NEURO-SYMBOLIC ENTITY ALIGNMENT VIA VARIA TIONAL INFERENCE

Anonymous authors

Paper under double-blind review

ABSTRACT

Entity alignment (EA) aims to merge two knowledge graphs (KGs) by identifying equivalent entity pairs. Existing methods can be categorized into symbolic and neural models. Symbolic models, while precise, struggle with substructure heterogeneity and sparsity, whereas neural models, although effective, generally lack interpretability and cannot handle uncertainty. We propose NeuSymEA, a probabilistic neuro-symbolic framework that combines the strengths of both methods. NeuSymEA models the joint probability of all possible pairs' truth scores in a Markov random field, regulated by a set of rules, and optimizes it with the variational EM algorithm. In the E-step, a neural model parameterizes the truth score distributions and infers missing alignments. In the M-step, the rule weights are updated based on the observed and inferred alignments. To facilitate interpretability, we further design a path-ranking-based explainer upon this framework that generates supporting rules for the inferred alignments. Experiments on benchmarks demonstrate that NeuSymEA not only significantly outperforms baselines in terms of effectiveness and robustness, but also provides interpretable results.

024 025 026

004

010 011

012

013

014

015

016

017

018

019

021

1 INTRODUCTION

027 028

029 Knowledge graphs (KGs) are crucial for organizing structured knowledge about entities and their relationships, enhancing search capabilities across various applications. They are widely used in question-answering systems (Bast & Haussmann, 2015; Dong et al., 2023), recommendation sys-031 tems (Catherine & Cohen, 2016), social network analysis (Tang et al., 2008), Natural Language Processing (Weikum & Theobald, 2010), etc.. Despite their utility, real-world KGs often face issues 033 like incompleteness, domain specificity, and language constraints, which hinder their effectiveness 034 in cross-disciplinary or multilingual contexts. To address these issues, entity alignment (EA) aims to merge disparate KGs into a unified, comprehensive knowledge base by identifying and linking equivalent entities across different KGs. For example, aligning entities between a biomedical KG 037 and a pharmaceutical KG allows for mining cross-discipline relationships through the aligned en-038 tities, such as identifying the same drugs and their effects on different diseases to enhance drug repurposing efforts. This alignment enables more nuanced exploration and interrogation of interconnected data, providing richer insights into how entities function across multiple domains. 040

- 041 Entity alignment models aim to determine the equivalence of two entities by assessing their align-042 ment probability. Existing methods can be broadly categorized into symbolic models and neural 043 models. Symbolic models (Suchanek et al., 2012; Jiménez-Ruiz & Cuenca Grau, 2011; Qi et al., 044 2021) provide interpretable and precise inference by mining ground rules, but they struggle with aligning long-tail entities, especially those without aligned neighbors. In such cases, the lack of supporting rules leads to low recall. Conversely, neural models, such as translation models (Chen 046 et al., 2017; Sun et al., 2018) and Graph Convolutional Networks (GCNs) (Mao et al., 2020; 2021; 047 Wang et al., 2018; Mao et al., 2022; Li et al., 2024), excel in recalling similar entities by embedding 048 them in a continuous space, yet they often fail to distinguish entities with similar representations, 049 causing a drop in precision as the entity pool grows. Neuro-symbolic models aim to combine the strengths of both approaches, offering the interpretability and precision of symbolic models along-051 side the high recall capabilities of neural models. 052
- However, neuro-symbolic reasoning in entity alignment (EA) faces several challenges. First, combining symbolic and neural models into a unified framework is suboptimal due to the difficulty in

aligning their objectives. Current approaches either use neural models as auxiliary modules for symbolic models to measure entity similarity (Qi et al., 2021) or employ symbolic models to refine pseudo-labels (Liu et al., 2022; Chen et al., 2024b). Second, in EA task, the search space for rules is large, as the EA task requires deriving ground rules from both intra-KG and inter-KG structural patterns, leading to an exponentially large search space with increasing rule length. Finally, generating interpretations for EA remains underexplored. Effective interpretations should not only generate supporting rules but also quantify their confidence through rule weights.

061 To overcome these challenges, we propose NeuSymEA, a neuro-symbolic framework that combines 062 the strengths of both symbolic and neural models. NeuSymEA models the joint probability of truth 063 score assignment for all possible entity pairs using a Markov random field, regulated by a set of 064 weighted rules. This joint probability is optimized via a variational EM algorithm. During the Estep, a neural model parameterizes the truth scores and infers the missing alignments. In the M-step, 065 the rule weights are updated based on both observed and inferred alignments. To leverage long 066 rules without suffering from the exponential search space, we employ logic deduction to decompose 067 rules of any length into shorter, unit-length rules. This allows for efficient inference and weight 068 updates for long rules. After training, the learned rules are adapted into an explainer, enhancing 069 interpretability. Specifically, we reverse the logic deduction process to calculate the weights of long rules based on the learned weights of their shorter counterparts. Additionally, an explainer is 071 introduced to efficiently generate supporting rules with quantified confidence through breadth-first search, as explanations for inferred alignments. Our contributions are summarized as below: 073

- A principled neuro-symbolic reasoning framework: NeuSymEA seamlessly integrates neural and symbolic models through a principled variational EM framework, combining their strengths for effective entity alignment.
- Efficient optimization via logical decomposition: We introduce a logic deduction mechanism that decomposes long rules into shorter ones, significantly reducing the complexity of rule inference and enabling efficient reasoning over large knowledge graphs.
- Interpretable inference: The explainer utilizes learned rules to generate support paths for interpreting both aligned and misaligned pairs. It offers two modes: (1) Hard-anchor mode—generates supporting paths from prealigned anchor pairs; and (2) Soft-anchor mode—incorporates inferred anchor pairs for more informative interpretation.
 - Empirical validation and superior results: NeuSymEA demonstrates state-of-the-art performance on benchmark datasets, delivering both robust alignment accuracy and insightful rule-based interpretations.
- 2 PRELIMINARIES
- 2.1 PROBLEM STATEMENT

A knowledge graph \mathcal{G} comprises a set of entities \mathcal{E} , a set of relations \mathcal{R} , and a set of relation triples \mathcal{T} where each triple $(e_i, r_k, e_j) \in \mathcal{T}$ represents a directional relationship between its head entity and tail entity. Given two KGs $\mathcal{G} = \{\mathcal{E}, \mathcal{R}, \mathcal{T}\}, \mathcal{G}' = \{\mathcal{E}', \mathcal{R}', \mathcal{T}'\}$, and a set of observed aligned entity pairs $\mathcal{O} = \{(e_i, e'_i) | e_i \in \mathcal{E}, e'_i \in \mathcal{E}'\}_{i=1}^n$, the goal of entity alignment is to infer the missing alignments by reasoning with the existing alignments. This problem can be formulated in a probabilistic way: each pair $(e, e'), e \in \mathcal{E}, e' \in \mathcal{E}'$ is associated with a binary indicator variable $v_{(e,e')}$. $v_{(e,e')} = 1$ means (e, e') is an aligned pair, and $v_{(e,e')} = 0$ otherwise. Given some observed alignments $v_O = \{v_{(e,e')} = 1\}_{(e,e')\in\mathcal{O}}$, we aim to predict the labels of the remaining hidden entity pairs $\mathcal{H} = \mathcal{E} \times \mathcal{E}' \setminus \mathcal{O}$, i.e., $v_H = \{v_{(e,e')}\}_{(e,e')\in\mathcal{H}}$.

