Interpretable Link Prediction via Neural-Symbolic Reasoning

Rodrigo Castellano Ontiveros[1(✉)], Ehsan Bonabi Mobaraki[2], Francesco Giannini[3], Pietro Barbiero[4], Marco Gori[1], and Michelangelo Diligenti[1]

[1] University of Siena, Siena, Italy
{rodrigo.castellano,michelangelo.diligenti}@unisi.it
[2] Aalborg Universitet, Aalborg, Denmark
ebmo@cs.aau.dk
[3] Scuola Normale Superiore, Pisa, Italy
francesco.giannini@sns.it
[4] Università della Svizzera Italiana, Lugano, Switzerland
pietro.barbiero@usi.ch

Abstract. Knowledge Graph Embedding models have shown remarkable performances in different tasks like knowledge completion. However, they inherently lack interpretability, making it difficult to understand the reasoning behind their predictions. While different Neural-Symbolic (NeSy) models have been proposed to achieve interpretable reasoning through logic rules, existing evaluations primarily focus on accuracy, overlooking the critical assessment of explanation quality. This paper addresses this gap by introducing fully "interpretable-by-design" NeSy approaches for link prediction inspired by recently proposed models. Our framework employs reasoners that generate explicit logic proofs, utilizing either predefined or learned logic rules, ensuring transparent and explainable predictions. We go beyond traditional accuracy assessments, evaluating the quality of these explanations using established XAI metrics, including coherence. By quantitatively assessing the interpretability of our model, we aim to advance the development of trustworthy and understandable link prediction systems for Knowledge Graphs.

Keywords: Knowledge Graphs · Explainable AI · First-Order Logic

1 Introduction

Knowledge Graphs (KGs) are collections of incomplete factual knowledge represented as triplets (subject, predicate, object). Link prediction on KGs aims to infer these missing relationships by predicting the validity of new triplets based on the existing structure of the graph. Knowledge Graph Embedding (KGE) models have achieved remarkable success in link prediction tasks, effectively capturing complex relationships within knowledge graphs [19,23]. However, a significant drawback of these models from the eXplainable AI (XAI) perspective

R. Guidotti et al. (Eds.): xAI 2025, CCIS 2577, pp. 319–331, 2026.
https://doi.org/10.1007/978-3-032-08324-1_14

is their lack of transparency. The reasoning process behind their predictions is based on opaque latent representations, making it difficult to understand why a particular link is predicted [14]. While KGEs may implicitly encode certain relational or logic properties, such as transitivity or symmetry rules, it often remains unclear whether these properties are genuinely utilized in the prediction process [12,17]. Moreover, the reasoning performed by these methods can be prone to biases or impose overly rigid constraints, limiting their adaptability to diverse KGs.

XAI Methods over KGs. The majority of explainability methods for KGEs are symbolic based, where a rule miner is applied first and then logic reasoning can be applied on the extracted rule set. For instance, AMIE+ [10] mines rules by measuring the correlations over groups of atoms in the KG, and reports the most frequent rules according to different metrics such as the confidence or coverage. AnyBURL [16] and GenI [1] also exploit KG embeddings to learn the rules, but rely on ad hoc heuristics, generating tens of thousands of possible rules. These methods often rely on simple heuristic-based reasoners and, even if it is well known that they would benefit from using a fully fledge reasoner like ProbLog [7], scalability to even medium sized KGs would be hindered. To address the challenge of obscure reasoning in KGEs, various NeSy methods have also been proposed, with the aim of integrating symbolic logic rules into the reasoning process [25]. For instance, DRUM [20] and RNNLogic [18] exploit Recurrent Neural Networks to learn rules over KG embeddings, while similarly NCRL [6] adopts a recurrent attention unit. While these models improve reasoning capabilities compared to standard KGEs, they are often limited to simple composition rules and lack a comprehensive evaluation of the interpretability of their outputs. Since the predictions in these models often emerge as a join application of a large number of rules within a complex and non-linear decision process, the methods do not provide inspection capabilities to determine which rule is used to answer a query, therefore limiting their explainability. Indeed, the reported evaluations of these NeSy models primarily focus on predictive accuracy, neglecting the critical assessment of explanation quality. Recently, Relational Concept Bottleneck Models (R-CBMs) [3,15] have been proposed as a class of NeSy models capable of providing explanations on relational domains. We believe that R-CBMs represent a promising direction, utilizing logic rules to provide more interpretable query answering over KGs. However, the explanations generated by R-CBMs have been evaluated only qualitatively, lacking quantitative assessments of classic XAI metrics, like accuracy, coherence, and coverage. Moreover, R-CBMs predictions depend on an initial KGE score of a fact, thus not guaranteeing the rules are applied to answer the queries. In addition, the explanations were often limited to one-hop reasoning, failing to capture more intricate, multi-step logical proofs.

