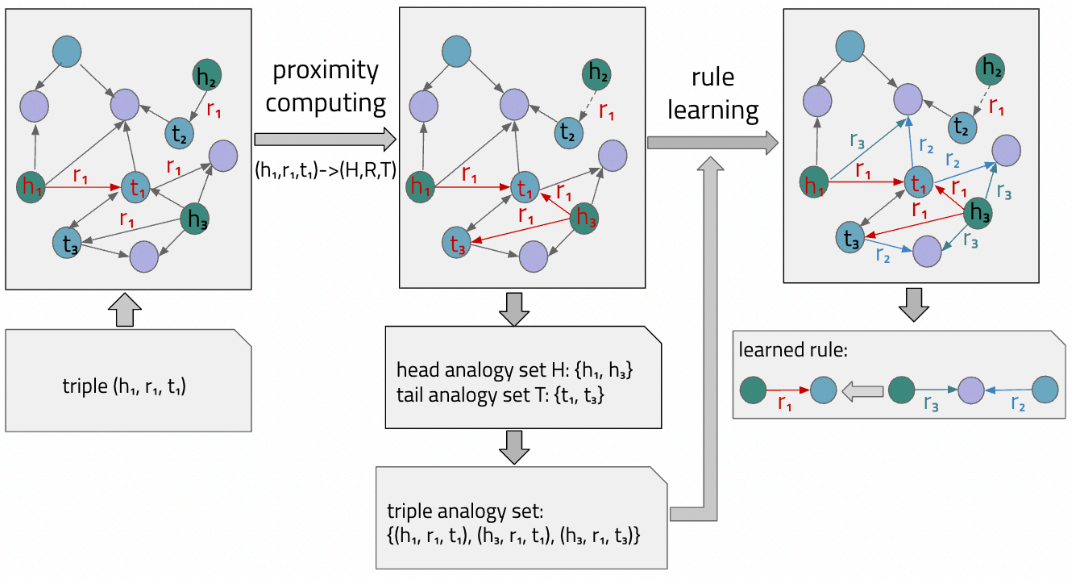


Figure 3.2: A specific inference case, given a triple (h_1, r_1, t_1) and its subgraph. (Each node and edge color represents a distinct entity and relation type. the dotted arrow connecting h_2 and t_2 indicates it is excluded from the analogy set due to a much more distant embedding)

in the embedding space. A smaller score indicates a closer distance, as well as a stronger semantic connection, making the triple more plausible.

The rotate embedding model trained on the given data builds a numerical representation of entities and relations. To search for the set of entities that are similar to a given entity, we evaluate the similarity among a pair of entities by computing their Cosine Similarity Coefficient:

$$Sim(h_1, h_2) = \frac{e_{h1} \cdot e_{h2}}{\|e_{h1}\| \cdot \|e_{h2}\|} \tag{3.2}$$

here e_{h1} and e_{h2} represent the rotate embedding of the two entities. We take the top 5 most similar entities related to h and r separately to build the analogy set in terms of each single triple from the testing dataset. However, some entities or relations of the testing triples verified a lack of sufficiently similar entities or relations, resulting in the testing triples failing to construct a valid analogy set. In such cases, we exclude these triples from the analogical inference process to ensure the reliability of rule mining and inference and simply use the original rotate embedding model to predict the results.

3.1.1. Rule Mining

The main purpose is to identify logical rules that indicate cyclic patterns within the context of triples. Our rule mining approach was facilitated based on ANYBURL (Anytime Bottom Up Rule Learning) [2] but converted the scope into an analogical context containing the proximity-related knowledge.

With respect to the definition for the rule in KG, we take into account the cyclic rule and serve $h(Y, X) \leftarrow b_1(X, A_1), ..., b_n(A_n, Y)$ as a cyclic rule. Here, X and Y represent variables that appear in the head, while A_i is a variable that appears in the body. The head of the rule is h(...) and b1(...) through bn(...) is its rule body [Meilicke et al., 2019]. From a KG G, we iteratively sample random paths of length n, where n is the number of edges. These paths are then assembled into ground logical rules where the first left triple is considered as the rule head and the remaining as the rule body [Wu et al., 2022].

A small subset of a knowledge graph $DRKG$ is shown in Figure 3.3 . There is a sequence of relationships starting with the head entity $Compound :: DB01029$ and forming a closed cycle involving three other entities. Here we are interested in finding rules that explain why the compound $Irbesartan$ causes side effects $abnormal\ feces$, which corresponds to the fact $Hetionet :: CcSE\ Effect(Irbesartan, abnormal\ feces)$. We formatted the graph by writing the following sample path:

[2] https://web.informatik.uni-mannheim.de/AnyBURL/

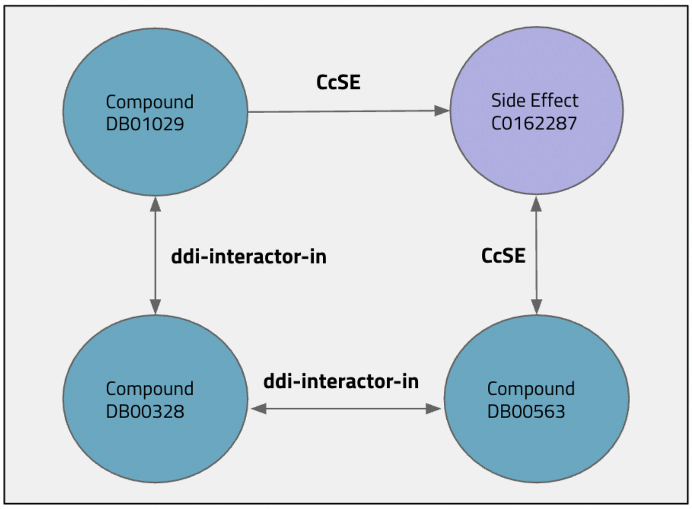


Figure 3.3: A cyclic path from DRKG. (We marked the Compound as blue , and Side Effect as purple.)

```
Hetionet::CcSE(Compound::DB01029,Side Effect::C0162287) ←
DRUGBANK::ddi-interactor-in(Compound::DB01029,Compound::DB00328),
DRUGBANK::ddi-interactor-in(Compound::DB00328,Compound::DB00563),
Hetionet::CcSE(Compound::DB00563,Side Effect::C0162287)
```

where the left-pointed arrow indicates the inference from a sequence of triples to a target triple based on a chain of biomedical interactions. In this case, we deduce that $compound :: DB01029$ is associated with the $Side\ Effect :: C0162287$ by chaining two $ddi - interactor - in$ (drug-drug-interaction) links and a $CcSE$ (compound cause side effect) link. This example illustrates a typical multi-hop reasoning path used in rule mining, where indirect relationships are combined to infer new, plausible links within the knowledge graph. To extract meaningful rules, we extract all paths with the same type of entity and relation. The concluded rule from those paths with length 3 looks like:

```
Hetionet::CcSE::Compound:Side Effect(X,Y) ←
DRUGBANK::ddi-interactor-in::Compound:Compound(X,A),
DRUGBANK::ddi-interactor-in::Compound:Compound(A,B),
Hetionet::CcSE::Compound:Side Effect(B,Y)
```

Referring back to Figure 3.2, we previously noted that the triples in the analogy set are used to enhance contextual connections. During the rule mining stage, the search is conducted from the triples in the analogy set $\{(h_1, r_1, t_1),\ (h_3, r_1, t_1),\ (h_3, r_1, t_3)\}$. As indicated by the red arrows, we observe that all three paths share a common structural pattern: the relation r_1 between a given type of head and tail entity can consistently be inferred through an intermediate entity (represented by the purple node) that serves as a connector. This recurring motif reflects a relational dependency that provides the foundation for rule formulation.

The bottom right corner of Figure 3.2 illustrates this learned rule using a notation that follows the standard relational structure of a knowledge graph. The grey left-pointing arrow indicates that the rule head on the left can be logically inferred from the rule body on the right. In essence, the rule generalizes the frequently observed path structure: if an intermediate entity of a specific type (purple node) connects the head entity (green) and tail entity (blue), then the direct relation r_1 between them is likely to hold.

Following the rule mining process, it is important to quantify each rule's reliability by assigning a suitable confidence score. The widely used standard confidence assesses the accuracy of a rule by computing the ratio of correct predictions to the total number of predictions generated by that rule. In addition to standard confidence, several alternative scoring metrics have been proposed to better rank and select rules for Knowledge Graph Completion (KGC), such as PCA confidence [Khajeh Nassiri et al., 2023; Betz et al., 2023] and Laplace-Smoothed Confidence [Kikuchi et al., 2015]. In our

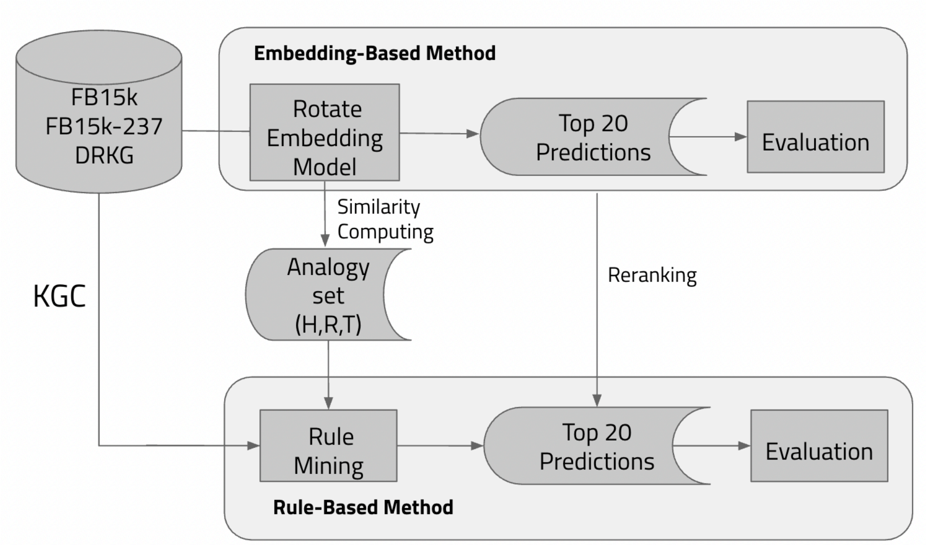


Figure 3.4: The mechanism behind analogy-based rule mining for any single query.

implementation, we follow the standard definition used by Meilicke et al., with a slight enhancement through additive (Laplace) smoothing to improve robustness, especially in cases where rules generate few predictions.