100 101

102

106

074

075

076

077

078

079 080

081

082

084

085

091

2.2 Symbolic reasoning for entity alignment

Given an aligned pair (e_j, e'_j) , a new aligned pair (e_i, e'_i) can be inferred with confidence score $w_{p,p'}$ if they are each connected to the existing pair via a relational path p and p' respectively, formally:

$$w_{p,p'}: \quad (e_j \equiv e'_j) \land p(e_i, e_j) \land p'(e'_i, e'_j) \Longrightarrow (e_i \equiv e'_i), \tag{1}$$

107 where $p = |\mathcal{R}|^L$, $p' = |\mathcal{R}'|^L$ are a pair of paths each consisting of L connected relations, and $w_{p,p'}$ measures the rule quality that considers the intra-KG structure and inter-KG structure, such as the



Figure 1: Framework illustration of NeuSymEA. The yellow solid line represents the alignment of an anchor pair. The symbolic model computes the matching probability of entity pairs by mining supporting rules (path pairs from anchor pairs) and evaluating their corresponding weights. The neural model learns embeddings and calculates entity-level matching scores based on embedding similarity. NeuSymEA models the agreement between the symbolic reasoning and neural representations using a joint probability distribution over observed pairs and parameterized truth scores for hidden pairs, optimized through a variational EM algorithm.

129

133

138 139

140

147

121

indicative of each path, and the similarity between two paths. By instantiating such *rule* with the constants (real entities and relations) in the KG pair, a symbolic model predicts the label distribution of an entity pair (e, e') by:

$$p_w(\boldsymbol{v}_{(e,e')}|\mathcal{G},\mathcal{G}'), \quad \text{for } (e,e') \in \{\mathcal{O} \cup \mathcal{H}\}.$$
 (2)

Using logic rules to infer the alignment probability can leverage the high-order structural information for effective alignment as well as provide interpretability. However, exact inference is intractable due to the massive amount of possible instantiated rules (exponential to *L*), limiting its applicability to real-world KGs.

3 NEURO-SYMBOLIC REASONING FRAMEWORK FOR ENTITY ALIGNMENT

141 3.1 VARIATIONAL EM142

Given a set of observed labels v_O , our goal is to maximize the log-likelihood of these labels, i.e., log $p_w(v_O)$. Directly optimizing this objective is intractable because it requires computing an integral over all the hidden variables. Instead, we optimize the evidence lower bound (ELBO) of the log-likelihood as follows:

$$p_w(\boldsymbol{v}_O) \ge E_{q(\boldsymbol{v}_H)} \left[\log p_w(\boldsymbol{v}_O, \boldsymbol{v}_H) - \log q(\boldsymbol{v}_H) \right] = \text{ELBO}(q, \boldsymbol{v}_O; w), \tag{3}$$

here, $q(v_H)$ is a variational distribution of the hidden variables v_H . This inequality holds for all qbecause $p_w(v_O) = \text{ELBO}(q, v_O; w) + D_{KL}(q(v_H) || p_w(v_H | v_O))$, where $D_{KL}(q(v_H) || p_w(v_H | v_O))$ $v_O) \ge 0$ is the KL-divergence between $q(v_H)$ and $p_w(v_H | v_O)$. Under this framework, the log-likelihood $p_w(v_O)$ can be optimized using an EM algorithm, an efficient method to find the maximum likelihood where the model depends on unobserved hidden variables: during the E-step, we fix w and update the variational distribution q; during the M-step, we update w to maximize the log-likelihood of all the entity pairs, i.e., $E_{q(v_H)}[\log p_w(v_O, v_H)]$, as illustrated in Figure 1.

Explicitly representing the variational distribution q is parameter intensive, which requires $\approx |\mathcal{E}||\mathcal{E}'|$ variables because the observed pairs are very sparse. To this end, we parameterize q with a neural model as q_{θ} , with θ being the parameters of the neural model.

158

160

- 159 3.2 E-STEP: INFERENCE
- 161 In this step, we fix w and update q_{θ} to minimize the KL divergence D_{KL} . Directly minimizing the KL divergence is intractable, as it involves computing the entropy of q_{θ} . Therefore, we follow

162 Qu & Tang (2019) and optimize the reverse KL divergence of q_{θ} and p_{w} , leading to the following objective:

$$\phi_{\boldsymbol{v}_H,\theta} = \sum_{(e,e')\in\mathcal{H}} \mathbf{E}_{p_w(\boldsymbol{v}_{(e,e')}|\boldsymbol{v}_O)} q_\theta(\boldsymbol{v}_H).$$
(4)

To optimize this objective, we first use the symbolic model with weighted rules to predict $p_w(v_{(e,e')} | v_O)$ for each $(e, e') \in \mathcal{H}$. If $p_w(v_{(e,e')} | v_O) > \delta$, where δ is a threshold, we treat this entity pair as a positive label; otherwise, we regard the pair as a negative pair that can be selected during negative sampling process of the neural model.

The observed labels can also be used as training data for supervised optimization. The objective is:

$$\phi_{\boldsymbol{v}_O,\theta} = \sum_{(e,e')\in\mathcal{O}} \log q_{\theta}(\boldsymbol{v}_{(e,e')} = 1).$$
(5)

174 175 176

173

171 172

165

The final objective for q_{θ} is obtained by combining these two objectives: $\phi_{\theta} = \phi_{v_{H,\theta}} + \phi_{v_{O,\theta}}$.

177 3.3 M-STEP: RULE WEIGHT UPDATE

In this step, we fix q_{θ} and update the rule weight w to maximize $\text{ELBO}(q, v_O; w)$. Since the right term of the ELBO in equation 3 is constant when q_{θ} is fixed, the objective is equivalent to maximizing the left term $E_{q_{\theta}(v_H)}[\log p_w(v_O, v_H)]$, which is the log-likelihood function.

