Wasserstein-Fisher-Rao Embedding: Logical Query Embeddings with Local Comparison and Global Transport

Zihao Wang[*†], **Weizhi Fei**[*♣], **Hang Yin**[♣], **Yangqiu Song** [†], **Ginny Y. Wong**[★], **Simon See**[★]
† CSE, HKUST, HKSAR, China
♣ Department of Mathematical Sciences, Tsinghua University, Beijing, China
★ NVIDIA AI Technology Center (NVATIC), NVIDIA, Santa Clara, USA
zwanggc@cse.ust.hk, {fwz22,hyin-20}@mails.tsinghua.edu.cn
yqsong@cse.ust.hk, {gwong,ssee}@nvidia.com

Abstract

Answering complex queries on knowledge graphs is important but particularly challenging because of the data incompleteness. Query embedding methods address this issue by learning-based models and simulating logical reasoning with set operators. Previous works focus on specific forms of embeddings, but scoring functions between embeddings are underexplored. In contrast to existing scoring functions motivated by *local* comparison or *global* transport, this work investigates the *local* and *global* trade-off with unbalanced optimal transport theory. Specifically, we embed sets as bounded measures in $\mathbb{R}$ endowed with a scoring function motivated by the Wasserstein-Fisher-Rao metric. Such a design also facilitates closed-form set operators in the embedding space. Moreover, we introduce a convolution-based algorithm for linear time computation and a block-diagonal kernel to enforce the trade-off. Results show that WFRE can outperform existing query embedding methods on standard datasets, evaluation sets with combinatorially complex queries, and hierarchical knowledge graphs. Ablation study shows that finding a better *local* and *global* trade-off is essential for performance improvement.[1]

1 Introduction

Knowledge graphs (KGs) store real-world factual knowledge as entity nodes and relational edges (Miller, 1995; Bollacker et al., 2008; Vrandečić and Krötzsch, 2014). And they facilitate many downstream tasks (Xiong et al., 2017a; Wang et al., 2019; Lin et al., 2020). Notably, answering complex logical queries is an essential way to exploit the knowledge stored in knowledge graphs (Ren et al., 2020, 2021).

Formally speaking, complex logic queries can be expressed via first-order logic (Ren et al., 2020; Marker, 2002). Specific groups of queries, whose predicates and logical connectives can be converted as set operations (Wang et al., 2021), are of particular interest due to their clear semantics. Therefore, the logical reasoning process to answering complex queries is transformed to execute set projections and operations in an operator tree (Ren et al., 2020; Wang et al., 2021). Figure 1 shows the operator tree for the query "Who is the non-American director that has won Golden Globes or Oscar".

What makes this task difficult is the data incompleteness of knowledge graphs. Modern large-scale KGs are naturally incomplete because they are constructed by crowdsource (Bollacker et al., 2008; Vrandečić and Krötzsch, 2014) or automatic information extraction pipelines (Carlson et al., 2010). This issue is acknowledged as the Open World Assumption (Libkin and Sirangelo, 2009) (OWA). It leads to the fact that applying query answering algorithms for complete databases will not result in complete answers because of the data incompleteness. Also, it is not able to prune the search space with the observed incomplete knowledge graph, which results in a large computational cost (Ren et al., 2020). It makes the problem even harder when answering logical queries on large knowledge graphs with billions of edges (Ren et al., 2022). We refer readers to recent surveys for more about logical queries on knowledge graphs (Wang et al., 2022b; Ren et al., 2023).

Query embedding methods (Hamilton et al., 2018; Ren et al., 2020) in fixed dimensional spaces are proposed to overcome the above difficulties. The data incompleteness is addressed by generalizing learnable set embeddings and operators to unseen data (Ren et al., 2020; Ren and Leskovec, 2020). And the computation cost does not grow with the Developing efficient forms of set embeddings and operators is one of the recent focuses (Choudhary et al., 2021a; Zhang et al., 2021; Alivanistos et al., 2022; Bai et al., 2022; Chen

[*] Equal Contribution

[1] Our implementation can be found at https://github.com/HKUST-KnowComp/WFRE.