After deriving logical rules from the corresponding grounding paths and assigning confidence scores, we proceed with the reasoning process to predict missing entities.

3.1.2. Analogy Reasoning

At the third step, we apply predefined logical rules to infer new entities or validate existing ones based on the current knowledge graph data.

The relationship between the number of queries and the number of associated rules is typically one-to-many (1–N), whereas the relationship between the rules and the candidate predictions they generate is many-to-many (N–N). Specifically, given a query $(h, r, ?)$, a rule can generate one tail candidate, multiple tail candidates, or no tail candidates; one tail candidate can be inferred from one or more rules.

There are a variety of ways to aggregate the results generated by the rules. As a basis, we choose the most robust candidates scoring method. We define the final score of an entity as the maximum of the confidence scores of all the rules that generated it. If the maximum score of several candidates is the same, we order these candidates via the second-best rule that generates them, and so on, until we find a rule that makes a difference [Meilicke et al., 2019].

As shown in Figure 3.4 , we perform a study utilizing both the traditional embedding-based model and the rule mining model on FB15k, FB15k-237, and DRKG. The FB15k is taken as a supplementary knowledge base to analyze how the type of inverse relationships affects the model performance.

Our method reranking the RotatE keeps the original predictions when the analogy context seldom provides useful neighbor information, especially when the analogy set is empty. The analogy set highlights the top 5 distance-closed entities for both head and tail entities in the testing dataset. We then stand out the paths that existed from the head entity and tail entity in KG. The analogy reasoning in our method takes the first 20 entities to rerank the candidates based on the predicted RotatE model.

Although our architecture requires constructing two distinct computational modules—one for the RotatE embedding model and another for the rule-based model—the inference and evaluation phases can be executed in parallel to preserve computational efficiency. RotatE training relies heavily on large-scale GPU-based matrix operations, while rule-based inference benefits from multi-threaded CPU environments optimized for parallel logical reasoning. However, our computational pipeline introduces a crucial dependency: the rule search space is guided by the semantic structure formed from the pre-trained RotatE embeddings. This dependency restricts the parallel execution of certain stages, particularly during the model generation phase, where rule extraction must wait for the completion

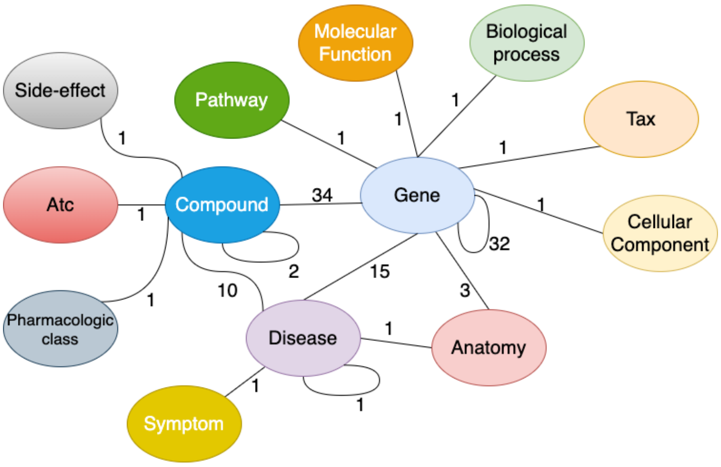


Figure 3.5: Entity Interactions in the DRKG. The number over an edge indicates the number of relation types for that entity pair in DRKG. Reproduced from [Ioannidis et al., 2020]

.

of embedding training. As a result, while prediction and evaluation can proceed concurrently, earlier stages of the pipeline remain sequential due to this inherent interdependence between the two underlying models.

3.2. Experimental Settings

In this section, we first display the KGs employed to justify the functionality of the analogy-based rule-mining method. Next, we list the detailed information with regard to model configurations.

3.2.1. Datasets

Typically, we treat FB15k and FB15k-237 as benchmarks for knowledge base completion, encompassing knowledge base relation triples and textual references to Freebase entity pairs. FB15k [Bordes et al., 2013b] contains many inverse relations, such as

```
(Actor A , acted in, Film B)
(Film B, has actor, Actor A)
```

where $acted\ in$ and $has\ actor$ have exactly opposite head and tail entities.

Extracting from the FB15k [Bordes et al., 2013b], it filters out triples that hold the inverse relation to avoid leakage from the training to testing and validation splits. FB15k-237 provides a challenging and realistic benchmark for evaluating the performance of various models in many tasks [**?**Ren et al., 2020; Wang et al., 2022; Bi et al., 2024].

However, FB15k-237 is fairly small, and the goal of our work is to extend symbolic rule learning to larger knowledge graphs to make AnyBURL applicable to more realistic scenarios. As a widely used biological knowledge graph, DRKG includes comprehensive knowledge from some well-known medical databases, including DrugBank, Hetionet, GNBR, String, IntAct and DGIdb, and updated data collected from publications particularly related to Covid19 [Ioannidis et al., 2020]. The Figure 3.5 shows the interactions among 13 entity-types and 107 edge-types. We adopt the splited DRKG file from PYKEEN, containing the initial DRKG in the format of (h, r, t) triples.

As depicted by Table 3.1, the basic features with respect to entities, relations, and their sparsity are given to compare among FB15k, FB15k-237, and DRKG. Since FB15k-237 removes many redundant inverse relations, it shrinks the total number of relations from 1,345 to 237. Negatively correlated, the average triples per relation on FB15k-237 is higher than on FB15k. FB15k-237 has a much lower average degree (21.4), indicating that entities are more connected.

Notably, DRKG probably possesses nearly 19 times the data volume of FB15k and FB15k-237, whereas the number of relation types is less than half of that of FB15k-237 and significantly lower than FB15k. Thus, DRKG exhibits denser connectivity among their nodes in terms of almost three times the average degree of FB15k-237.

	FB15k	FB15k-237	DRKG
domain	daily	daily	biomedical
purpose	benchmark for KGC	benchmark for KGC	biomedical reasoning
total entities	14,951	14,514	97,238
type of relations	1,345	237	107
total triples	592,213	310,116	5.8 million
avg degree	39.6	21.4	59.8
avg triples per relation	440	1,174	58,000

Table 3.1: Details concerning the KGs used for the experiments.

Also, there are relational format distinctions between FB15k-237 and DRKG. Relations from FB15k and FB15k-237 follow a hierarchical path-based format resembling RDF triples. For example, relation $/film/film/directed\_by$ consist of a domain $/film/film/$ and an actual relation type $directed\_by$, which are structured in a taxonomy-based manner. Their hierarchical grouping is split by slashes (/). DRKG uses a prefix-based format, typically with :: as separators. For $DRUGBANK :: ddi{-}interactor{-}in :: Compound : Compound$, prefix $DRUGBANK$ indicate the source database; $ddi - interactor - in$ specify the type of interaction; and $Compound : Compound$ define the subject and object entity types at the end.

3.2.2. Search Strategies

In this section, we will discuss traditional algorithms for obtaining the possible grounding path, as well as feasible search strategies. The first stage of rule mining involves extracting the logic rules from these grounding paths.

Our research aims to find potential rules within the aforementioned semantical analogy set. Thus, the triples in our analogy set are rule heads for path exploring. Given the rule head, the search for candidate logic rules is simplified by finding plausible paths for rule bodies [Wang et al., 2023]. Specifically,

- Each fact triple in the generated analogy set (e.g., (h_1,r,t_1)) is first taken as the rule head, which specify the starting nodes h_1 and ending nodes t_1 of the candidate grounding path.
- With respect to the denoted the max rule length, we abstract all grounding paths sequentially by random-walk strategy from the overall KG. From the head entity to the tail entity of the rule head, the path select neighbouring nodes in a stochastic direction and walk with a concrete length (e.g., (h_1,r_1,e_1) ^(e_1,r_2,e_2) ^(e_2,r_3,t_1))
- In this case, the candidate logic rule r:

  $$r(X,Y) \leftarrow r_1(X,A) \hat{}\, r_2(A,B) \hat{}\, r_3(B,Y) \tag{3.3}$$

  is concluded from the given KG.
- To eliminate low-quality rules, we conduct a further filtering process based on the given confidence threshold.

This naive algorithm enumerates all possible combinations of conjunctions of atoms. However, to maximize the number of rules in the learning process, it is crucial to balance between small-size KG and large-scale KG. If we set a relatively higher threshold, the algorithm will spend a lot of time exploring the paths to produce frequently appearing rules, while if it is relatively lower, crucial rules may be overlooked. There is a crucial trade-off between the expressiveness of the mined rules and the speed of the mining process.