Specifically, we start by predicting the labels of hidden variables using the current neural model. For each $(e, e') \in \mathcal{H}$, we predict the labels $\hat{v}_{(e,e')}(\theta)$ and obtain the prediction set $\hat{v}_H(\theta) = \{\hat{v}_{(e,e')}(\theta)\}_{(e,e')\in\mathcal{H}}$. In this way, maximizing the likelihood practically becomes maximizing the following objective:

$$\phi_w = \log p_w(\boldsymbol{v}_O, \hat{\boldsymbol{v}}_H(\theta)). \tag{6}$$

To obtain the pseudo-label $\hat{v}_{(e,e')}$ using q_{θ} , we employ the trained neural model to compute the matching score of any entity pair $(e,e') \in \mathcal{H}$. However, this strategy can easily introduce false positives into the pseudo-label set especially when the number of entities is large. To mitigate this, we consider one-to-one matching to sift only the most confident pairs. Practically, we first sort all pairs by their confidence score, then we annotate the pairs as positive following the order of the confidence. If a pair contains an entity observed in the annotated pairs, then this pair is skipped. This simple greedy strategy significantly reduces the amount of false positives.

194 195

196 197

199

200

206

207 208 209

186

4 OPTIMIZATION AND INFERENCE

4.1 EFFICIENT OPTIMIZATION VIA LOGICAL DEDUCTION

Inference and learning with logic rules of length L can be computationally intensive, as the search space for paths grows exponentially with increasing L. To enhance reasoning efficiency, we decompose a rule in equation 1 using logic deduction, inspired by Cheng et al. (2023) in KG completion:

$$w_{p,p'}: \quad (e_j \equiv e'_j) \land \left(\bigwedge_{k=1}^L r_k(e_{k-1}, e_k)\right) \land \left(\bigwedge_{k=1}^L r'_k(e'_{k-1}, e'_k)\right) \implies (e_i \equiv e'_i). \tag{7}$$

Here $\bigwedge_{k=1}^{L} r_k(e_{k-1}, e_k)$ represents the path formed by $r_1, r_2, ..., r_L$ connecting e_i to e_j with $e_0 = e_i$ and $e_k = e_j$. This can be reorganized as *a series of single-step logic reasoning*:

$$w_{p,p'}: \quad (e_j \equiv e'_j) \land \left(\bigwedge_{k=1}^L \left[r_k(e_{k-1}, e_k) \land r'_k(e'_{k-1}, e'_k) \right] \right) \implies (e_i \equiv e'_i). \tag{8}$$

In this way, each logic rule of length L can be viewed as *a deductive combination of* L *short rules of length* 1. At each step, following Suchanek et al. (2012), we perform one-step inference to update $p_w(v_{(e,e')})$ for each $(e,e') \in \mathcal{H}$ by aggregating the alignment probability from neighbors:

214
$$1 - \prod_{\substack{(e,r,e_t) \in \mathcal{T}, \\ (e',r',e'_t) \in \mathcal{T}'}} \left(1 - \eta(r) p_{sub}(r \subseteq r') p_w(\boldsymbol{v}_{(e_t,e'_t)})\right) \times \left(1 - \eta(r') p_{sub}(r' \subseteq r) p_w(\boldsymbol{v}_{(e_t,e'_t)})\right).$$
(9)

where $\eta(r)$ is a relation pattern of r measuring the uniqueness of e through relation r given a specified tail entity e_t , quantified by $\eta(r) = \frac{|\{e_t|(e_h, r, e_t) \in \mathcal{T}\}}{|\{(e_h, r, e_t) \in \mathcal{T}\}|}$. $p_{sub}(r \subseteq r')$ denotes the probability that relation r is a subrelation of r'. This technique enables inference with confidence by explicitly quantifying confidence w during each inference step by introducing η and $p_{sub}(r \subseteq r')$. Moreover, in this way, the update of the weight w simplifies to updating $p_{sub}(r \subseteq r')$ during the M-step (equation 6), as $\eta(r)$ for each relation r is constant. In practice, the update of $p_{sub}(r \subseteq r')$ can be computed by:

$$\frac{\sum \left(1 - \prod_{(e'_h, r', e'_t) \in \mathcal{T}'} \left(1 - \boldsymbol{v}_{(e_h, e'_h)} \boldsymbol{v}_{(e_t, e'_t)}\right)\right)}{\sum \left(1 - \prod_{e'_h, e'_t \in \mathcal{E}'} \left(1 - \boldsymbol{v}_{(e_h, e'_h)} \boldsymbol{v}_{(e_t, e'_t)}\right)\right)}.$$
(10)

where $v_{(e_h, e'_h)}$ and $v_{(e_t, e'_t)}$ are labels (or pseudo-labels) from $v_O \cup \hat{v}_H(\theta)$.

After optimization, rule weights can be computed by taking the product of the importance scores η of relations and the sub-relation probabilities of the corresponding relation pair:

$$w_{p,p'} \coloneqq \prod_{k=1}^{L} \eta(r_k) \cdot \eta(r'_k) \cdot \frac{p_{sub}(r_k \subseteq r'_k) + p_{sub}(r'_k \subseteq r_k)}{2}.$$
(11)

4.2 INFERENCE WITH INTERPRETABILITY

To predict new alignments, there are two approaches: using the symbolic model or the neural model. The symbolic model infers alignment probabilities with the optimized weights w. Due to scalability concerns, symbolic methods generally adopt a lazy inference strategy that only preserves the confident pairs implied by the neighbor structure during inference. On the other hand, the neural model computes similarity scores for all entity pairs $(e, e') \in \mathcal{H}$ using the learned parameters θ , generating a ranked candidate list for each entity.

The evaluation of these models thus differs. Symbolic models are generally evaluated by precision,
recall, and F1-score for their binary outputs, while neural models are assessed using hit@k and
mean reciprocal ranks (MRR) for their ranked candidate lists. Following the practices in Qi et al.
(2021) and Liu et al. (2022), we unify the evaluation metrics by treating the recall metric of symbolic
models as equivalent to hit@1, facilitating comparison with neural models.

To enhance the interpretability of predictions, we adapt the optimized symbolic model into an explainer. For any given entity pair, the explainer generates a set of supporting rule path pairs that justify their alignment, each associated with a confidence score indicating its significance. The explainer operates in two modes: (1) hard-anchor mode, which generates supporting paths only from prealigned pairs, and (2) soft-anchor mode, which includes paths from both prealigned and inferred pairs, providing more informative interpretations.

By integrating a breadth-first search algorithm (detailed in Appendix A.3), the explainer efficiently
generates high-quality interpretations. For truly aligned pairs, it typically produces high-confidence
interpretations, while for non-aligned pairs, the interpretations may result in an empty set (indicating
no supporting evidence) or have low confidence scores. See Figure 3 for a visualized comparison.