Findings of the Association for Computational Linguistics: ACL 2023, pages 13679–13696
July 9-14, 2023 ©2023 Association for Computational Linguistics

Natural Language: Find non-American directors whose movie won Golden Globes or Oscar?
Logical Formula: $q = V_? \exists V_1 . (\mathrm{Won}(V_1, \mathrm{GoldenGlobes}) \vee \mathrm{Won}(V_1, \mathrm{Oscar})) \wedge \neg \mathrm{BornIn}(V_?, \mathrm{America}) \wedge \mathrm{Direct}(V_?, V_1)$
Set Operator Tree: $\mathrm{DirectorOf}(\mathrm{WinnerOf}(\mathrm{GoldenGlobes}) \cup \mathrm{WinnerOf}(\mathrm{Oscar})) \cap \mathrm{BornIn}(\mathrm{America})^C$

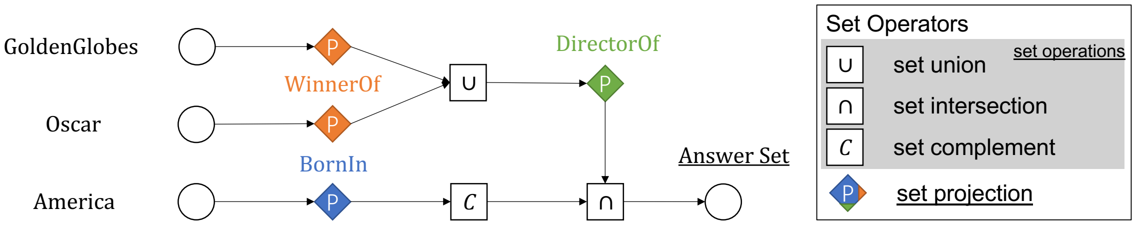


Figure 1: Answering logical queries on knowledge graphs. Natural language sentences can be interpreted as logical formulas and then converted to set operator trees (Wang et al., 2021).

et al., 2022; Yang et al., 2022). However, the scoring function between sets, though it also characterizes set embeddings and plays a vital role in training models, is underexplored in the existing literature. Existing scoring functions are chosen from two categories that emphasize either *local* comparisons (Ren and Leskovec, 2020; Amayuelas et al., 2022) or *global* transport between geometric regions (Ren et al., 2020; Choudhary et al., 2021b; Zhang et al., 2021). The following example motivated us to develop scoring functions for embeddings with both *local* and *global* trade-off.

Example 1.1. Consider four "one-hot" vectors with dimension $d = 100$:

$$A = [1, 0, 0, ..., 0], \tag{1}$$

$$B = [0, 1, 0, ..., 0], \tag{2}$$

$$C = [0, 0, 1, 0, ..., 0], \tag{3}$$

$$D = [0, \ldots, 0, 1]. \tag{4}$$

We observe that:

- Local function (e.g., Euclidean distance) L CANNOT discriminate different similarities between A, B, C, and D. Specifically, $L(A, B) = L(A, C) = L(A, D) = L(B, C) = L(B, D) = L(C, D) = 1$.
- Global function (e.g. Wasserstein metric) G CAN discriminate. Specifically, $G(A, B) = 1 < G(A, C) = 2 < G(A, D) = 99$. However, G is risky for optimization. For example, if $G(A, D) + G(A, B)$ appears in the objective function of a batch, $G(A, D)$ will dominate $G(A, B)$ because it is 100 times larger, making the optimization ineffective.
- Local and global trade-off function (such as the WFR scoring function proposed in this paper) harnesses this risk by constraining the transport within a window size. Our paper finds that the proper window size is 5, which truncated the transport distances between faraway samples like A and D. Then, $WFR(A, D) = 5$, and the optimization is stabilized.

In this paper, we develop a more effective scoring function motivated by the Wasserstein-Fisher-Rao (WFR) metric (Chizat et al., 2018a), which introduces the *local* and *global* trade-off. We propose to embed sets as Bounded Measures in $\mathbb{R}$, where each set embedding can be discretized as a bounded histogram on uniform grids of size d. This set embedding can be interpreted *locally* so that the set intersection, union, and negation can be easily defined by element-wise fuzzy logic t-norms (Hájek, 1998). We propose an efficient convolution-based algorithm to realize the computation of entropic WFR in $O(d)$ time, and a block diagonal kernel to enforce the *local* and *global* trade-off. We conduct extensive experiments on large number of datasets: (1) standard complex query answering datasets over three KGs (Ren and Leskovec, 2020), (2) large-scale evaluation set emphasizing the combinatorial generalizability of models in terms of compositional complex queries (Wang et al., 2021), and (3) complex queries on a hierarchical knowledge graph (Huang et al., 2022). Ablation studies show that the performance of complex query answering can be significantly improved by choosing a better trade-off between *local* comparison and *global* transport.