Meilicke et al. introduce an approach that uses reinforcement learning in ANYBURL, which adopting the support, confidence, and length for the rules according to a reward strategy R_s.

$$R_{s\times c/2^l}(\mathbb{S}) = \sum_{r\in\mathbb{S}} \frac{\text{support}(r) \times \text{confidence}(r)}{2^{length(r)}} \tag{3.4}$$

where S refers to the analogy set of rules, $length(r)$ is the rule length of a rule r. Support is defined as the probability that a rule is observed in the KG, namely, the frequency with which a rule holds. Confidence measures the reliability of a rule, i.e., the probability that when the premises hold, the conclusion also holds.

For Rule 3.3, confidence is defined as:

$$\text{confidence}(r) = \frac{\text{support}(r)}{\text{support}(rule\ head)} \tag{3.5}$$

This reward strategy for reinforcement learning is based on the following three perceptions:

- **Support:** the quality of a rule set can be measured by the number of triples it aligns to the knowledge graph.
- **Confidence:** rules that make correct predictions with high confidence will be more desirable because they are inclined to produce top-ranked candidates.
- **Rule Length:** shorter rules usually represent a wider range of relationships and cover more combinations of entities with a much faster performance. In other words, rules with more atoms can capture more complicated correlations, but come with a larger search space.

This scoring function rewards rules with high support and high confidence while penalizing longer rules (decaying by parameter v). This favors rules that are common (high support), reliable (high confidence), and short (short rule length).

3.2.3. Model Configuration

In this section, we demonstrate how the embedding model and rule-based model are constructed, as our approach integrates them to gather high-level information for KGC tasks. Moreover, we simply state the parameter-adjusting insights that occurred in our experiment.

Rotate embedding model. For the rotate embedding model, we perform the training loop on a remote GPU server equipped with multicore and large memory. It helps to speed up the embedding generation in high-dimensional vector space and make sure applying for the large-scale DRKG is safe and reliable.

Embedding-based models have hyperparameters that need to be optimized on a validation dataset. We selectively take the experimental artifacts as reference from the previous reproducibility study in $PyKEEN$. For better model efficacy, reimplementing the Rotate Embedding Model based on $FB15k$ and $FB15k237$. With training in the biomedication domain, we slightly refine the best-reported hyperparameters to adapt to DRKG and provide the specific numerical value for parameters, as discussed in Appendix A.1.

Analogy-based rule mining model. Apart from the memory requirements of the large-scale knowledge graph, the complicated thread control for rule extraction is also applied to speed up the graph pattern recognition within the analogy context. In our setting, mining rules for given head and tail entities will trigger a designated number of threads to explore rules with different lengths simultaneously. However, the employment of multithreading also introduces the additional overhead of context switching. Besides, when a set of threads was appointed for different subtasks, the resource consumption appeared obvious, given that the system kept waiting for active threads to finish the job.

Specifically, we wrap up the repetitive mining task and submit it to a thread pool, which was created in advance via the $ThreadPoolExecutor$ class in $java$. This flexible framework for managing a pool of worker threads supports both getting the execution result of a task and managing the run-time status of a task. At the same time, we introduce cooperative scheduling to ensure system stability because our design, to some extent, relies on threads actively releasing CPU resources, which may lead to certain threads occupying the CPU for a long time. Our method will first change the state of the thread pool as $SHUTDOWN$ and then interrupt vacant threads, allowing us to terminate rule learning at any point.

Also, we refined our analogy-based rule mining method based on the rule learner ANYBURL. By refining the code, we launch the rule abstraction focused on the local sample path, which is semantically aggregated by calculating cosine similarity instead of covering the full KG. In this case, during the learning process, only the triple with the proximity context will be able to learn via generalization

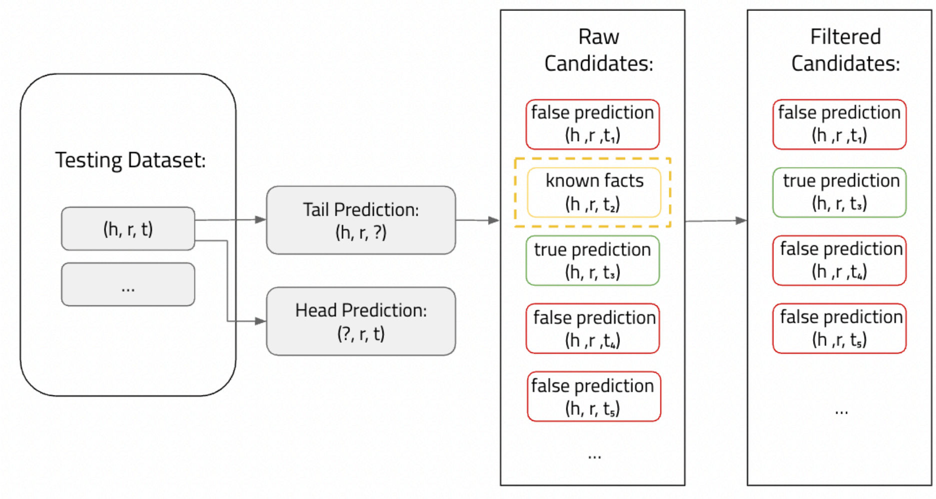

Testing Dataset:
(h, r, t)
...
Tail Prediction:
(h, r, ?)
Head Prediction:
(?, r, t)
Raw Candidates:
false prediction (h ,r ,t₁)
known facts (h ,r, t₂)
true prediction (h, r, t₃)
false prediction (h ,r ,t₄)
false prediction (h, r, t₅)
...
Filtered Candidates:
false prediction (h ,r ,t₁)
true prediction (h, r, t₃)
false prediction (h ,r ,t₄)
false prediction (h ,r, t₅)
...

Figure 3.6: The filtered ranking strategy by removing known facts (h, r, t_2). (the red box denotes a false answer, the yellow box indicates a known fact from either the training or validation dataset, and the green box represents the true answer to the query.)

techniques. We adjust the rule length and confidence threshold to weight against model performance as shown in Appendix A.2.

Rule-based systems also have hyperparameters. However, in our experiments, we found them easy to set for knowledge base completion even without a validation dataset. As these hyperparameters are typically a mechanism to tune running time versus completeness of the rule learning process, we simply modified the most expressive setting in the configuration file that describes the input and output of rule learning, including:

- $SNAPSHOTS\_AT$ The time used for learning the rules was restricted to 1000 seconds (17 minutes) in most of the runs [Meilicke et al., 2019]. The benefits of AnyBURL-based alorithms allow us design the temporal list, such as [5, 100, 1000], to terminate the mining and save the rules at these points. However, our experiment targeted at analogy set for every testing triple. Aims at finishing within a reasonable time, we denote $SNAPSHOTS\_AT$ by 2 after the tradeoff between the time efficiency and quality of extracted rules.

- $WORKER\_THREADS$ As we mentioned, the number of worker threads differs from the laptop with different compute cores. To take full advantage of computational efficiency, we simply set the core number of our server for $WORKER\_THREADS$.

- MAX_LENGTH_CYCLIC We keep the original best setting in paper [Meilicke et al., 2019] for Fb15k and Fb15k237, namely $MAX\_LENGTH\_CYCLIC = 3$. For DRKG, it is possible to increase the length of the cyclic rules from three (default value) to five, which gives a small additional plus.

These hyperparameters directly affect the number of learned rules. The state-of-the-art rule-based mechanism yields very promising results with comparatively small sample sizes, resulting in shorter run times. Except for the modification of these three hyperparameters, we created the results using the default parameter setting from the original AnyBURL.

3.3. Filtered Ranking

Filtered ranking is a commonly used ranking strategy in KGC tasks to perform a fair evaluation. The standard evaluation approach involves ranking all possible candidate entities and measuring how well the correct entity is ranked. However, there is a situation in which some of these candidates may actually be correct facts but are still ranked lower [Bordes et al., 2013a].

As illustrated in Figure 3.6, the raw ranking method generates a sorted list of candidate entities in descending order for a given query $(h, r, ?)$. In this example, the top-ranked candidate is actually a false answer, while both (h, r, t_2) and (h, r, t_3) are true answers. However, the candidate triple (h, r, t_2) already exists in the given knowledge graph (either in the training or validation set) and is therefore treated as a known fact. Ideally, the truth t_2 should be ranked at the top. However, due to the inclusion of these already known facts in the candidate list, inferred true answers such as t_3 may be pushed lower in the ranking, which adversely affects the evaluation metrics and underestimates model performance.

To resolve this issue, we filter out candidates that already exist in the training or validation set before ranking, except for the test triple itself. Furthermore, it ensures that the ranking better reflects the model's ability to predict truly missing facts rather than just memorizing the dataset.

3.4. Evaluation Metrics for Ranking-based KGC

Our research was conducted to predict missing triples in three knowledge graphs by mining analogy-based rules. When applying rules to infer unknown entities, each candidate triple is associated with a score and sorted in a descending order.

If we take the i-th query $(h, r, ?)$, given a candidate tail entity t, its rank is the position in the list of all candidates. In our implementation, KGC algorithms are evaluated with a testing dataset in terms of the following metrics:

- **Mean Rank (MR):** measuring the average rank of the correct answer across all queries from testing triples.

  $$\text{MR} = \frac{1}{N}\sum_{i=1}^{N} \text{rank}_i \tag{3.6}$$

  where N is the number of queries, and rank_i is the rank of the correct answer for the i-th query. According to Equation 3.6, as the correct triples appear at a lower rank, MR decreases lower, along with a better performance.