258 259

260

262

224 225 226

228

229

236

5 EXPERIMENTS

261 5.1 EXPERIMENTAL SETTINGS

In this section, we conduct experiments to answer the following questions. RQ1: Can NeuSymEA
 outperform existing neural, symbolic, and neuro-symbolic methods in terms of alignment performance? RQ2: Can symbolic and embedding-based methods complement each other in our framework? RQ3: How does the incorporated embedding-based model affect the alignment performance?
 RQ4: How does NeuSymEA interpret the inferred entity pairs, and how is the effectiveness of interpretations with respect to the rule length?

Datasets. We utilize the multilingual DBP15K dataset, which consists of three cross-lingual knowledge graph (KG) pairs: ja-en, fr-en, and zh-en. Note that the original full version (Sun et al., 2017) of this dataset contains a significant number of long-tail entities, presenting challenges for GCN-based models in terms of sparsity and large size. Therefore, many recent works (Wang et al., 2018; Mao et al., 2020; Liu et al., 2022) employ a condensed version, where long-tail entities and their connected triples are removed. For a comprehensive evaluation, we use both versions. Detailed statistics for these datasets are provided in Appendix B.1. Each dataset is divided into training, validation, and test sets following a 5-fold cross-validation scheme, with a ratio of 2:1:7.

Baselines and metrics. Baseline models include six neural models - GCNAlign (Wang et al., 2018), AlignE, BootEA (Sun et al., 2018), RREA (Mao et al., 2020), Dual-AMN (Mao et al., 2021), LightEA (Mao et al., 2022), one symbolic models - PARIS (Suchanek et al., 2012), and two neuro-symbolic models - PRASE (Qi et al., 2021), EMEA (Liu et al., 2022). We use Hit@1, Hit@10, and MRR as the evaluation metrics. For PARIS and PRASE that have binary outputs, we report their recall as Hit@1, following Liu et al. (2022).

Hyperparameters. NeuSymEA has two key hyperparameters: the number of EM iterations and the threshold δ for selecting positive pairs from the symbolic model. We tune these hy-perparameters and select the best values based on the validation set. The search space for δ is $\{0.6, 0.7, 0.8, 0.9, 0.95, 0.98, 0.99\}$, while the number of iterations is searched from 1 to 9.

5.2 RESULTS

COMPARISON WITH BASELINES 5.2.1

Table 1: Entity alignment results on DBP15K dataset. The suffixes "-D" and "-L" indicate the use of Dual-AMN and LightEA as the neural models. The results of RREA and EMEA are omitted on the full dataset due to an OOM (Out of Memory) error.

Category	Model	Hit@1	ja-en Hit@10	MRR	Hit@1	fr-en Hit@10	MRR	Hit@1	zh-en Hit@10	MRR
Group 1: Results on the full DBP15K dataset										
Neural	GCNAlign	0.221	0.461	0.302	0.205	0.475	0.295	0.189	0.438	0.271
	BootEA	0.454	0.782	0.564	0.443	0.799	0.564	0.486	0.814	0.600
	AlignE	0.356	0.715	0.476	0.346	0.731	0.475	0.333	0.690	0.453
	Dual-AMN	0.627	0.883	0.717	0.652	0.908	0.744	0.650	0.884	0.732
	LightEA	0.736	0.894	0.793	0.782	0.919	0.832	0.725	0.874	0.779
symbolic	PARIS	0.589	-	-	0.618	-	-	0.603	-	-
Neuro-Symbolic	PRASE	0.611	-	-	0.647	-	-	0.652	-	-
Ours	NeuSymEA-D	0.806	0.942	0.855	0.827	0.952	0.871	0.801	0.925	0.843
	NeuSymEA-L	<u>0.781</u>	<u>0.907</u>	0.826	0.834	<u>0.937</u>	0.871	0.785	<u>0.894</u>	<u>0.825</u>
Group 2: Results on the condensed DBP15K dataset										
	GCNAlign	0.331	0.662	0.443	0.325	0.688	0.446	0.335	0.653	0.442
	BootEA	0.530	0.829	0.631	0.579	0.872	0.961	0.575	0.847	0.668
Normal	AlignE	0.433	0.783	0.552	0.457	0.821	0.580	0.474	0.806	0.587
Neural	RREA	0.749	0.935	0.818	0.797	0.958	0.859	0.762	0.938	0.827
	Dual-AMN	0.750	0.927	0.815	0.793	0.954	0.854	0.756	0.919	0.816
	LightEA	0.778	0.911	0.828	0.827	0.943	0.830	0.770	0.894	0.816
symbolic	PARIS	0.565	-	-	0.584	-	-	0.543	-	-
Neuro-Symbolic	PRASE	0.580	-	-	0.622	-	-	0.593	-	-
	EMEA	0.736	-	0.807	0.773	-	0.841	0.748	-	0.815
0	NeuSymEA-D	0.805	0.930	0.849	0.835	0.953	0.879	0.815	0.926	0.855
Ours	NeuSymEA-L	$\frac{0.000}{0.811}$	$\frac{0.928}{0.928}$	0.854	0.858	0.954	0.894	0.804	0.904	0.840
	······					<u></u>		<u></u>		

To answer RQ1, RQ2 and RQ3, we compare NeuSymEA with baseline models on two versions of the DBP15K dataset: the full version (group1) and the condensed version (group2). Results are presented in Table 1. The results for PRASE, and EMEA on the condensed DBP15K are adopted from the original EMEA paper. We employ both Dual-AMN and LightEA as the neural models in our framework, denoted as NeuSymEA-D and NeuSymEA-L, respectively. These results lead to several key observations:

First, NeuSymEA surpasses both symbolic and KG embedding-based models. This can be attributed to the fact that it combines the capacity of both sides: 1) the ability to precisely infer confident pairs with rules by leveraging the multi-hop relational structures; and 2) the ability to effectively model entity representations in a unified space to jointly optimize the alignment of two KGs. NeuSymEA seamlessly combines both symbolic reasoning and neural representations in a principled variational framework, leading to improved performance.

330 Second, NeuSymEA outperforms both neuro-symbolic models, PRASE and EMEA. This im-331 provement can be largely attributed to the model objective design in our framework. While PRASE 332 and EMEA treat the symbolic and neural models as separate components, NeuSymEA integrates 333 them within a principled variational EM framework, unifying their objective as the log-likelihood 334 of the observed variables. This approach allows for the joint optimization of both components, ensuring they work together to maximize the log-likelihood. By iteratively refining both components 335 through the EM algorithm, NeuSymEA achieves a more coherent and convergent solution, leading 336 to superior performance, as shown by the empirical results. 337

338 Finally, NeuSymEA demonstrates superior robustness across both versions of the DBP15K 339 dataset. Comparisons between two groups of results offer an interesting insight: embedding-based models experience significant performance degradation when moving from the condensed version 340 341 to the full version of DBP15K (e.g., MRR of Dual-AMN decreases from 0.815 to 0.717 on ja-en), while symbolic models, in contrast, show improvements. We attribute this to two key factors: (1) 342 Embedding-based models rely on entity-level matching, which is sensitive to dataset size. As the 343 dataset grows, the number of similar entity embeddings increases, leading to reduced accuracy. The 344 full version of DBP15K contains significantly more entities than the condensed version, exacer-345 bating this effect. (2) Symbolic models, on the other hand, perform path-level matching. Their 346 effectiveness is constrained more by substructure heterogeneity and sparsity than dataset size. The 347 full version of DBP15K includes more relational triples, which enhances the rule-mining process, 348 ultimately making the symbolic models more effective in this scenario. NeuSymEA, by combin-349 ing symbolic reasoning with KG embeddings, mitigates the shortcomings of both approaches, 350 making it robust to changes in dataset scale and structure. This synergy allows NeuSymEA to 351 consistently outperform baselines in both sparse and dense settings.