Our Contribution. In this paper, we address these limitations by defining a fully interpretable-by-design NeSy model over KGs, inspired by the message-passing scheme of R-CBMs. Furthermore, we enhance the explanation capabil-

ities of different NeSy models that can be considered as special R-CBMs. In particular, we make the following key contributions.

- We define a class of fully interpretable-by-design R-CBMs.
- We perform a thorough evaluation of the explanations generated by our framework, employing established XAI metrics such as accuracy and coherence. This allows us to quantify the interpretability of our model and provide a more rigorous assessment of the reasoning process.
- We extend the capabilities of models to generate more complex explanations in the form of deeper logic proofs, moving beyond simple one-hop reasoning. This enables the model to capture more nuanced and intricate relationships within the KG, providing more comprehensive and insightful explanations.

By quantifying the interpretability of our model and enabling the extraction of complex logic proofs, we aim to advance the development of trustworthy and understandable link prediction systems for KGs. We believe that this research contributes to a deeper understanding of the reasoning processes within NeSy models, paving the way for more transparent and reliable KG analysis.

The paper is organized as follows. Section 2 introduces the background on which our model (Sect. 3) is built. Section 4 reports our experimental analysis, and finally Sect. 5 draws some conclusions and future directions.

2 Preliminaries

Relational Languages. A relational setting can be formalized using a function-free First-Order Logic (FOL) language composed of constants (entities) $\mathcal{C}$, variables $\mathcal{X}$, and predicates (relations) $\mathcal{P}$. Atoms, such as $locIn(Paris, France)$ or $nation(x)$, are expressions of predicates (e.g. $locIn$ and $nation$) applied to entities (e.g. $Paris$ and $France$) or variables (e.g. x). Standard logic connectives $\{\neg, \wedge, \vee, \rightarrow\}$ and quantifiers $\{\forall, \exists\}$ are used to build literals (an atom or its negation) and more complex logic formulas from these atoms, such as $\forall x nation(x) \rightarrow \exists y locIn(x, y)$, expressing that "Each nation is in a continent". In this paper, we focus on logic theories, i.e. sets of formulas composed of Horn clauses, i.e. disjunctions of literals with a single positive literal. A Horn clause is equivalent to a rule $b_1 \wedge \ldots \wedge b_n \rightarrow h$, where $b_1, \ldots, b_n$ are called *body* atoms and h is called the *head* atom.

Grounding FOL Theories. Grounding converts formulas with variables into ground formulas, containing only constants, by substituting variables with specific constants. For example, grounding $nation(x) \rightarrow locIn(x, y)$ with the substitution $\{x/France, y/Europe\}$ yields $nation(France) \rightarrow locIn(France, Europe)$. The Herbrand Universe (HU) is the set of all possible ground formulas derived from a given FOL theory. Full grounding constructs the entire HU, while grounding often refers to creating only a subset. Methods like Markov Logic Networks use a Grounded Markov Network (GMN), a graph representation of the HU, where nodes are ground atoms and edges connect atoms that appear together in a grounded formula. For instance, given the GMN

$[nation(France) \rightarrow locIn(France, Europe)]$, we get nodes for $nation(France)$ and $locIn(France, Europe)$, and an edge connecting them. In the considered models, we rely on the GMN of a FOL theory to build a dependency graph.