- **Mean Reciprocal Rank (MRR):** measuring the average reciprocal rank of the correct answer across all queries from testing triples.

  $$\text{MRR} = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{\text{rank}_i} \tag{3.7}$$

  Compared with the MR, it takes the inverse of the rank at which the correct answer is found. Thus, a higher MRR indicates better model performance, which is inversely proportional to MR.

- **Hits@K:** computing the ratio of correct answers ranked in the top-k positions for predicted candidates.

  $$\text{MRR} = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{\text{rank}_i} \tag{3.8}$$

  We choose Hits@1, Hits@3, and Hits@10 to illustrate the model performance at the evaluation stage. Higher values of Hits@k indicate better performance, meaning the model can often predict the correct triple in the top-k ranked results.

For our rule-based KGC tasks, we evaluate the model's performance by a combination of ranking-based metrics, including MR, MRR, Hit@1, Hits@3 and Hits@k. Since we only retrieve the top 20 predicted results, a common issue arises when the true answer does not appear in the top 20 predictions. If the true answer is not sorted in the top 20, the undefined rank_i missing in the rank array will futher skew the evaluation. Aims at reasonable evaluation metrics, we assign a default rank beyond the cutoff to adjust all metrics.

Besides, Hit@1 and Hits@3 shallowly represent the true prediction efficacy in our tasks to some extent. The reason is that there are seldom 1-N and N-1 types of triples occured in FB15k and FB15k237; even none of them existed in DRKG. In this case, it is slightly expected to capture only one true answer for a specific query. We mainly take adjusted MRR and Hits@10 as inferences to review our experiment.

4

Results & Analysis

This chapter presents the final performance results of our foundational RotatE embedding model (RotatE) and the proposed Analogy-based Rule Mining (ARM) approach. We conducted the comparison based on the testing dataset among three KGs: FB15k, FB15k-237, and DRKG. Then, we demonstrate how analogy-based rule mining enhances the interpretability of predicted results through chain-based reasoning. Finally, we analyze some of the most influential features that lead to the worse performance by introducing some specific scenarios.

4.1. Results

Table 4.1: Performance Comparison of RotatE (Rotate Embedding Model) and ARM (Analogy-based Rule Mining Model) on FB15k, FB15k-237, and DRKG.

Dataset	Model	MR(%) ↓	MRR(%) ↑	Hits@1(%)	Hits@3(%)	Hits@10(%)
FB15k	RotatE	52.557	73.930	65.032	67.552	75.859
	ARM	40.730	78.580	66.785	75.971	81.585
FB15k-237	RotatE	147.209	28.926	19.155	31.524	48.483
	ARM	169.822	19.172	8.266	19.384	35.07
DRKG	RotatE	61.992	28.392	11.826	27.599	41.081
	ARM	57.260	31.701	14.481	34.544	48.683

4.1.1. Performance over FB15k

In FB15k, the Analogy-based Rule Mining model (40.730) achieves a lower MR than the RotatE model (52.557), indicating that the rule-based model ranks the correct triples closer to the top on average. The MRR of analogy-based rule mining (78.580) outperforms RotatE (73.930), suggesting our approach provides more accurate top-ranked predictions.

Compared with Hits@1, a significant improvement is seen in Hits@3, where the rule-based model (75.971) outperforms RotatE (67.552), indicating better ranking performance in the top 3 results. Furthermore, the most notable gain is in Hits@10, where the analogy-based rule mining model achieves 81.585, compared to 75.859 from RotatE. This suggests the rule-based model generates more relevant candidates within the top 10 predictions.

As Table 4.1 shows, the analogy-based rule mining model outperforms RotatE across all metrics on FB15k, demonstrating its effectiveness in structured datasets with rich semantic patterns.

4.1.2. Performance over FB15k-237

RotatE ranks the correct triples closer to the top more frequently since the RotatE model (147.209) has a lower MR than the rule-based model (169.822). Moreover, the RotatE model achieves higher MRR at 28.926, significantly outperforming the rule-based model, which arrived at 19.172.

One exception is the biggest drop for the analogy-based rule mining model occurs in Hits@1 (8.266 vs. 19.155 for RotatE). This suggests that the rule-based model still struggles with highly sparse and less predictable relationships in FB15k-237. The rule-based model lags behind RotatE for Hits@3 and Hits@10, showing weaker ranking capability within the top 3 and 10 positions with fewer correct predictions appearing.

Comparing with FB15k, the RotatE model performs significantly better than the rule-based model on FB15k-237. This means our approach dose not perform well when the KGC tasks are performed on FB15k-237. To some degree, this result is expected due to the specific way FB15k-237 was constructed. As we previously mentioned, FB15k-237 removed many relations widely existed in FB15k, the relation types significantlt decreased by more than 82%. Thus, FB15k-237 serves as a more challenging dataset for KGC even in the original Rotate Embedding ModelSun et al. [2019], where the MRR only arrives at 0.338. Rule-based models with these filtered relations, making inference and explanation even harder[Meilicke et al., 2018].

4.1.3. Performance over DRKG

In terms of MR, the analogy-based rule mining model (57.260) decreases over RotatE (61.992), ranking correct triples closer to the top. Consistently, for MRR, the rule-based model (31.701) obviously outperforms RotatE (28.392), showing better ranking quality.

A major improvement is seen in Hits@1, where the rule-based model (14.481) outperforms RotatE (11.826). The results suggests that rule mining is more effective at ranking the correct entity in the first position. Notably, the rule-based model (48.683) surpasses RotatE (41.081) by around 7.6 % in Hits@10. This shows superior predictive power in identifying correct relationships than the Rotate Embedding Model.

4.1.4. Overall Performance

In conclusion, our analogy-based rule mining approach performs better on multihop-structured datasets (such as FB15k and DRKG) compared to the standalone RotatE embedding model. This improvement is reasonable and expected, as FB15k includes a large proportion of inverse and redundant relational structures, which naturally favor multi-step reasoning. Similarly, biomedical knowledge graphs like DRKG are governed by strict symbolic and biological patterns, where meaningful inferences often emerge from chained or hierarchical relationships.

In contrast, traditional embedding models perform better on challenging datasets like FB15k-237, where inverse relations are filtered to get rid of the direct data leakage.

Our results support the view that embedding-based and rule-based approaches perform well on different types of completion tasks and that it is beneficial to combine predictions from both types of models [Shen et al., 2022]. Also, the performance of the model highly depends on the characteristics of the dataset and is more appropriate for KGs that hold intrinsically strong structured patterns.

4.2. Interpretibility Analysis

After inferences made by the analogy-based rule mining, we explain predictions by retrieving the related path from the KG. In terms of candidate lists, our study allows interpretation given a specified query. We take a query to predict the tail entities as an example:

$$(TP53,\ INTACT :: ASSOCIATION :: Gene : Gene,\ ?) \tag{4.1}$$

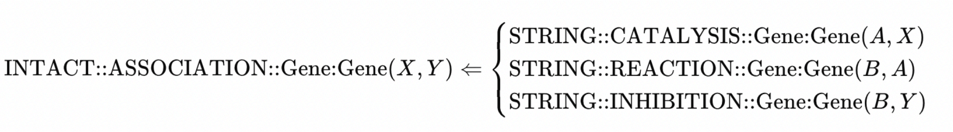
where the gene $TP53$ acts as a tumor suppressor, and the relation $INTACT :: ASSOCIATION :: Gene : Gene$ refers to the association between two genes. This query focuses on protein-protein interaction to predict which gene is likely to associated with $TP53$.

The experiment design allow us extract the corresponding logic chain to explain the tail candidates as shown in Table 4.2, where we list the top 3 predicted tail candidates based on their confidence and provide the corresponding biomedical interaction chains.

If we take the top candidate $CDKN1A$ as an example, its chain-based explanation illustrates how $TP53$ connects $CDKN1A$ through a series of biologically meaningful gene interactions: $TP53$ is catalyzed by $MDM2$, which interacts with $AKT1$, and in turn, $AKT1$ inhibits $CDKN1A$. This chain-level interpretability strengthens trust in our method, particularly in critical biomedical applications where reliability is essential. With the chain-based explanation, researchers can trace back the logical reasoning

	Explanation
CDKN1A	STRING::CATALYSIS::Gene:Gene(MDM2,TP53), STRING::REACTION::Gene:Gene(AKT1,MDM2), STRING::INHIBITION::Gene:Gene(AKT1,CDKN1A)
GADD45A	STRING::CATALYSIS::Gene:Gene(MDM2,TP53), STRING::REACTION::Gene:Gene(MAPK1,MDM2), STRING::INHIBITION::Gene:Gene(MAPK1,GADD45A)
RB1	STRING::CATALYSIS::Gene:Gene(HSP90AA1,TP53), STRING::REACTION::Gene:Gene(PIK3CA,HSP90AA1), STRING::INHIBITION::Gene:Gene(PIK3CA,RB1)

Table 4.2: Explanations for the top 3 tail candidates, given a mined rule in Figure 4.1.



$$\text{INTACT::ASSOCIATION::Gene:Gene}(X,Y) \Leftarrow \begin{cases}\text{STRING::CATALYSIS::Gene:Gene}(A,X)\\ \text{STRING::REACTION::Gene:Gene}(B,A)\\ \text{STRING::INHIBITION::Gene:Gene}(B,Y)\end{cases}$$

Figure 4.1: A learned rule on DRKG. (X, Y, A and B indicate variables of the entity that conform to the rule.)

steps, assess the biological plausibility of predictions, and gain deeper insights into the mechanistic foundations underlying the AI-generated associations.