5.2.2 EVOLUTION OF RULES AND EMBEDDINGS

352 353

354

355 356

357

359

360

361

362

364

365

366 367

368

369

370

371

372

373



Figure 2: (Left) Evolution of rule inferred pairs, with solid lines representing total inferred pairs and dashed lines representing true inferred. The shaded areas indicate the number of false pairs. Precision values are annotated at each data point. (Right) Convergence of MRR of the neural model.

We study how rules and embeddings evolve and interact with each other during the EM steps, with results shown in Figure 2. Results in the left subplot indicate that in each EM iteration, the number of rule-inferred pairs grows consistently with high precision, implying that the embedding model continuously improves the inference performance of rules. These precise pairs, in turn, enhance the performance of the neural model. As shown in the right subfigure, the MRR of the neural model converges within a few iterations.

5.2.3 INTERPRETATIONS BY THE EXPLAINER375

We investigate the interpretations generated by the explainer on the fr-en dataset. The left subfigure of Figure 3 shows the probability density of confidence scores for supporting rules (with a maximum rule length of 3) associated with entity pairs. Positive pairs are derived from the test set, while



Figure 3: (Left) Probability density of the top supporting rule's confidence; (Middle) Number of supporting rules (thresholded by confidence score) relative to the maximum rule lengths under the soft anchor mode; (Right) Number of supporting rules relative to the maximum rule lengths under the hard anchor mode.

Table 2: Examples of supporting rules for query pairs in fr-en. Anchor pairs are shown in bold.

Query Pair	Supporting Rule	Confidence
Maison_de_Savoie House_of_Savoy	(Humbert_III_(roi_d'Italie), dynastie, Maison_de_Savoie), (Humbert_III_(roi_d'Italie), conjoint, Marie-José_de_Belgique) (Umberto_II_of_Italy, house, House_of_Savoy), (Umberto_II_of_Italy, spouse, Marie_José_of_Belgium)	0.80
Légion_espagnole Spanish_Legion	(Légion_espagnole, commandantHistorique, Francisco_Franco), (Francisco_Franco, conjoint, Carmen_Polo) (Spanish_Legion, notableCommanders, Francisco_Franco), (Francisco_Franco, spouse, Carmen_Polo1st_Lady_of_Meirás)	0.59
Premier_ministre_du_Danemark Prime_Minister_of_Denmark	(Premier_ministre_du_Danemark, titulaireActuel, Lars_Løkke_Rasmussen) (Prime_Minister_of_Denmark, incumbent, Lars_Løkke_Rasmussen)	0.79

negative pairs are created by replacing one entity in each test pair with another randomly sampled entity. The distinct confidence distributions indicate that positive pairs generally have more evidence for alignment, which aligns with intuition. However, the probability density distribution also reveals that some positive pairs do not have high confidence scores. Upon further examination, we found that many test pairs are isolated, i.e., they lack directly aligned neighbors. Despite this, 405 NeuSymEA successfully generates supporting rules for isolated pairs by exploiting multihop 406 dependencies. In Table 2, we provide several examples of supporting rules and their associated confidence scores for the queried entity pairs. 408

To examine the impact of rule length on the explainer's effectiveness, the middle and right subfigures 409 in Figure 3 show the number of supporting rules for test positive pairs as the maximum rule length 410 increases. Compared to hard anchor mode, the explainer in soft anchor mode generates more high-411 quality supporting rules by leveraging inferred pairs as complementary anchor pairs, mitigating the 412 sparsity issue. We also observe that increasing the maximum rule length leads to more high-413 quality rules; however, the number of high-confidence rules grows more slowly than lower-414 confidence rules. This can be attributed to our method for calculating confidence: the logical 415 deduction-based approach computes a rule's confidence as the product of the confidences of its 416 decomposed length-one sub-rules (as in Equation (11)). For example, a rule with two sub-rules, 417 each with confidence 0.8, results in an overall confidence of $0.8 \times 0.8 = 0.64$. Considering this, the confidence score tends to decrease when the rule length increases, thus increasing the maximum 418 length tends to discover supporting rules with lower scores. 419

420 421

422

5.2.4 **ROBUSTNESS IN LOW RESOURCE SCENARIO**

423 Figure 5 demonstrates the model performance under low-resource settings. As the percentage of 424 training data decreases, all models experience noticeable drops in Hit@1 performance. Despite this, 425 NeuSymEA (both NeuSymEA-L and NeuSymEA-D) exhibits remarkable robustness across all 426 datasets, consistently outperforming other models even with limited data. Notably, with only 427 1% of pairs used as training data, NeuSymEA-L achieves a Hit@1 score exceeding 0.7 on fr-en, 428 rivaling or even surpassing the performance of some state-of-the-art models trained on 20% of the 429 data. While models such as LightEA, Dual-AMN, and BootEA show improvement with additional training data, they still fall short of NeuSymEA. In contrast, traditional models like GCNAlign 430 and AlignE demonstrate significant struggles in low-resource settings, underscoring their heavier 431 dependence on larger training datasets.

393 394

401

402

403

404

407

386

387



Figure 4: Alignment performance with varying percentages of pairs as training data. For brevity, we present only the Hit@1 metric, with comprehensive results available in Appendix B.2.

6 RELATED WORK

Neuro-symbolic reasoning on knowledge graphs. Neuro-symbolic methods aim to merge sym-452 bolic reasoning with neural representation learning, leveraging the precision and interpretability of 453 symbolic approaches alongside the scalability and high recall of neural methods. In KG completion 454 task, Guo et al. (2016) and Guo et al. (2018) employ horn rules to regularize the learning of KG 455 embeddings; Cheng et al. (2022) and Cheng et al. (2023) model the rule-based predictions as dis-456 tributions conditioned on the input relational sequences, and parameterize these distributions using 457 a recurrent neural network; Qu et al. (2020) and Cheng et al. (2022) enhances rule grounding by 458 augmenting the fact set using a pre-trained KG-embedding model; Qu & Tang (2019), Zhang et al. 459 (2020) and Chen et al. (2024a) models the joint probability of neural model and the symbolic model 460 with a Markov random field, and employ gradient descent for weight updates. Despite extensive 461 advancements of neuro-symbolic reasoning in KG completion(DeLong et al., 2024; Cheng et al., 462 2024), these studies only consider single-KG structures, thus cannot be directly adopted to entity 463 alignment which requires consideration of both intra-KG and inter-KG structures.