2 Related Works

We discuss other query embedding methods in fixed dimensions and optimal transport in this section. Other methods for complex query answering are discussed in Appendix A,

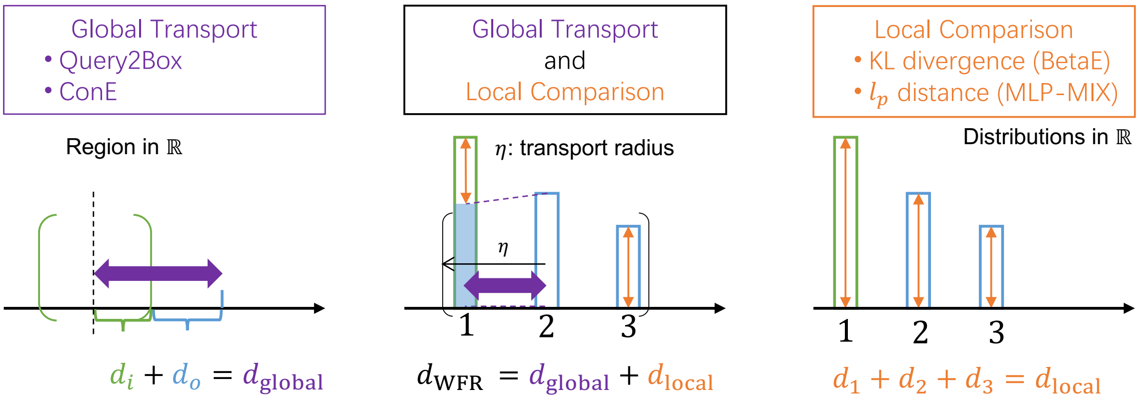

Global Transport
• Query2Box
• ConE
Region in $\mathbb{R}$
$d_i + d_o = d_{\text{global}}$
Global Transport
and
Local Comparison
η: transport radius
η
1
2
3
$d_{\text{WFR}} = d_{\text{global}} + d_{\text{local}}$
Local Comparison
• KL divergence (BetaE)
• l_p distance (MLP-MIX)
Distributions in $\mathbb{R}$
1
2
3
$d_1 + d_2 + d_3 = d_{\text{local}}$

Figure 2: Illustration of different scoring functions. **Left**: global transport, where the difference is measured by how to move mass from one place to another (purple arrows); **Right**: local comparison, where the difference is measured by in-place comparison (yellow arrows); **Mid**: local and global trade-off, where we first move mass in the transport radius η, then compare the unfilled mass.

2.1 Query Embeddings

As a predominant way to answer logical queries, *query embeddings* (Hamilton et al., 2018) embed answer sets into continuous spaces and models set operations with neural networks. The scope of logical queries that query embedding methods can solve is expanded from conjunctive queries (Hamilton et al., 2018), to Existential Positive First-Order (EPFO) queries (Ren et al., 2020; Choudhary et al., 2021a,b), and First-Order (FO) queries (Ren and Leskovec, 2020; Zhang et al., 2021; Amayuelas et al., 2022; Bai et al., 2022; Yang et al., 2022; Wang et al., 2023; Yin et al., 2023; Bai et al., 2023).

Set embeddings of various forms have been heavily investigated, such as vectors (Amayuelas et al., 2022; Chen et al., 2022; Huang et al., 2022), geometric regions (Ren et al., 2020; Zhang et al., 2021; Choudhary et al., 2021b), and probabilistic distributions (Ren and Leskovec, 2020; Choudhary et al., 2021a; Yang et al., 2022). Despite of the various forms of the embeddings, their scoring function captures either *local* comparison, such as Euclidean distance (Amayuelas et al., 2022), inner product (Chen et al., 2022), and KL-divergence (Ren and Leskovec, 2020; Yang et al., 2022), or *global* transport, such as heuristic distance between geometric regions (Ren et al., 2020; Choudhary et al., 2021b; Zhang et al., 2021), Mahalanobis distance (Choudhary et al., 2021a), or the similarity between target particle and the closest particle of a point cloud (Bai et al., 2022).

In this work, we establish a novel scoring function motivated by unbalanced optimal transport theory (Chizat et al., 2018a). As a variant of the optimal transport, it inherits the advantages and balances the *local* comparison and *global* transport.