Knowledge Graph Embeddings. Knowledge graphs (KGs) represent relational knowledge as graphs, where entities are nodes, relations are edges, and facts are triples of two entities and a relation. These graphs are generally incomplete, and Knowledge Graph Embeddings (KGEs) address the task of knowledge graph completion by mapping entities and relations to a latent space, thus predicting of missing facts. KGEs learn embeddings by optimizing scoring functions that align with observed data. For example, RotatE [21] models relations as rotations in the complex embedding space, where each relation p corresponds to a rotation from the subject entity a to the object entity b. RotatE assigns each entity and relation an embedding vector e_a, e_b, and e_p, respectively, and calculates the score of the fact $p(a, b)$ by the distance between $e_a \circ e_p$ and e_b, being $\circ$ the Hadamard product. Other KGEs are ComplEx [22] or TransE [4].

Relational Concept Bottleneck Models. Relational Concept Bottleneck Models (R-CBMs) [3] merge concept-based XAI [13] and Graph Neural Networks (GNNs) [24] for relational domains. They process atoms via an encoder, predict with a scoring function, and aggregate predictions using a GNN-like dependency graph. The pipeline includes: (i) atom encoding and prediction, (ii) message-passing on the atom dependency graph, and (iii) prediction aggregation.

(i) A ground atom $A = p(a, b)$ is initially encoded as $h^0(A) = g_p(\mathbf{e}_a, \mathbf{e}_b) \in \mathbb{R}^H$, where a, b are entities and $\mathbf{e}_a, \mathbf{e}_b \in \mathbb{R}^H$ are their embeddings, p is a relation, g_p is the atom encoder, and H denotes the embedding size. The initial prediction is calculated as $y^0(A) = s(h^0(A))$, $s : \mathbb{R}^H \rightarrow [0, 1]$ being a learnable predictor, such as an MLP with a sigmoid activation function or a kge scoring function.

(ii)-(iii) Update of the embeddings and predictions of the atoms is expressed as a message-passing GNNs scheme [11] over the GMN of a FOL theory R. For every atom $A = p(a, b)$, we denote by $\mathcal{N}_r(A)$ the set of nodes connected to A via the rule $r \in R$ in the GMN, and by $\mathcal{R}(A)$ the set of rules containing the atom A. Then for T steps, the message-passing of R-CBMs is defined as:

$$h_r^t(A) = u_{l(r)}\left(h^{t-1}(A), \left[h^{t-1}(B)\right]_{B \in \mathcal{N}_r(A)}\right) \tag{1}$$

$$y_r^t(A) = f_{l(r)}\left(y^{t-1}(A), \left[h_r^t(B), y^{t-1}(B)\right]_{B \in \mathcal{N}_r(A)}\right) \tag{2}$$

$$h^t(A) = \sum\nolimits_{r \in \mathcal{R}(A)} h_r^t(A) \tag{3}$$

$$y^t(A) = \bigoplus\nolimits_{r \in \mathcal{R}(A)} y_r^t(A) \tag{4}$$

where $u_{l(r)}$ and $f_{l(r)}$ represent edge-type dependent functions. Specifically, $u_{l(r)}$ performs a combine/update step, yielding an improved latent representation $h_r^t(A)$, while $f_{l(r)}$ executes a local readout, producing a prediction based solely on the neighborhood $\mathcal{N}_r(A)$. The symbol $\bigoplus$ denotes an aggregation operation, such as maximum or summation, applied to the predictions across all neighborhoods r belonging to the set $\mathcal{R}(A)$.

3 Methods

R-CBMs are not fully interpretable because the final prediction aggregates the predictions per-rule $y_r^t(A)$ for an atom A using a generic aggregator, which is generally opaque and complex. Furthermore, the dependency of the per-rule output $y_r^t(A)$ directly depends on the output of latent representations h_r^t, which are black boxes for a human operator. In this section, we propose a sequence of modifications to define a class of R-CBMs, where the decision process corresponds to a logic reasoning process, which can be unwound and traced back to provide a human interpretable explanation. In all the proposed algorithms, the basic idea is to structure the computation into a logic rule generation stage, which can depend on latent representations, and a rule execution stage, which is fully interpretable and understandable.

Depending on the different design decisions, we propose different models, providing different trade-offs between expressivity and interpretability, relying on the FOL theory of Horn clauses R.