464 **Entity alignment.** Recent models have sought to combine symbolic and neural approaches for 465 entity alignment. For instance, Qi et al. (2021) enhances probabilistic reasoning by utilizing KG 466 embeddings, employing a KG-embedding model to measure similarities during both updating and 467 inference processes. Liu et al. (2022) implements self-bootstrapping with pseudo-labeling in a neu-468 ral framework, using rules to choose confident pseudo-labels. However, it relies solely on unit-length rules, which restricts its effectiveness for long-tail entities. Recently, Tian et al. (2024) proposes to 469 generate interpretations to explain entity alignment, but their interpretations are subgraphs extracted 470 by semantic matching using a pre-trained neural model. 471

In contrast, our work measures agreement between symbolic and neural models by modeling joint
probability within a Markov random field, optimizing both components toward a unified objective
within a variational EM framework. We employ logic deduction to scale reasoning with long rules
of any length. These learned rules provide interpretations during inference.

476 477

478

446

447

448 449

450 451

7 CONCLUSIONS

In this work, we presented NeuSymEA, a neuro-symbolic framework for entity alignment that effectively integrates symbolic methods with neural approaches within a principled variational EM framework. By unifying these traditionally distinct methodologies, NeuSymEA addresses the challenges of substructure heterogeneity, sparsity, and uncertainty. Our empirical results demonstrate that NeuSymEA not only outperforms existing approaches but also provides interpretable alignment predictions through its path-ranking-based explainer. These findings underscore the potential of this unified neuro-symbolic framework to advance knowledge fusion by enabling effective entity alignment with uncertainty-aware interpretations.

486 REFERENCES

- Hannah Bast and Elmar Haussmann. More accurate question answering on freebase. In Proceedings
 of the 24th ACM international on conference on information and knowledge management, pp. 1431–1440, 2015.
- 491 Rose Catherine and William Cohen. Personalized recommendations using knowledge graphs: A
 492 probabilistic logic programming approach. In Proceedings of the 10th ACM conference on
 493 recommender systems, pp. 325–332, 2016.
- Muhao Chen, Yingtao Tian, Mohan Yang, and Carlo Zaniolo. Multilingual knowledge graph embeddings for cross-lingual knowledge alignment. In Proceedings of the 26th International Joint Conference on Artificial Intelligence, pp. 1511–1517, 2017.
- Shengyuan Chen, Yunfeng Cai, Huang Fang, Xiao Huang, and Mingming Sun. Differentiable neuro symbolic reasoning on large-scale knowledge graphs. <u>Advances in Neural Information Processing</u>
 Systems, 36, 2024a.
- Shengyuan Chen, Qinggang Zhang, Junnan Dong, Wen Hua, Qing Li, and Xiao Huang. Entity alignment with noisy annotations from large language models. <u>arXiv preprint arXiv:2405.16806</u>, 2024b.
- Keiwei Cheng, Nesreen K Amed, and Yizhou Sun. Neural compositional rule learning for knowl edge graph reasoning. In <u>International Conference on Learning Representations</u>, 2023.
- Kewei Cheng, Jiahao Liu, Wei Wang, and Yizhou Sun. Rlogic: Recursive logical rule learning
 from knowledge graphs. In Proceedings of the 28th ACM SIGKDD Conference on Knowledge
 Discovery and Data Mining, pp. 179–189, 2022.
- Kewei Cheng, Nesreen K Ahmed, Ryan A Rossi, Theodore Willke, and Yizhou Sun. Neural symbolic methods for knowledge graph reasoning: A survey. <u>ACM Transactions on Knowledge</u>
 Discovery from Data, 2024.
- Lauren Nicole DeLong, Ramon Fernández Mir, and Jacques D Fleuriot. Neurosymbolic ai for reasoning over knowledge graphs: A survey. <u>IEEE Transactions on Neural Networks and Learning Systems</u>, 2024.
- Junnan Dong, Qinggang Zhang, Xiao Huang, Keyu Duan, Qiaoyu Tan, and Zhimeng Jiang.
 Hierarchy-aware multi-hop question answering over knowledge graphs. In Proceedings of the
 ACM Web Conference 2023, pp. 2519–2527, 2023.
- Shu Guo, Quan Wang, Lihong Wang, Bin Wang, and Li Guo. Jointly embedding knowledge graphs and logical rules. In <u>Proceedings of the 2016 conference on empirical methods in natural language</u> processing, pp. 192–202, 2016.
- Shu Guo, Quan Wang, Lihong Wang, Bin Wang, and Li Guo. Knowledge graph embedding with iterative guidance from soft rules. In <u>Proceedings of the AAAI Conference on Artificial Intelligence</u>, volume 32, 2018.
- Ernesto Jiménez-Ruiz and Bernardo Cuenca Grau. Logmap: Logic-based and scalable ontology
 matching. In Proceedings of the 10th International Semantic Web Conference, pp. 273–288,
 2011.
- Zhixun Li, Xin Sun, Yifan Luo, Yanqiao Zhu, Dingshuo Chen, Yingtao Luo, Xiangxin Zhou, Qiang
 Liu, Shu Wu, Liang Wang, et al. GSLB: the graph structure learning benchmark. <u>Advances in</u> Neural Information Processing Systems, 36, 2024.
- Bing Liu, Harrisen Scells, Wen Hua, Guido Zuccon, Genghong Zhao, and Xia Zhang. Guiding neural entity alignment with compatibility. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pp. 491–504. Association for Computational Linguistics, 2022.
- Xin Mao, Wenting Wang, Huimin Xu, Yuanbin Wu, and Man Lan. Relational reflection entity alignment. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management, pp. 1095–1104, 2020.