2.2 Optimal Transport for Embeddings

Optimal transport (OT) (Peyré et al., 2019) introduces the power metric between probabilistic distributions and facilitates many applications in language and graph data (Alvarez-Melis and Jaakkola, 2018; Zhao et al., 2020a; Xu et al., 2021; Li et al., 2021; Tang et al., 2022; Wang et al., 2022a; Li et al., 2022; Tan et al., 2023). It is particularly efficient when embedding graph vertices and words as probabilistic distributions in the Wasserstein space (Muzellec and Cuturi, 2018; Frogner et al., 2018).

Wasserstein-Fisher-Rao (WFR) metric (Chizat et al., 2018a) generalizes the OT between distributions to the measures by balancing local comparison and global transport with a transport radius η. Existing investigations (Zhao et al., 2020b) demonstrated that the WFR metric is a robust and effective measurement for embedding alignment. Previous work measures pretrained embeddings in the WFR space (Wang et al., 2020), while this work is the first to learn embeddings in the WFR space. Moreover, we validate the advantage of WFR space in the context of query embedding.

3 Preliminaries

3.1 Knowledge Graph and Complex Queries

A knowledge grpah $\mathcal{KG} = \{(h, r, t) \in \mathcal{V} \times \mathcal{R} \times \mathcal{V}\}$ is a collections of triples where $h, t \in \mathcal{V}$ are entity nodes and $r \in \mathcal{R}$ is the relation.

Complex queries over knowledge graphs can be defined by first-order formulas. Following previous works (Ren and Leskovec, 2020), we consider a query Q with one free variable node $V_?$ and quantified nodes $V_i, 1 \leq i \leq n$, an arbitrary logical formula can be converted to prenex and DNF forms as follows (Marker, 2002).

$$Q[V_?] = \Box V_1 \cdots \Box V_n . c_1 \vee \cdots \vee c_l, \tag{5}$$

where each quantifier $\Box$ is either $\exists$ or $\forall$, each $c_i, 1 \leq i \leq l$ is a conjunctive clause such that $c_i = y_{i1} \wedge \cdots \wedge y_{im_i}$, and each $y_{ij}, 1 \leq j \leq m_i$ represents an atomic formula or its negation. That is, $y_{ij} = r(a, b)$ or $\neg r(a, b)$, where $r \in \mathcal{R}$, a and b can be either a variable $V_\cdot$ or an entity in $\mathcal{V}$.

3.2 Answer Queries with Set Operator Trees

Queries that can be answered by set operators are of particular interest (Ren and Leskovec, 2020). The answers can be derived by executing set operators in bottom-up order. The leaves of each operator tree are known entities, which are regarded as sets with a single element. The input and the output of each set operator are all sets. We note that queries solvable by set operators are only a fragment of the first-order queries due to their additional assumptions that guarantee their conversion to operation trees (Wang et al., 2021). Moreover, the choice of set operators is not unique to representing the entire class. In this work, we focus on the following operators:

Set Projections Derived from the relations.

Set Operations :

- **Set Intersection** Derived from conjunction.
- **Set Union** Derived from disjunction.
- **Set Complement** Derived from negation.

3.3 Wasserstein-Fisher-Rao (WFR) Metric

Wasserstein-Fisher-Rao metric defines the distances between two measures (Chizat et al., 2018a). Consider two discrete measures in $\mathbb{R}^d$, i.e., $\mu = \sum_{i=1}^M u_i \delta_{x_i}$ and $\nu = \sum_{j=1}^N v_i \delta_{y_j}$, where δ is the Dirac function, $u_i, v_j \geq 0$, and $x_i, y_j \in \mathbb{R}^d$ are the corresponding coordinates for $1 \leq i \leq M$ and $1 \leq j \leq N$. For short hand, $\mathbf{u} = [u_1, ..., u_M]^\top$ and $\mathbf{v} = [v_1, ..., v_N]^\top$ denote column mass vectors. Then the WFR metric is defined by solving the following minimization problem.