3.1 Interpretable Semantic Based Regularization (I-SBR)

Semantic Based Regularization (SBR) [9] conjuncts the predictions of the body atoms to compute the prediction of the head atom in every rule r, using a selected t-norm. These methods were originally limited to unary predicates and a single reasoning propagation step. Here, we propose a relational recursive extension based on message-passing:

$$y_r^t(A) = t\text{-}norm\left(\left[y^{t-1}(B)\right]_{B\in\mathcal{N}_r(A)}\right) \tag{5}$$

$$y^t(A) = \max_{r\in\mathcal{R}(A)} y_r^t(A)\ , \tag{6}$$

where the y_r are initialized using $y_r^0(\alpha) = kge(\alpha)$, where α is the head node. There are t reasoning hops by applying the rules in R, with $0 \leq t \leq T$. The idea of the proposed architecture is to derive new facts by recursively executing the rules, which is equivalent to performing forward chaining starting from the kge predictions. Forward chaining in I-SBR relies on t-norms for the logic execution step, which guarantee differentiability and the possibility to train the kge end-to-end within a single computation graph. Thanks to the use of maximum aggregation in the computation of the prediction $y^t(\alpha)$, the score can be traced back from α to the body nodes, and this process can be recursively repeated providing a proof tree which can be provided as an explanation. In the experimental section, we show different examples of proof trees extracted using this methodology.

3.2 Interpretable Deep Concept Reasoners (I-DCR)

DCR [2] learns a formula for each head atom, given a set of candidate body atoms, then computes the output by using a t-norm:

$$y_r^t(A) = t\text{-}norm\left(\Phi_r(h^0(B), y^{t-1}(B))\right) \tag{7}$$

$$y^t(A) = \max_{r \in \mathcal{R}(A)} y_r^t(A) \tag{8}$$

where the y_r are initialized using $y_r^0(\alpha) = kge(\alpha)$, and $\Phi_r : \mathbb{R}^{H+1} \rightarrow [0, 1]$ represents a logic formula processing the embedding representation and prediction of each atom in each rule r, to get a learned Horn Clause. In the original formulation, DCR was defined for a single step of propagation, however, we extend DCR to multiple iterations t, with $0 \leq t \leq T$, to enable multi-hop reasoning and restrict it to a max aggregation operator to merge the information from different rules. The resulting architecture, called I-DCR, can take advantage of latent representations to discover the rules to apply in a given context, unlike I-SBR, which assumes all rules to be predefined. I-DCR also preserves full human interpretability, as the generated rules are executed symbolically. Like in I-SBR, the use of t-norms to perform the logic reasoning step allows an end-to-end optimization of the KGE layer.

4 Experiments

4.1 Experimental Setup

We conducted a comprehensive series of experiments on diverse benchmark datasets to evaluate our proposed approach from different points of view.

The three benchmark datasets used for evaluation are: Countries [5], Family [6], and WN18RR [8]. The Countries dataset consists of three tasks (S1, S2, S3) that increase in difficulty, predicting locations based on neighborhood relations and locations. Family encodes familial relationships. WN18RR, derived from WordNet, ensures no inverse relation leakage.

The countries dataset follows predefined logical rules:

R_1: $LocIn(x, w) \wedge LocIn(w, z) \rightarrow LocIn(x, z)$

R_2: $NeighOf(x, y) \wedge LocIn(y, z) \rightarrow LocIn(x, z)$

R_3: $NeighOf(x, y) \wedge NeighOf(y, k) \wedge LocIn(k, z) \rightarrow LocIn(x, z)$

Rules for other datasets are extracted using AMIE [10], selecting them based on standard confidence. Dataset statistics are shown in Table 1.

Baseline Models. We compare two Neural-Symbolic (NeSy) models, I-SBR and I-DCR, against the KGE ComplEx.

Hyperparameter Settings. All models use 100-dimensional embeddings, Adam optimizer (10^{-2} learning rate), and are trained for 100 epochs.

Table 1. Detailed statistics of the datasets employed in our experiments.