These interaction chains are obtained during the rule mining phase and stored for subsequent analysis. If no applicable rule or valid prediction can be found within the knowledge graph, our approach returns a null result, indicating insufficient evidence for inference.

Additionally, explanation support is also available in the 2022 version of AnyBURL [Meilicke et al., 2023]. which provides a dedicated module for interpreting rule-based predictions. As a seperated component of the AnyBURL system, the explanation can be extend to any other knowledge graph completion technique with generated predictions. Taking a KG and its predictions as input, it interprets every prediction in the top-k ranking.

4.3. Failure Analysis

From Table 4.1, we conclude some reasons behind the different performance for the FB15k, FB15k-237, and DRKG, and give the analysis accordingly. The affected factors are categorized into the below three parts: data incompleteness, data imbalance, and less semantic understanding. The following content describes the actual query cases in our dataset encountered with worse evaluations.

4.3.1. Data incompleteness

Our model sometimes misinterprets missing entities as negative samples, leading to incorrect predictions. As is well known, KGs are often incomplete, meaning that they contain only a subset of the true relationships between entities; many valid facts are missing from the KG. Human curation biases bring about manually annotated or crowd-sourced data, which may miss crucial links. Moreover, data sparsity prevents the recording of a significant number of real-world relationships.

The lower average degree value is displayed in Table 2.1 on FB15k-237 as compared with the large-scale DRKG. Since our analogy-based rule mining approach relies on existing triples to generalize patterns, missing data can severely impact their performance. Also, applying the closed-world assumption (CWA) results in our model assuming that facts not explicitly stated in the KG are incorrect, which is not always true. This assumption leads to wrongly labeling missing but true triples as negative samples.

For example, in FB15k-237:

$$Query\ Case : (?, /music/genre/artists, /m/0137g1) \tag{4.2}$$

where we have a missing head entity represented by ?, a relation (/music/genre/artists) and a known entity (/m/0137g1, a machine ID in Freebase). Entity /m/0137g1 refers to Beck Hansen, an American

	Confidence	Applied Rule	Predictions
Rule1	0.394	/film/film_subject/films(X,Y) ← /film/film_subject/films(X,A), /film/director/film(B,A), /film/director/film(B,Y)	'/m/05489'
Rule2	0.341	/film/film_subject/films(X,Y) ← /film/film_subject/films(X,A), /award/award_nominee/award_nominations. /award/award_nomination/nominated_for(B,A), /film/film/produced_by(Y,B)	'/m/075fzd' '/m/06796'
Rule3	0.329	/film/film_subject/films(X,Y) ← /film/film_subject/films(X,A), /film/film/produced_by(A,B), /film/film/produced_by(Y,B)	'/m/075fzd' '/m/06796'

Table 4.3: Head candidates concerning the Query Case: (?, /film/film_subject/films, /m/02yvct).

musician born in 1970. This query aims at finding the artists Beck Hansen belong to which music genre. According to the logic rule:

```
/music/genre/artists(X,Y) ←
/music/genre/parent_genre(A,X),
/music/genre/artists(A,Y)
```

indicating that if an artist Y belongs to a subgenre A, and the genre A has a parent genre X, then Y should also be considered under the parent genre X.

The analogy-based rule mining model predicts missing head entities, returning candidate genres in descending order such as $/m/09nwwf$ (funk rock), $/m/01243b$ (indie rock), etc. Ideally, we expect the candidates contain to include folk music, since it is factually accurate that the musician Beck Hansen is associated with folk music—a genre classified as a subclass of contemporary folk music (as verified via Wikidata entry $Q11901$ [1] and $Q574651$ [2]) However, among all candidates, this grounding truth answer is not retrieved from the above rule. It turned out this relationship exists, but the potential fact is absent from the dataset FB15k-237. Unless explicitly defined within rules, our method does not inherently differentiate between hierarchical relationships (e.g., subclass-superclass relations) unless explicitly defined within rules.

In the analogy-based rule mining, missing facts not only prevent the identification of analogy sets but also distort frequent pattern extraction, leading to incomplete or biased rule generation. Especially if certain patterns are incomplete, rule learning can be biased toward observed triples while reducing generalization to valid but unobserved rules.

4.3.2. Data imbalance

In KGC, data imbalance refers to the situation where certain types of inner knowledge appear significantly more frequently than others in the training data. For example, in Drug Repurposing KG, the most common relations, like $DRUGBANK :: ddi-interactor-in :: Compound : Compound$ (drug-drug interactions), may be overrepresented, which is supported by 1099817 triples. While mutiple rare interactions (e.g., $DGIDB :: INHIBITOR :: Gene : Compound$, $DGIDB :: ACTIVATOR :: Gene : Compound$) may be underrepresented during rule-based inference. This long-tail problem affects our rule-based KGC in rule learning. To be specific, our sample-oriented rule mining algorithms tend to discover frequent structures in randomly sampled paths. Due to the predominance of highly appearing relations, the extracted rules widely reflect these common relations. However, rules involving rare relationships are less likely to be learned because of the lack of training samples.

The evaluation of rule quality in our research is based on standard confidence. Rules consisting of rare relations receive lower confidence scores because they appear less often, even if they are of

[1] https://www.wikidata.org/wiki/Q11901

[2] https://www.wikidata.org/wiki/Q574651

great practical importance. In these cases, the correct answer predicted avoidably appears in a later position within an ascending ranking array.

We take an example to explain the influence of the long-tail problem. Considering the below query case:

$$Query\ Case : (?, /film/film\_subject/films, /m/02yvct) \tag{4.3}$$

where we are looking forward to discovering the missing head entity (belonging to the film subject) associated with films $/m/02yvct$ (referring to Inglourious Basterds). As shown in Table 4.3, each line displays the confidence of the rule, the mined logic rule, and its inferring results. The data of Table 4.3 come from our experimental conduction on FB15k.

Understandably, the inference of a single query often involves different rules; also, one applied rule is able to generate at least one candidate. As mentioned before, we rank the candidates based on the maximum confidence of all the rules that have generated them.

The true answer for Query 4.3 is $/m/06796$ (referring to psychotherapy), which can be traced back from both Rule2 and Rule3. Nevertheless, our method assigns a higher confidence for Rule1 and places $/m/05489$ in the top position of its candidate list. This can be attributed to sub-structure binding relation $/film/film\_subject/film$ with $/film/director/film$ are slightly more frequent or reliable than with $/award/award\_nominee/award\_nominations./award/award\_nomination/nominated\_for$ and $/film/film/produced\_by$. This imbalanced distribution leads to degraded quality of learned Rule2 and Rule3, which further underestimates the score of predicted candidates.

4.3.3. Difficulty in semantic understanding

Compared to purely embedding-based models for KGC, our method leverages RotatE embeddings primarily to form local, semantically dense clusters. The core mechanism emphasizes capturing heuristic graph patterns, as suggested by the rule-based approach. However, this reliance on structural cues introduces certain limitations in modeling the intrinsic properties of entities and relations, which may affect overall performance in more complex KGC scenarios. To illustrate how this limited capacity to encode nuanced semantics impacts prediction, we examine the following query case:

$$(?, /award/award\_nominee/award\_nominations./award/award\_nomination/award, /m/01c92g) \tag{4.4}$$

where $/award/award\_nominee/award\_nominations$ represents a person or entity that has received award nominations and $/award/award\_nomination/award$ specifies the award for which the person was nominated. Thus, the query focuses on who has received a nomination for the given award $/m/01c92g$ (referring to $Grammy\ Award\ for\ Best\ Male\ Pop\ Vocal\ Performance$). According to our rule mining method, one applied rule is present:

```
/award/award_nominee/award_nominations./award/award_nomination/award(X,Y) ←
/people/person/sibling_s./people/sibling_relationship/sibling(A,X),
/award/award_nominee/award_nominations./award/award_nomination/award(A,Y)
```

which implicates if person A was nominated for award Y, and A has a sibling X, then X is also likely to be nominated for the same award.

From this rule, the tail entity $/m/0gbwp$ which represent for Janet Jackson, is reasoned accordingly. The insight behind the rule is that siblings in the entertainment industry often have similar talents and opportunities. In other words, many children growing up in a musical family are likely to win the same musical awards. In fact, the Jackson family[3], famous as a musical group, followed the success of Michael and Janet Jackson. The explanation for the predicted candidate Janet Jackson lay on the fact that Michael was nominated for an award $Grammy\ Award\ for\ Best\ Male\ Pop\ Vocal\ Performance$. Undoubtedly, this analogical relation should not be established between Michael and Janet given the award $/m/01c92g$, since it has already been specified to the male musician. It turned out our approach struggles with capturing these entity and relation properties.

Embeddings can capture similarities between entities, even when those similarities are not explicitly mentioned in the KG. This capability is achieved by learning dense vector representations that encode contextual meaning based on the entity's relationships in the graph. However, our method, relying on

[3] https://en.wikipedia.org/wiki/Jackson_family

local path clusters, is difficult in capturing complex contextual meanings. In this case, a rule-based system would require explicitly mined rules to specify the properties, which may not always be comprehensive.