- 540 Xin Mao, Wenting Wang, Yuanbin Wu, and Man Lan. Boosting the speed of entity alignment $10 \times :$ 541 Dual attention matching network with normalized hard sample mining. In Proceedings of the 542 Web Conference 2021, pp. 821–832, 2021. 543 Xin Mao, Wenting Wang, Yuanbin Wu, and Man Lan. LightEA: A scalable, robust, and inter-544 pretable entity alignment framework via three-view label propagation. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, 2022. 546 547 Zhiyuan Qi, Ziheng Zhang, Jiaoyan Chen, Xi Chen, Yuejia Xiang, Ningyu Zhang, and Yefeng 548 Zheng. Unsupervised knowledge graph alignment by probabilistic reasoning and semantic em-549 bedding. In Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, 2021. 550 551 Meng Qu and Jian Tang. Probabilistic logic neural networks for reasoning. In Advances in Neural 552 Information Processing Systems (NeurIPS), 2019. 553 554 Meng Qu, Junkun Chen, Louis-Pascal Xhonneux, Yoshua Bengio, and Jian Tang. Rnnlogic: Learning logic rules for reasoning on knowledge graphs. arXiv preprint arXiv:2010.04029, 2020. 555 556 Fabian M. Suchanek, Serge Abiteboul, and Pierre Senellart. Paris: Probabilistic alignment of rela-557 tions, instances, and schema. In Proceedings of the 38th International Conference on Very Large 558 Databases, pp. 157-168, 2012. 559 Zequn Sun, Wei Hu, and Chengkai Li. Cross-lingual entity alignment via joint attribute-preserving 560 embedding. In Proceedings of the 16th International Semantic Web Confernece, pp. 628-644. 561 Springer, 2017. 562 563 Zequn Sun, Wei Hu, Qingheng Zhang, and Yuzhong Qu. Bootstrapping entity alignment with knowledge graph embedding. In Proceedings of the 27th International Joint Conference on Artificial 565 Intelligence, number 2018, 2018. 566 Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. Arnetminer: extraction and 567 mining of academic social networks. In Proceedings of the 14th ACM SIGKDD international 568 conference on Knowledge discovery and data mining, pp. 990-998, 2008. 569 570 Xiaobin Tian, Zequn Sun, and Wei Hu. Generating explanations to understand and repair 571 embedding-based entity alignment. In 2024 IEEE 40th International Conference on Data Engineering (ICDE), pp. 2205–2217. IEEE, 2024. 572 573 Zhichun Wang, Qingsong Lv, Xiaohan Lan, and Yu Zhang. Cross-lingual knowledge graph align-574 ment via graph convolutional networks. In Proceedings of the 2018 conference on empirical 575 methods in natural language processing, pp. 349–357, 2018. 576 577 Gerhard Weikum and Martin Theobald. From information to knowledge: harvesting entities and relationships from web sources. In Proceedings of the twenty-ninth ACM 578 SIGMOD-SIGACT-SIGART symposium on Principles of database systems, pp. 65–76, 2010. 579 580 Yuyu Zhang, Xinshi Chen, Yuan Yang, Arun Ramamurthy, Bo Li, Yuan Qi, and Le Song. Efficient 581 probabilistic logic reasoning with graph neural networks. In International Conference on Learning 582 Representations, 2020. 583 584 585 586 587 588 589 590
- 501
- 592
- 593

A NOTATIONS AND ALGORITHMS

A.1 NOTATIONS

Notation	Description
\mathcal{G},\mathcal{G}'	The source and target knowledge graphs, respectively
$\mathcal{E}, \mathcal{E}'$	The sets of entities in \mathcal{G} and \mathcal{G}' , respectively
$\mathcal{R}, \mathcal{R}'$	The sets of relations in \mathcal{G} and \mathcal{G}' , respectively
\mathcal{T},\mathcal{T}'	The sets of relational triplets in \mathcal{G} and \mathcal{G}' , respectively
O	The set of observed aligned entity pairs between two knowledge graphs \mathcal{G} and \mathcal{G}'
\mathcal{H}	Set of unobserved entity pairs, i.e., $\mathcal{E} \times \mathcal{E}' \setminus \mathcal{O}$
$v_{(e,e')}$	Binary indicator variable for an entity pair (e, e') , where $v_{(e,e')} = 1$ indicates alignment
$w_{n,n'}$	Confidence score of a rule-inferred alignment based on paths p and p'
$p_w^{r,r}(\boldsymbol{v}_{(e,e')} \mathcal{G},\mathcal{G})$	") Probability distribution of the alignment indicator $v_{(e,e')}$ given knowledge graphs \mathcal{G} and \mathcal{G}'
θ	Parameters of the neural model
δ	Threshold to select positive pair from the symbolic model
n(r)	Relation pattern measuring the uniqueness of an entity through relation r

Table 3: Notations

A.2 COMPLEXITY ANALYSIS OF THE SYMBOLIC REASONING

In the following, we present the analysis of runtime complexity and parameter complexity one by one.

617 618 619

620

614 615

616

594

595 596

597 598

A.2.1 RUNTIME COMPLEXITY

In variational inference, the process of learning and inferring long rules (Eq. 7) is simplified by decomposing them into unit-length rules (Eq. 8). Consequently, rule weight learning (Eq. 10) is only conducted for unit-length rules. The inference process for an *L*-length rule is then estimated by iteratively applying inference steps with unit-length rules (Eq. 9) for *L* iterations. This strategy effectively avoids the exponential search space associated with longer rules, making the computational complexity of the inference linear with respect to the rule length *L*.

Each iteration of reasoning with unit-length rules comprises an inference step (Eq. 9) and a ruleweight learning step (Eq. 10). These steps require computing the matching probability for all possible entity pairs and relation pairs, respectively. As a result, the computational complexity of the inference step and the weight updating step are $O(|\mathcal{E}||\mathcal{E}'|)$ and $O(|\mathcal{R}||\mathcal{R}'|)$, respectively.

Thus, the total **computational complexity** for reasoning with an *L*-length rule is $O(L \cdot (|\mathcal{E}||\mathcal{E}'| + |\mathcal{R}||\mathcal{R}'|))$. Given that entity sizes are typically much larger than relation sizes, this complexity can be approximated as $O(L \cdot |\mathcal{E}||\mathcal{E}'|)$.

⁶³⁴ Notably, the computations involved in Eq. 9 and Eq. 10 can be accelerated through parallel ⁶³⁵ processing, which we have implemented. This optimization reduces the **runtime complexity** to ⁶³⁶ $O\left(L \cdot \frac{|\mathcal{E}||\mathcal{E}'|}{n}\right)$, where *n* represents the number of CPU cores available for parallelization.

638 639

640

A.2.2 PARAMETER COMPLEXITY

641 The total number of alignment probabilities for all entity pairs is $|\mathcal{E}||\mathcal{E}'|$, which is large when the 642 entity sizes increase. We adopt a lazy inference strategy to enhance parameter efficiency. This 643 strategy involves only saving the alignment probabilities of the most probable alignments:

$$\left\{ p_w(v_{(e_i,e_i')}), |, e_i \in \mathcal{E}, e_i' \in \mathcal{E}', p_w(v_{(e_i,e_i')}) = \max\left(\max_{e \in \mathcal{E}} p_w(v_{(e,e_i')}), \max_{e' \in \mathcal{E}'} p_w(v_{(e_i,e')})\right) \right\}$$
(12)

Probabilities of other entity pairs can be inferred from these saved alignment probabilities using Eq. 9. In this way, **parameter complexity** is reduced to $O(\max(|\mathcal{E}| + |\mathcal{E}'|))$.