Dataset	#Entities	#Relations	#Facts	Avg. Degree	#Rules
Countries S1	272	3	1,110	4.28	1
Countries S2	272	4	1,062	4.35	2
Countries S3	272	4	978	4.35	3
Family	3007	12	19,845	6.47	48
WN18RR	40,559	11	86,835	2.14	28

4.2 Evaluation Metrics

Mean Reciprocal Rank (MRR) measures the average reciprocal rank of the first correct answer in a ranked list:

$$\text{MRR} = \frac{1}{|Q|} \sum_{q \in Q} \frac{1}{\text{rank}_q} \tag{9}$$

where Q is the set of queries, and rank_q is the position of the first correct answer.

Hits@N computes the proportion of queries where a correct answer appears within the top-N predictions:

$$\text{Hits@N} = \frac{1}{|Q|} \sum_{q \in Q} I(\exists \text{ correct answer in top-}N) \tag{10}$$

where $I(\cdot)$ is an indicator function. In neural-symbolic models, explanations depend on the accuracy of predictions. Thus, high MRR and Hits@N are not just performance metrics but necessary conditions for generating faithful and trustworthy explanations, reinforcing their central role in evaluating explainability.

Coherence measures the agreement between two models by computing the ratio of queries for which both models produce the same top-ranked prediction:

$$\text{Coherence} = \frac{1}{|Q|} \sum_{q \in Q} I(\text{top prediction}_{\text{model}_1} = \text{top prediction}_{\text{model}_2}) \tag{11}$$

where $I(\cdot)$ is an indicator function [2,13].

Together, these metrics capture both the predictive performance of the model (MRR, Hits @ N) and the transparency of its reasoning (coherence, proof traces).

4.3 Results

The primary objective of our experimental evaluation is to assess the performance of our proposed reasoners, namely I-SBR and I-DCR, in comparison with a baseline approach across multiple datasets. The baseline model used for this evaluation is ComplEx, which serves both as a direct benchmark and as the knowledge graph embedding method for the reasoners. By analyzing different

Table 2. Results for the metrics MRR and Hits (H) for the Countries Dataset.

Countries S2				
Model	MRR	H@1	H@3	H@10
ComplEx	0.976	0.954	0.996	1.0
I-DCR	0.969	0.958	0.967	1.0
I-SBR	0.969	0.958	0.962	1.0

Countries S3				
Model	MRR	H@1	H@3	H@10
ComplEx	0.866	0.808	0.879	1.0
I-DCR	0.979	0.962	0.996	1.0
I-SBR	0.983	0.967	1.0	1.0

Table 3. Results for the metrics MRR and Hits (H) for the Family and WN18RR datasets. For each method M, we indicate with $M^\star$ the results obtained when restricting the training set to the set of queries which are provable in the logic theory defined by the considered set of rules.

Family				
Model	MRR	H@1	H@3	H@10
ComplEx	0.787	0.653	0.916	0.951
I-DCR	0.764	0.764	0.764	0.764
I-SBR	0.764	0.763	0.764	0.764
I-DCR*	1.0	1.0	1.0	1.0
I-SBR*	1.0	1.0	1.0	1.0

WN18RR				
Model	MRR	H@1	H@3	H@10
AMIE+		0.358	-	0.388
AnyBurl	∼0.47	∼0.44	-	∼0.55
ComplEx	0.384	0.376	0.387	0.397
I-DCR	0.380	0.350	0.370	0.381
I-SBR	0.382	0.340	0.369	0.385
I-DCR*	0.938	0.899	0.956	0.985
I-SBR*	0.931	0.906	0.965	0.987

evaluation metrics, we aim to provide insights into how well the reasoners perform in terms of ranking quality and coherence of predictions[1]

In the Countries dataset, the results for the S1 benchmark have been omitted due to the saturation of the evaluation metrics, which reach the optimal maximum value across all tested methods. In the S2 benchmark, ComplEx achieves an MRR score of 0.976 MRR. Both I-SBR and I-DCR exhibit very similar performance to the baseline when evaluated using MRR and Hits metrics, yielding a score of 0.969. Coherence of the I-SBR and I-DCR with the ComplEx model providing the initial embedded representations of the atoms is high, largely due to the overall strong performance of all models across the other metrics. However, in scenario S3, the performance difference becomes more pronounced. The reasoners outperform ComplEx significantly, with over a 10% increase in MRR and a 15% increase in Hits metrics. This substantial improvement corresponds to a lower coherence score with ComplEx, which is recorded at 0.80 for I-SBR and 0.81 for I-DCR. These results are detailed in Table 2 and 4.