All these challenges are widely prevalent in KGC, especially when dealing with sparse and imbalanced data distributions. Our approach enhances generalization by extending sparse triples with structurally similar analogical paths, allowing the model to learn knowledge from frequent to rare patterns. However, to further facilitate the reasoning process, it is crucial to incorporate external semantic resources and structural constraints. We further discuss the potential direction of future work in Chapter 5.2.

5 Conclusion and Future work

5.1. Conclusion

In this work, we develop a joint inference approach based on the rule mining and the embedding model. This architecture allows us to combine the strengths of both paradigms: the expressive generalization capability of neural embeddings and the structured, interpretable nature of symbolic reasoning.

On the basis of the well-performed rotate embedding model, we re-rank its predictions by incorporating rule mining over proximity-aggregated triples. It is verified that the analogy-based rule mining method yielded relatively better results in specific KGC benchmarks in terms of making accurate predictions. In contrast, standalone embedding models perform better on challenging datasets like FB15k-237, where inverse relations are filtered from FB15k. Moreover, its chain-based explanations enhance interpretability and provide valuable insights—particularly beneficial in the biomedical domain where understanding the reasoning process is crucial. After analyzing several real query cases that resulted in lower prediction accuracy, we categorize the issues into three main aspects and present corresponding improvement strategies to optimize our method. Importantly, our framework is modular and extensible, making it possible to integrate a wide range of neural and symbolic methods into a unified system, thereby supporting flexible adaptation to various reasoning tasks and domains.

5.2. Future work

During our experiment, an important limitation is that we try to combine multiple rules that predict the same triple and compute an overall confidence score for that prediction. It is inspired by the Noisy-Or model [Srinivas, 2013] from probabilistic reasoning, commonly used in Bayesian networks. However, it turned out worse performance than using the maximum confidence of rules. Meilicke et al. applied this strategy, but the inner dependent relationship between rules is not considered [Meilicke et al., 2019]. Thus, future work might focus on finding suitable ways of modeling the joint distribution over the rules. As a development of PSL, rules could be grouped according to syntactic similarity, distributions might be estimated from more advanced statistics such as pairwise confidences, or marginals could be approximated more rigorously as suggested from [Betz et al., 2023].

Another research direction falls into leveraging external ontologies and hierarchical structures (e.g., from the Unified Medical Language System, UMLS) to enable hierarchical reasoning. As we mentioned in Chapter 4.3.1, the analogy-based rule mining approach lacks the inherent capability to recognize and leverage ontological relationships, such as subclass-superclass hierarchies and part-whole structures. Incorporating those external structures may help the model better handle semantic relationships, which are currently difficult to capture without explicitly defined rules.

Furthermore, our model allows the seamlessly unification of other advanced methods in multi-relational embedding. The well-known models like TuckER [Balazevic et al., 2019], CompGCN [Vashishth et al., 2020], and BoxE [Abboud et al., 2020] have shown strong capabilities in modeling multi-relational structures and handling low-frequency entities by embedding relational semantics more richly and flexibly. Integrating these semantic-aware components into our architecture could further alleviate sparsity-related failures and improve both robustness and interpretability across a wider range of KGC scenarios.

A

Hyper-parameters

A.1. Parameters for rotate embedding model

As shown in Figure A.1 , in order to train the rotate embedding model, we take 1000 embedding dimensions to run 1000 epochs with 1024 batch sizes based on FB15k and FB15k237.

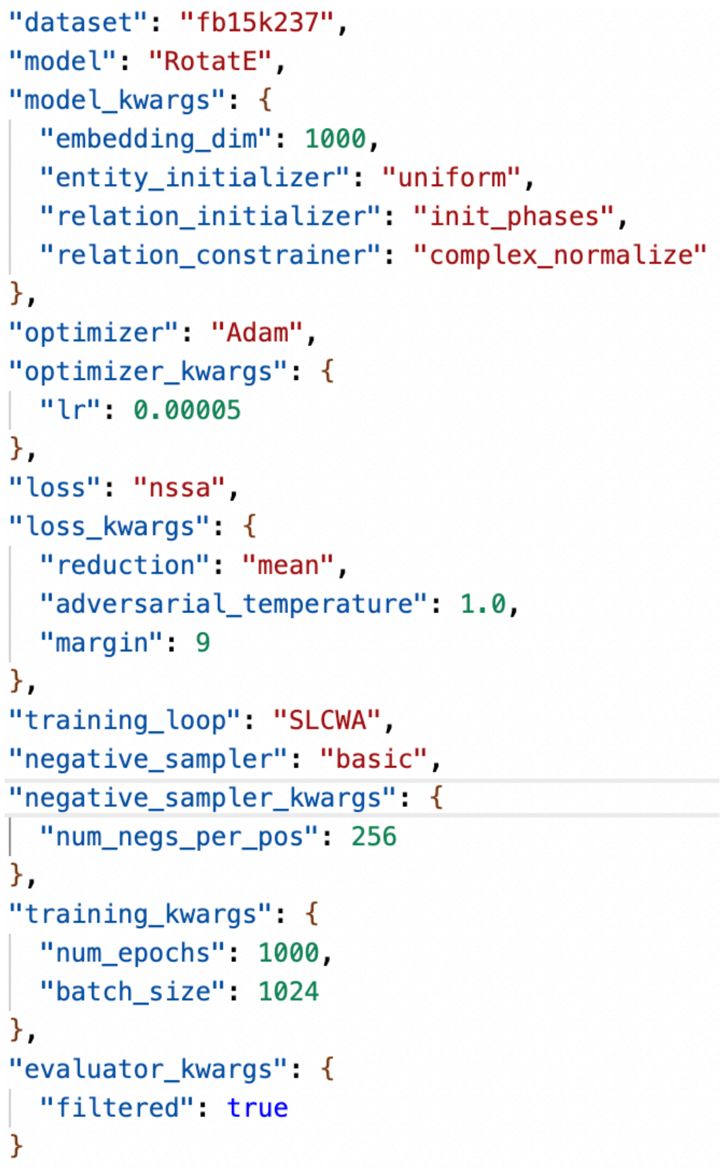
Different from them, we appropriately reduce the embedding dimensions at 200 for DRKG to decrease the training time and computational resources reasonably, also adjusting the number of epochs and batch size accordingly in Figure A.2. The number of negative samples per positive sample declined by half to trade off computational efficiency with model performance, reducing training time and memory usage while maintaining adequate contrastive learning.

A.2. Parameters for analogy-based rule mining model

As shown in Figure A.3 , the parameters enumerated came from the $log.txt$ file generated from the experimental execution for our analogy-based rule mining model in the FB15k and FB15k237 datasets, which are also applicable to DRKG except for MAX_LENGTH_CYCLIC. The detailed meaning of the different parameters can be found on the AnyBURL official web page `https://web.informatik.uni-mannheim.de/AnyBURL/`.

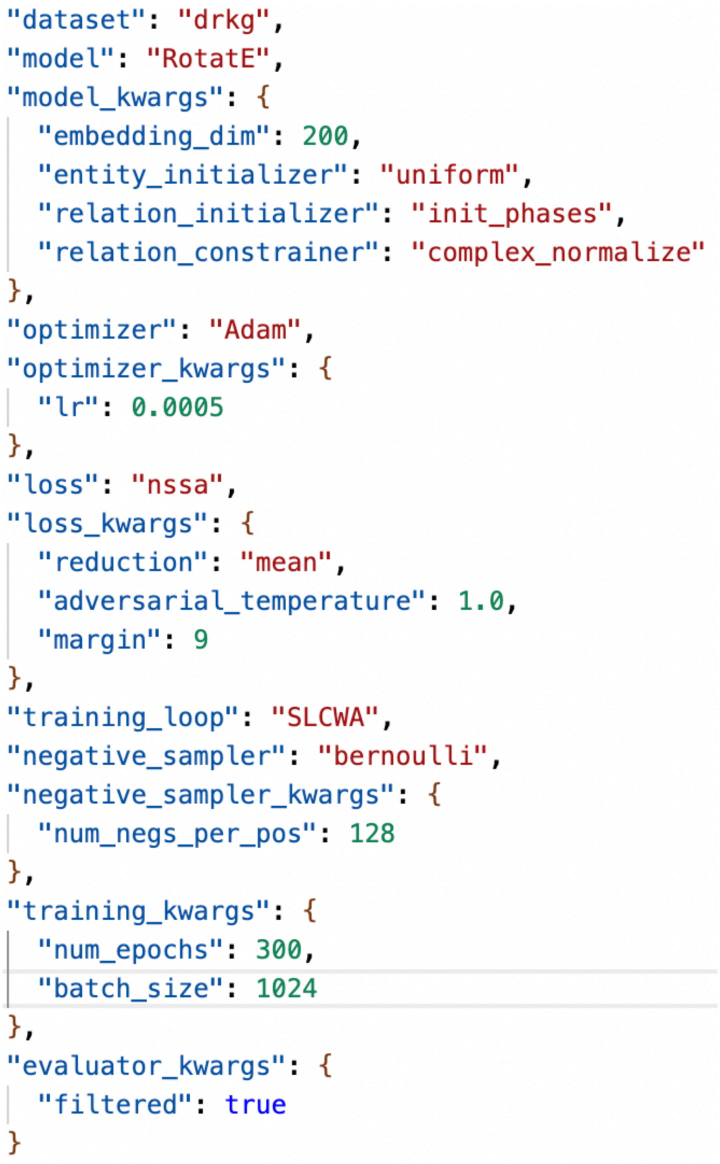
Although shorter rules are more general, they inevitably lack the specificity needed to capture complex patterns. Our study shows that adopting 4 as MAX_LENGTH_CYCLIC in DRKG performs better for inferring missing entities and can achieve better predictive performance. However, the computational cost grows with the exponentially increase in space complexity. Thus, a balanced strategy that takes into account both rule length and computational efficiency is essential to achieve optimal pperformance.