648 A.3 PSEUDO-CODE OF EXPLAINER

653

674

675

676

677 678

679 680

682

683

Below is the pseudo-code of how the explainer generates supporting rules as interpretations for the
query pair. It consists of two stages: searching reachable anchor pairs, and parsing rule paths as well
as calculating rule confidences.

654 Algorithm 1 Generating Interpretations for the Queried Entity Pair with Weighted Rules

655 **Inputs:** Subrelation probabilities $p_{sub}(r \subseteq r'), p_{sub}(r' \subseteq r)$ for $r, r' \in \mathcal{R}$; Knowledge Graph pair $(\mathcal{G}, \mathcal{G}')$; Maximum rule length \mathcal{L} ; Anchor pairs \mathcal{A} with source-to-target mapping S2T and target-to-source mapping 656 T2S; Query entity pair (e_q, e'_q) 657 Outputs: Ranked rules based on confidence 658 1. Search Reachable Anchor Pairs within Max Depth \mathcal{L} 659 $RN \leftarrow \text{BFS}(e_q, \mathcal{G}, \mathcal{L})$ / \star Search reachable neighbors of e_q using breadth-first search, max depth \mathcal{L} \star / 660 $RN' \leftarrow BFS(e'_q, \mathcal{G}', \mathcal{L})$ /* Search reachable neighbors of e'_q using breadth-first search, max depth $\mathcal{L} * / \mathcal{L}$ $RN_a \leftarrow RN \cup T2S(RN'; \mathcal{A})$ 661 /* Find source nodes of reachable anchor pairs using hash mapping */ $RA \leftarrow \{(e, S2T(e; \mathcal{A})) \mid e \in RN_a\}$ /* Identify reachable anchor pairs */ 662 2. Parse and Rank Rules Based on Confidence 663 for $\forall (e, e') \in RA$ do Extract paths: $p(e, e_q) = r_1 \wedge r_2 \wedge \ldots, p'(e', e'_q) = r'_1 \wedge r'_2 \wedge \ldots$ 665 if $|p(e, e_q)| \neq |p'(e', e'_q)|$ then $/ \star$ If path lengths don't match, rule confidence is 0 $\star /$ 666 $w_{p(e,e_q),p'(e',e'_q)} \leftarrow 0$ else 667 $w_{p(e,e_q),p'(e',e_q')} \leftarrow \prod_{i=1}^{|p|} \eta(r_i) \cdot \eta(r_i') \cdot \frac{p_{sub}(r_i \subseteq r_i') + p_{sub}(r_i' \subseteq r_i)}{2}$ / * Compute rule confidence by 668 products of subrelation probabilities and relation functionalities */ 669 end if 670 end for 671 Sort the rules (p, p') by $w_{p,p'}$ in descending order 672 Return the ranked rules 673

A.4 COMPLEXITY ANALYSIS OF THE EXPLAINER

B EXPERIMENTAL DETAILS

B.1 DATASET STATISTICS

The DBP15K dataset is designed for cross-lingual knowledge graph alignment and comes in two versions: the full version and the condensed version. The full version of DBP15K includes comprehensive data across three language pairs. The condensed version eliminates the less frequent entities and their connected triples. Statistics of both versions are presented below.

Datasets	KG	Entities	Relations	Rel. Triplets	Aligned Entity Pairs
zh-en	Chinese (zh) English (en)	66,469 98,125	2,830 2,317	153,929 237,674	15,000
ja-en	Japanese (ja) English (en)	65,744 95,680	2,043 2,096	164,373 233,319	15,000
fr-en	French (fr) English (en)	66,858 105,889	1,379 2,209	192,191 278,590	15,000

Table 4: Data statistics of the full DBP15K dataset.

B.2 COMPREHENSIVE RESULTS WITH DIFFERENT RATIO OF TRAINING DATA

Below we present the comprehensive performance evaluation with various ratios of pairs as training data. In these 3x3 subfigures, each row shows the results of the same dataset, and each column shows the results of the same metric. To summarize, we have the following observations from Figure 5:

699 700

696 697

• Neural-symbolic models, particularly our proposed NeuSymEA (NeuSymEA-L and NeuSymEA-D), consistently outperform other approaches across all datasets and metrics.





Figure 5: Alignment performance with varying percentages of pairs as training data.

These models demonstrate remarkable robustness, maintaining high performance even with limited training data (as low as 1%). Their success highlights the effectiveness of combining neural and symbolic approaches in entity alignment tasks, offering a significant advantage over traditional neural or purely symbolic methods.

- The performance of all models generally improves as the percentage of training data increases, but the rate of improvement varies significantly. Traditional neural models like GCNAlign and AlignE show the steepest improvement curves, indicating their heavy reliance on large training datasets. In contrast, neural-symbolic models, especially our proposed NeuSymEA, demonstrate high performance even with minimal training data, showcasing their efficiency and potential for low-resource scenarios.
- The experiments reveal varying levels of difficulty in entity alignment across different language pairs. Japanese-English alignment generally shows the highest performance across models, followed by French-English, while Chinese-English proves to be the most challenging. This variance underscores the importance of considering language-specific characteristics in entity alignment tasks and suggests that some language pairs may require more sophisticated approaches or additional resources for effective alignment. The ob-

served differences might be attributed to factors such as linguistic distance, writing system differences, or the availability and quality of pre-existing resources for each language pair. These findings emphasize the importance of developing flexible models that can adapt to the specific challenges posed by different language combinations.

- The exceptional performance of NeuSymEA across various scenarios different language pairs, metrics, and data availability levels – points to their robustness and scalability. This consistency suggests that these models might be more easily adaptable to new languages or domains without requiring extensive retraining or modification. Such robustness is crucial for developing general-purpose entity alignment systems that can be deployed across diverse linguistic and domain-specific contexts. Additionally, its ability to perform well with limited data indicates potential for scalability to a wider range of languages and domains, including those with limited resources, which could significantly expand the reach and applicability of cross-lingual knowledge integration technologies.

B.3 PARAMETER ANALYSIS

We present the hit@1 performance of NeuSymEA across three datasets, varying hyperparameters, illustrated by a three-dimensional graph. The threshold hyperparameter δ is explored within the set {0.6, 0.7, 0.8, 0.9, 0.95, 0.98, 0.99}, while the number of EM iterations ranges from 1 to 9. Performance levels are indicated using a colormap. Performance sensitivity analysis in Figure 6 reveals that for all datasets, performance generally improves as the iteration increases. On the other hand, the performance is less sensitive to the threshold δ .



Figure 6: Performance sensitivity to hyperparameters iteration and threshold δ .