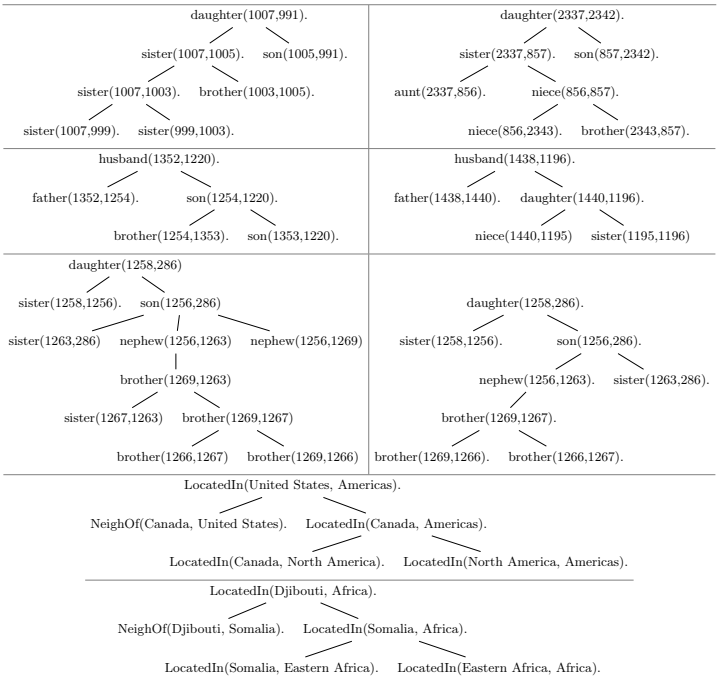
For the Family and WN18RR datasets, the reasoners perform closely to the baseline as shown in Table 3. In the case of Family, the reasoners outperform ComplEx significantly for Hits@1. For WN18RR, the performance remains close to ComplEx, in spite of the inherent explainability of our reasoners. In the case of WN18RR, coherence is relatively low due to the generally lower scores across

[1] The code is available at https://github.com/rodrigo-castellano/Interpretable-NeSy.

Table 4. Coherence values for I-SBR and I-DCR against the ComplEx model providing the initial embedded representations of the atoms.

Model	Countries S2	Countries S3	Family	WN18RR
I-DCR	0.934	0.812	0.654	0.362
I-SBR	0.942	0.800	0.654	0.365

Table 5. Examples of local explanations as proof trees obtained for the Kinship Family and Countries datasets.



all methods, making it difficult to obtain an identical top-ranked prediction. When comparing against existing symbolic reasoners, AMIE+ achieves similar performance to our models, whereas AnyBurl outperforms them. It is essential to highlight that our methods rely on a small and controlled set of human-understandable logical rules. On the other hand, AnyBurl/AMIE move the complexity in the rule extraction phase, which returns a set of complex and not-human understandable rules which are at least two orders of magnitude larger than the set that we consider. For example, AnyBurl considers more than 75000 automatically mined rules for WN18RR, which are mostly applied in a shallow fashion. This distinction makes our models more interpretable as the generated proofs closely resemble the step-by-step reasoning process of humans, unlike the single application of not intuitive and hard to understand complex rules used by the competitors [2,13].

Table 6. Sample of the global explanations obtained by aggregating the local explanations over a dataset.