Also, the $WORKER\_THREADS$ is exactly depends on the core number in our remote server to fully speed up the learning. For our CPU-intensive tasks, this configuration ensures that each thread runs independently on a single core, maximizing computational resources.



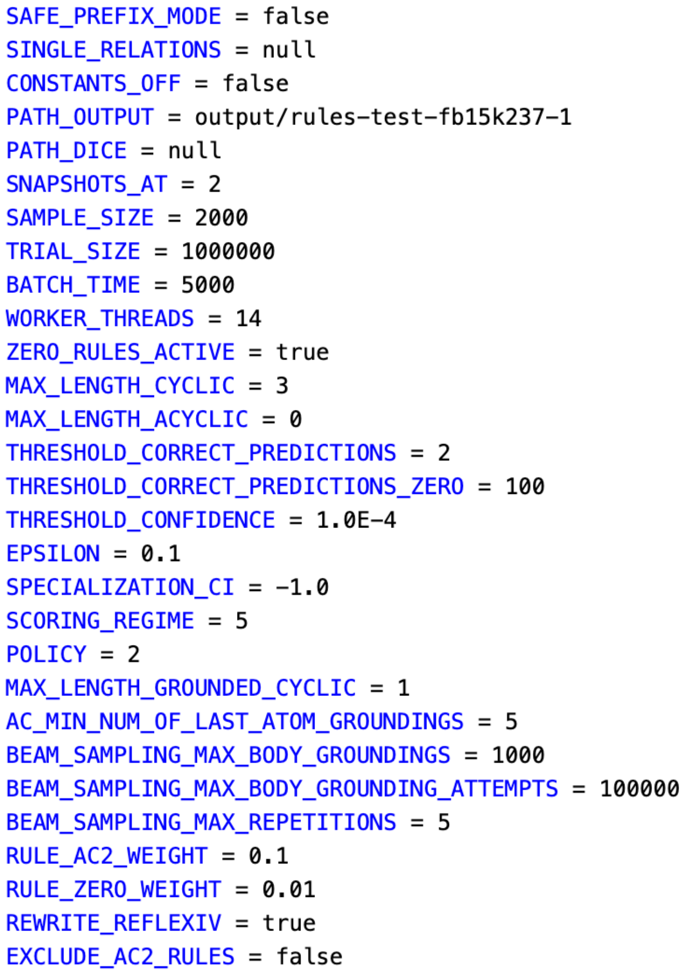
```
"dataset": "fb15k237",
"model": "RotatE",
"model_kwargs": {
  "embedding_dim": 1000,
  "entity_initializer": "uniform",
  "relation_initializer": "init_phases",
  "relation_constrainer": "complex_normalize"
},
"optimizer": "Adam",
"optimizer_kwargs": {
  "lr": 0.00005
},
"loss": "nssa",
"loss_kwargs": {
  "reduction": "mean",
  "adversarial_temperature": 1.0,
  "margin": 9
},
"training_loop": "SLCWA",
"negative_sampler": "basic",
"negative_sampler_kwargs": {
  "num_negs_per_pos": 256
},
"training_kwargs": {
  "num_epochs": 1000,
  "batch_size": 1024
},
"evaluator_kwargs": {
  "filtered": true
}
```

Figure A.1: The embedding model parameter on FB15k and FB15k-237.



```
"dataset": "drkg",
"model": "RotatE",
"model_kwargs": {
  "embedding_dim": 200,
  "entity_initializer": "uniform",
  "relation_initializer": "init_phases",
  "relation_constrainer": "complex_normalize"
},
"optimizer": "Adam",
"optimizer_kwargs": {
  "lr": 0.0005
},
"loss": "nssa",
"loss_kwargs": {
  "reduction": "mean",
  "adversarial_temperature": 1.0,
  "margin": 9
},
"training_loop": "SLCWA",
"negative_sampler": "bernoulli",
"negative_sampler_kwargs": {
  "num_negs_per_pos": 128
},
"training_kwargs": {
  "num_epochs": 300,
  "batch_size": 1024
},
"evaluator_kwargs": {
  "filtered": true
}
```

Figure A.2: The embedding model parameter on DRKG



```
SAFE_PREFIX_MODE = false
SINGLE_RELATIONS = null
CONSTANTS_OFF = false
PATH_OUTPUT = output/rules-test-fb15k237-1
PATH_DICE = null
SNAPSHOTS_AT = 2
SAMPLE_SIZE = 2000
TRIAL_SIZE = 1000000
BATCH_TIME = 5000
WORKER_THREADS = 14
ZERO_RULES_ACTIVE = true
MAX_LENGTH_CYCLIC = 3
MAX_LENGTH_ACYCLIC = 0
THRESHOLD_CORRECT_PREDICTIONS = 2
THRESHOLD_CORRECT_PREDICTIONS_ZERO = 100
THRESHOLD_CONFIDENCE = 1.0E-4
EPSILON = 0.1
SPECIALIZATION_CI = -1.0
SCORING_REGIME = 5
POLICY = 2
MAX_LENGTH_GROUNDED_CYCLIC = 1
AC_MIN_NUM_OF_LAST_ATOM_GROUNDINGS = 5
BEAM_SAMPLING_MAX_BODY_GROUNDINGS = 1000
BEAM_SAMPLING_MAX_BODY_GROUNDING_ATTEMPTS = 100000
BEAM_SAMPLING_MAX_REPETITIONS = 5
RULE_AC2_WEIGHT = 0.1
RULE_ZERO_WEIGHT = 0.01
REWRITE_REFLEXIV = true
EXCLUDE_AC2_RULES = false
```

Figure A.3: The rule-based model parameter on FB15k and FB15k-237

Bibliography

Abboud, R., İsmail İlkan Ceylan, Lukasiewicz, T., and Salvatori, T. (2020). Boxe: A box embedding model for knowledge base completion.

Alam, M., Gesese, G. A., and Paris, P.-H. (2024). Neurosymbolic methods for dynamic knowledge graphs.

Ali, M., Berrendorf, M., Hoyt, C. T., Vermue, L., Sharifzadeh, S., Tresp, V., and Lehmann, J. (2021). PyKEEN 1.0: A Python Library for Training and Evaluating Knowledge Graph Embeddings. *Journal of Machine Learning Research*, 22(82):1–6.

An, Z., Chen, B., Quan, H., Lin, Q., and Wang, H. (2020). Em-rbr: a reinforced framework for knowledge graph completion from reasoning perspective.

Bach, S. H., Broecheler, M., Huang, B., and Getoor, L. (2017). Hinge-loss markov random fields and probabilistic soft logic.

Balazevic, I., Allen, C., and Hospedales, T. (2019). Tucker: Tensor factorization for knowledge graph completion. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics.

Betz, P., Lüdtke, S., Meilicke, C., and Stuckenschmidt, H. (2023). On the aggregation of rules for knowledge graph completion.

Bi, Z., Cheng, S., Chen, J., Liang, X., Xiong, F., and Zhang, N. (2024). Relphormer: Relational graph transformer for knowledge graph representations. *Neurocomputing*, 566:127044.

Bonner, S., Barrett, I. P., Ye, C., Swiers, R., Engkvist, O., Bender, A., Hoyt, C. T., and Hamilton, W. L. (2022). A review of biomedical datasets relating to drug discovery: a knowledge graph perspective. *Briefings in Bioinformatics*, 23(6).

Bordes, A., Usunier, N., Garcia-Durán, A., Weston, J., and Yakhnenko, O. (2013a). Translating embeddings for modeling multi-relational data. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, page 2787–2795, Red Hook, NY, USA. Curran Associates Inc.

Bordes, A., Usunier, N., Garcia-Durán, A., Weston, J., and Yakhnenko, O. (2013b). Translating embeddings for modeling multi-relational data. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, page 2787–2795, Red Hook, NY, USA. Curran Associates Inc.

Chen, R., Wang, Y., Zhao, B., Cheng, J., Zhao, X., and Duan, Z. (2020). Knowledge graph completion: A review. *IEEE Access*, 8:192435–192456.

Cong, Q., Feng, Z., Li, F., Zhang, L., Rao, G., and Tao, C. (2018). Constructing biomedical knowledge graph based on semmeddb and linked open data. In *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 1628–1631.

DeLong, L. N., Mir, R. F., and Fleuriot, J. D. (2024). Neurosymbolic ai for reasoning over knowledge graphs: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, page 1–21.

Dettmers, T., Minervini, P., Stenetorp, P., and Riedel, S. (2018). Convolutional 2d knowledge graph embeddings.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding.

Galárraga, L., Teflioudi, C., Hose, K., and Suchanek, F. M. (2015a). Fast rule mining in ontological knowledge bases with amie

+

+. *The VLDB Journal*, 24(6):707–730.

Galárraga, L., Teflioudi, C., Hose, K., and Suchanek, F. M. (2015b). Fast rule mining in ontological knowledge bases with amie

+

+. *The VLDB Journal*, 24(6):707–730.