Dataset	Rule
Countries S1	$\forall a, b \exists c locIn(a, c) \wedge locIn(c, b) \rightarrow locIn(a, b)$
Countries S2	$\forall a, b \exists c neighOf(a, c) \wedge locIn(c, b) \rightarrow locIn(a, b)$
Countries S3	$\forall a, b \exists c, d neighOf(a, c) \wedge neighOf(c, d) \wedge locIn(d, b) \rightarrow locIn(a, b)$
	$\forall a, b \exists c neighOf(a, c) \wedge locIn(c, b) \rightarrow locIn(a, b)$
Family	$\forall a, b \exists c aunt(a, c) \wedge sister(b, c) \rightarrow aunt(a, b)$
	$\forall a, b \exists c aunt(a, c) \wedge brother(b, c) \rightarrow aunt(a, b)$
	$\forall a, b \exists c sister(a, c) \wedge father(c, b) \rightarrow aunt(a, b)$
	$\forall a, b \exists c sister(a, c) \wedge son(b, c) \rightarrow aunt(a, b)$
	$\forall a, b \exists c sister(a, c) \wedge mother(c, b) \rightarrow aunt(a, b)$
	$\forall a, b \exists c sister(a, c) \wedge daughter(b, c) \rightarrow aunt(a, b)$
WN18RR	$\forall a, b \exists c also\_see(b, a) \rightarrow also\_see(a, b)$
	$\forall a, b \exists c also\_see(a, c) \wedge also\_see(c, b) \rightarrow also\_see(a, b)$
	$\forall a, b deriv\_related\_form(b, a) \rightarrow deriv\_related\_form(a, b)$
	$\forall a, b \exists c has\_part(a, c) \wedge hypernym(c, b) \rightarrow hypernym(a, b)$
	$\forall a, b \exists c verb\_group(a, c) \wedge hypernym(c, b) \rightarrow hypernym(a, b)$

Due to the simplicity and compactness of the rule set used by our reasoners, not all test queries are provable within the logical framework. This is reflected in the rule coverage values of the datasets: 1.0 for Countries, 0.77 for Family, and 0.40 for WN18RR. For that reason, we decided to evaluate the models only on the test queries that can be proven with our rules. Under these conditions, both the baseline and the reasoners exhibit improved performance. This improvement is particularly striking for Family, where all metrics reach their maximum possible values. Similarly, in the case of WN18RR, the baseline and reasoner achieve MRR scores of 0.933 and 0.938, respectively.

4.4 Explanations

Global Explanations. Global explanations provide insight into the overall reasoning process by integrating local explanations across multiple test cases. This integration allows us to identify patterns in how the NeSy method classifies triplets, highlighting the logical rules most frequently used. Table 6 presents several representative examples from different datasets, showcasing the key logical patterns discovered by our model.

Local Explanations. The extracted local explanations take advantage of the deep logical reasoning used by the models. These proofs are not just chains, but are proof trees of arbitrary depth obtained thanks to the first-order logic formulas. This leads to a richer set of explanations that cover more queries. Using simple rules like those in Table 6 to build a proof tree is more intuitive and easier to understand than relying on long, flat rules. In contrast, applying such

extended rules, often produced in large numbers by systems such as AnyBurl [16], can obscure the reasoning process and diminish explanation clarity [2,16].

Table 5 presents examples of local explanations as proof trees obtained for the Family dataset. These proof trees demonstrate how multiple reasoning steps contribute to deriving conclusions. These structured explanations reinforce the transparency of our model and highlight the advantage of incorporating first-order logic into knowledge-graph reasoning.

5 Conclusions and Future Work

This paper presents a class of neural-symbolic methods, which integrate latent representations and logic reasoning. The main advantage of this methodology is in the fact that, once the latent representations have instantiated the reasoning process, the final decision can be explained with high interpretability to a human operator. We presented an application in the domain of link prediction for knowledge graphs. Unlike most symbolic approaches for link prediction, our methodology is designed to allow a full retracing of the reasoning steps, resulting in explanations in the form of deep logical proofs, instead of the shallow rules typically used by symbolic KG methods. Results have shown that the methods perform well for the selected datasets when compared to the baseline, improving results in some cases due to multi-hop reasoning.

As future work, we plan to expand the set of methods defined within the framework, as well as to consider more complex rule sets. Finally, we plan to apply the same methodology for post-hoc explainers, by distilling a black-box model into its interpretable neural-symbolic counterpart.

Acknowledgments. This work has been partially supported by the Partnership Extended PE00000013 - "FAIR - Future Artificial Intelligence Research" - Spoke 1 "Human-centered AI". This work was also supported by the EU Framework Program for Research and Innovation Horizon under the Grant Agreement No 101073307 (MSCA-DN LeMuR).

Disclosure of Interests. The authors have no competing interests.

References

1. Amador-Domínguez, E., Serrano, E., Manrique, D.: GEnI: a framework for the generation of explanations and insights of knowledge graph embedding predictions. Neurocomputing **521**, 199–212 (2023)
2. Barbiero, P., et al.: Interpretable neural-symbolic concept reasoning. In: ICML, pp. 1801–1825 (2023)
3. Barbiero, P., Giannini, F., Ciravegna, G., Diligenti, M., Marra, G.: Relational concept bottleneck models. In: NeurIPS (2024)
4. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: NeurIPS, vol. 26, pp. 2787–2795 (2013)

5. Bouchard, G., Singh, S., Trouillon, T.: On approximate reasoning capabilities of low-rank vector spaces. In: AAAI Spring Symposia (2015)
6. Cheng, K., Amed, N.K., Sun, Y.: Neural compositional rule learning for knowledge graph reasoning. In: ICLR (ICLR) (2023)
7. De Raedt, L., Kimmig, A., Toivonen, H.: Problog: a probabilistic prolog and its application in link discovery. In: IJCAI (2007)
8. Dettmers, T., Minervini, P., Stenetorp, P., Riedel, S.: Convolutional 2D knowledge graph embeddings. In: Proceedings of the AAAI Conference (2018)
9. Diligenti, M., Gori, M., Sacca, C.: Semantic-based regularization for learning and inference. Artif. Intell. **244**, 143–165 (2017)
10. Galárraga, L., Teflioudi, C., Hose, K., Suchanek, F.M.: Fast rule mining in ontological knowledge bases with AMIE+. VLDB J. **24**(6), 707–730 (2015)
11. Gilmer, J., Schoenholz, S.S., Riley, P.F., Vinyals, O., Dahl, G.E.: Message passing neural networks. In: Machine Learning Meets Quantum Physics, pp. 199–214. Springer (2020)
12. Gutierrez Basulto, V., Schockaert, S.: From knowledge graph embedding to ontology embedding? An analysis of the compatibility between vector space representations and rules (2018)
13. Koh, P.W., et al.: Concept bottleneck models. In: ICML, pp. 5338–5348. PMLR (2020)
14. Lecue, F.: On the role of knowledge graphs in explainable AI. Semantic Web **11**(1), 41–51 (2020)
15. Marra, G., Diligenti, M., Giannini, F.: Relational reasoning networks. Knowl.-Based Syst. 112822 (2025)
16. Meilicke, C., Chekol, M.W., Betz, P., Fink, M., Stuckeschmidt, H.: Anytime bottom-up rule learning for large-scale knowledge graph completion. VLDB J. **33**(1), 131–161 (2024)
17. Pavlović, A., Sallinger, E.: Building bridges: knowledge graph embeddings respecting logical rules. In: 15th Alberto Mendelzon International Workshop on Foundations of Data Management (2023)
18. Qu, M., Chen, J., Xhonneux, L.P., Bengio, Y., Tang, J.: RNNLogic: learning logic rules for reasoning on knowledge graphs. In: ICLR (2020)
19. Rossi, A., Barbosa, D., Firmani, D., Matinata, A., Merialdo, P.: Knowledge graph embedding for link prediction: a comparative analysis. ACM TKDD **15**(2), 1–49 (2021)
20. Sadeghian, A., Armandpour, M., Ding, P., Wang, D.Z.: Drum: end-to-end differentiable rule mining on knowledge graphs. NeurIPS **32** (2019)
21. Sun, Z., Deng, Z.H., Nie, J.Y., Tang, J.: Rotate: knowledge graph embedding by relational rotation in complex space. In: ICLR (2019)
22. Trouillon, T., Welbl, J., Riedel, S., Gaussier, É., Bouchard, G.: Complex embeddings for simple link prediction. In: ICML, pp. 2071–2080. PMLR (2016)
23. Wang, M., Qiu, L., Wang, X.: A survey on knowledge graph embeddings for link prediction. Symmetry **13**(3), 485 (2021)
24. Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., Philip, S.Y.: A comprehensive survey on graph neural networks. IEEE Trans. Neural Netw. Learn. Syst. **32**(1), 4–24 (2020)
25. Zhang, W., Chen, J., Li, J., Xu, Z., Pan, J.Z., Chen, H.: Knowledge graph reasoning with logics and embeddings: survey and perspective. In: 2024 IEEE International Conference on Knowledge Graph (ICKG), pp. 492–499. IEEE (2024)