Galárraga, L. A., Teflioudi, C., Hose, K., and Suchanek, F. (2013). Amie: association rule mining under incomplete evidence in ontological knowledge bases. In *Proceedings of the 22nd International Conference on World Wide Web*, WWW '13, page 413–422, New York, NY, USA. Association for Computing Machinery.

Ge, X., Wang, Y. C., Wang, B., Kuo, C.-C. J., et al. (2024). Knowledge graph embedding: An overview. *APSIPA Transactions on Signal and Information Processing*, 13(1).

Hao, X., Ji, Z., Li, X., Yin, L., Liu, L., Sun, M., Liu, Q., and Yang, R. (2021). Construction and application of a knowledge graph. *Remote Sensing*, 13(13):2511.

Ioannidis, V. N., Song, X., Manchanda, S., Li, M., Pan, X., Zheng, D., Ning, X., Zeng, X., and Karypis, G. (2020). Drkg - drug repurposing knowledge graph for covid-19. `https://github.com/gnn4dr/DRKG/`.

Kejriwal, M. (2022). Knowledge graphs: A practical review of the research landscape. *Information*, 13(4):161.

Khajeh Nassiri, A., Pernelle, N., and Saïs, F. (2023). Regnum: Generating logical rules with numerical predicates in knowledge graphs. In *The Semantic Web: 20th International Conference, ESWC 2023, Hersonissos, Crete, Greece, May 28–June 1, 2023, Proceedings*, page 139–155, Berlin, Heidelberg. Springer-Verlag.

Kikuchi, M., Yoshida, M., Okabe, M., and Umemura, K. (2015). Confidence interval of probability estimator of laplace smoothing. In *2015 2nd International Conference on Advanced Informatics: Concepts, Theory and Applications (ICAICTA)*, page 1–6. IEEE.

Lao, N. and Cohen, W. (2010). Relational retrieval using a combination of path-constrained random walks. *Machine Learning*, 81:53–67.

Li, J., Xiang, J., and Cheng, J. (2023). Earr: Using rules to enhance the embedding of knowledge graph. *Expert Syst. Appl.*, 232(C).

Lin, Y., Liu, Z., Sun, M., Liu, Y., and Zhu, X. (2015). Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, page 2181–2187. AAAI Press.

Liu, H., Wu, Y., and Yang, Y. (2017). Analogical inference for multi-relational embeddings.

Loyola-Gonzalez, O. (2019). Black-box vs. white-box: Understanding their advantages and weaknesses from a practical point of view. *IEEE access*, 7:154096–154113.

Lu, Y., Goi, S. Y., Zhao, X., and Wang, J. (2025). Biomedical knowledge graph: A survey of domains, tasks, and real-world applications.

Meilicke, C., Chekol, M. W., Betz, P., Fink, M., and Stuckeschmidt, H. (2023). Anytime bottom-up rule learning for large-scale knowledge graph completion. *The VLDB Journal*, 33(1):131–161.

Meilicke, C., Chekol, M. W., Ruffinelli, D., and Stuckenschmidt, H. (2019). Anytime bottom-up rule learning for knowledge graph completion. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, IJCAI'19, page 3137–3143. AAAI Press.

Meilicke, C., Fink, M., Wang, Y., Ruffinelli, D., Gemulla, R., and Stuckenschmidt, H. (2018). Fine-grained evaluation of rule- and embedding-based systems for knowledge graph completion. In *The Semantic Web – ISWC 2018: 17th International Semantic Web Conference, Monterey, CA, USA, October 8–12, 2018, Proceedings, Part I*, page 3–20, Berlin, Heidelberg. Springer-Verlag.

Miao, S., Liu, M., and Li, P. (2022). Interpretable and generalizable graph learning via stochastic attention mechanism. In *International Conference on Machine Learning*, pages 15524–15543. PMLR.

Nadkarni, R., Wadden, D., Beltagy, I., Smith, N. A., Hajishirzi, H., and Hope, T. (2021). Scientific language models for biomedical knowledge base completion: An empirical study. *arXiv preprint arXiv:2106.09700*.

Nickel, M., Rosasco, L., and Poggio, T. (2015). Holographic embeddings of knowledge graphs.

Nickel, M., Tresp, V., and Kriegel, H.-P. (2011). A three-way model for collective learning on multi-relational data. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, page 809–816, Madison, WI, USA. Omnipress.

Peng, C., Xia, F., Naseriparsa, M., and Osborne, F. (2023). Knowledge graphs: Opportunities and challenges. *Artificial Intelligence Review*, 56(11):13071–13102.

Preuer, K., Klambauer, G., Rippmann, F., Hochreiter, S., and Unterthiner, T. (2022). *Interpretable Deep Learning in Drug Discovery*, page 331–345. Springer-Verlag, Berlin, Heidelberg.

Pryor, C., Dickens, C., Augustine, E., Albalak, A., Wang, W., and Getoor, L. (2023). Neupsl: Neural probabilistic soft logic.

Ren, H., Hu, W., and Leskovec, J. (2020). Query2box: Reasoning over knowledge graphs in vector space using box embeddings.

Richardson, M. and Domingos, P. (2006). Markov logic networks. *Mach. Learn.*, 62(1–2):107–136.

Schlichtkrull, M., Kipf, T. N., Bloem, P., van den Berg, R., Titov, I., and Welling, M. (2017). Modeling relational data with graph convolutional networks.

Shen, T., Zhang, F., and Cheng, J. (2022). A comprehensive overview of knowledge graph completion. *Knowledge-Based Systems*, 255:109597.

Srinivas, S. (2013). A generalization of the noisy-or model.

Sun, Z., Deng, Z.-H., Nie, J.-Y., and Tang, J. (2019). Rotate: Knowledge graph embedding by relational rotation in complex space.

Tiwari, S., Al-Aswadi, F. N., and Gaurav, D. (2021). Recent trends in knowledge graphs: theory and practice. *Soft Computing*, 25:8337–8355.

Trouillon, T., Welbl, J., Riedel, S., Éric Gaussier, and Bouchard, G. (2016). Complex embeddings for simple link prediction.

Tutek, M. and Šnajder, J. (2022). Toward practical usage of the attention mechanism as a tool for interpretability. *IEEE Access*, 10:47011–47030.

Vashishth, S., Sanyal, S., Nitin, V., and Talukdar, P. (2020). Composition-based multi-relational graph convolutional networks.

Wang, L., Zhao, W., Wei, Z., and Liu, J. (2022). Simkgc: Simple contrastive knowledge graph completion with pre-trained language models.

Wang, Z., Zhang, J., Feng, J., and Chen, Z. (2014). Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, AAAI'14, page 1112–1119. AAAI Press.

Wang, Z., Zhao, K., He, Y., Chen, Z., Ren, P., de Rijke, M., and Ren, Z. (2023). Iteratively learning representations for unseen entities with inter-rule correlations.

Wei, Y., Huang, Q., Zhang, Y., and Kwok, J. (2023). Kicgpt: Large language model with knowledge in context for knowledge graph completion. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, page 8667–8683. Association for Computational Linguistics.

Wellawatte, G. P., Gandhi, H. A., Seshadri, A., and White, A. D. (2023). A perspective on explanations of molecular prediction models. *Journal of Chemical Theory and Computation*, 19(8):2149–2160. PMID: 36972469.

Werner, L. (2024). Neuro-symbolic integration for reasoning and learning on knowledge graphs. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'24/IAAI'24/EAAI'24. AAAI Press.

Wikipedia (2024). Horn clause — wikipedia, the free encyclopedia. [Online; accessed 15-March-2025].

Wu, L., Sallinger, E., Sherkhonov, E., Vahdati, S., and Gottlob, G. (2022). Rule learning over knowledge graphs with genetic logic programming. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*, pages 3373–3385.

Xiong, W., Yu, M., Chang, S., Guo, X., and Wang, W. Y. (2018). One-shot relational learning for knowledge graphs.

Yang, B., tau Yih, W., He, X., Gao, J., and Deng, L. (2015). Embedding entities and relations for learning and inference in knowledge bases.

Yang, R., Zhu, J., Man, J., Liu, H., Fang, L., and Zhou, Y. (2025). Gs-kgc: A generative subgraph-based framework for knowledge graph completion with large language models.

Yao, L., Mao, C., and Luo, Y. (2019). Kg-bert: Bert for knowledge graph completion.

Zhang, J., Chen, B., Zhang, L., Ke, X., and Ding, H. (2021). Neural, symbolic and neural-symbolic reasoning on knowledge graphs. *AI Open*, 2:14–35.

Zhang, Y., Chen, Z., Guo, L., Xu, Y., Zhang, W., and Chen, H. (2024). Making large language models perform better in knowledge graph completion.

Zheng, Z., Dong, Y., Wang, S., Liu, H., Wang, Q., and Li, J. (2025). Kg-cf: Knowledge graph completion with context filtering under the guidance of large language models.

Zong, L., Xie, Z., Ma, C., Liu, X., Zhang, X., and Xu, B. (2024). RENN: A rule embedding enhanced neural network framework for temporal knowledge graph completion. In Calzolari, N., Kan, M.-Y., Hoste, V., Lenci, A., Sakti, S., and Xue, N., editors, *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 13919–13928, Torino, Italia. ELRA and ICCL.

Zou, X. (2020). A survey on application of knowledge graph. In *Journal of Physics: Conference Series*, volume 1487, page 012016. IOP Publishing.