$$\mathcal{WFR}(\mu, \nu; \eta) = \min_{P \in \mathbb{R}_+^{M \times N}} J(P; \mu, \nu, \eta), \tag{6}$$

where $P \in \mathbb{R}^{M \times N}$ is the transport plan and P_{ij} indicates the mass transported from x_i to y_j. We denote the global minima P^* of the Problem (6) as the WFR optimal transport plan. The objective function reads,

$$\begin{aligned} J(P; \mu, \nu, \eta) = &\sum_{i=1}^M \sum_{j=1}^N C_{ij} P_{ij} \\ &+ \mathcal{D}(P \mathbb{1}_N \| \mathbf{u}) + \mathcal{D}(P^\top \mathbb{1}_M \| \mathbf{v}), \end{aligned} \tag{7}$$

where $\mathbb{1}_N$ is the column vector in $\mathbb{R}^N$ of all one elements, and $\mathcal{D}(\cdot \| \cdot)$ is the KL divergence. $C \in \mathbb{R}_+^{M \times N}$ is the cost matrix and C_{ij} indicates the cost from x_i to y_j,

$$C_{ij} = -2 \log \left(\cos_+ \left(\frac{\pi}{2} \frac{\|x_i - y_j\|}{\eta} \right) \right). \tag{8}$$

where $\cos_+(x) = \cos(x)$ if $|x| < \pi/2$, otherwise $\cos_+(x) = 0$. η is the hyperparameter for the transport radius.

One of the key properties of the WFR metric could be understood by the geodesics in WFR space, as stated in Theorem 4.1 by Chizat et al. (2018a). Specifically, for two mass points at positions x and y, the transport only applies when $\|x - y\| < \eta$, such as place 1 and 2 in Figure 2, otherwise, only local comparison is counted. We see that the η controls the scope of the transport process.

3.4 Entropic Regularized WFR Solution

The WFR metric in Equation (6) can be computed by the Sinkhorn algorithm with an additional entropic regularization term (Chizat et al., 2018b). Specifically, one could estimate WFR with the following entropic regularized optimization problem,

$$\min_{P \in \mathbb{R}_+^{M \times N}} J(P; \mu, \nu, \eta) + \overbrace{\epsilon \sum_{ij} P_{ij} \log P_{ij}}^{\text{Entropic Regularization}} . \tag{9}$$

The generalized Sinkhorn algorithm (Chizat et al., 2018b) solves the unconstraint dual problem of Problem (9), which maximizes the objective

$$\begin{aligned} D_\epsilon(\phi, \psi; \mathbf{u}, \mathbf{v}, K_\epsilon) = &\langle 1 - \phi, \mathbf{u} \rangle + \langle 1 - \psi, \mathbf{v} \rangle \\ &+ \epsilon \langle 1 - (\phi \otimes \psi)^{\frac{1}{\epsilon}}, K_\epsilon \rangle, \end{aligned} \tag{10}$$

where $K_\epsilon = e^{-\frac{C}{\epsilon}}$ is the kernal matrix, $\phi \in \mathbb{R}^M$ and $\psi \in \mathbb{R}^N$ are dual variables. The update procedure of the $(l+1)$-th step of the j-th Sinkhorn iteration is

$$\phi^{(l+1)} \leftarrow \left[\mathbf{u} \oslash \left(K_\epsilon \psi^{(l)} \right) \right]^{\frac{1}{1+\epsilon}}, \tag{11}$$

$$\psi^{(l+1)} \leftarrow \left[\mathbf{v} \oslash \left(K_\epsilon^\top \phi^{(l+1)} \right) \right]^{\frac{1}{1+\epsilon}}. \tag{12}$$

Let ϕ^* and ψ^* be the optimal dual variables obtained from a converged Sinkhorn algorithm. The optimal transport plan is recovered by

$$P^* = \text{diag}(\phi^*) K_\epsilon \text{diag}(\psi^*). \tag{13}$$

We could see that the Sinkhorn algorithm employs the matrix-vector multiplication that costs $O(MN)$ time. In contrast to the Wasserstein metric that can be approximated by 1D sliced-Wasserstein (Carriere et al., 2017; Kolouri et al., 2019) under $O((M+N)\log(M+N))$ time, there is no known sub-quadratic time algorithm for even approximated WFR metric, which hinders its large-scale application. In the next section, we restrict set embeddings to bounded measures in $\mathbb{R}$. We further develop an $O(d)$ algorithm by leveraging the sparse structure of kernel matrix K_ϵ.

4 Wasserstein-Fisher-Rao Embedding

The goal of this section is to present how to solve complex queries with set embeddings as the Bounded Measure in $\mathbb{R}$. Let the S be an arbitrary set, including the singleton set $\{e\}$ with a single entity e, its embedding is $m[S]$. We denote the collection for all bounded measures as $\mathcal{BM}(\mathbb{R})$. Our discussion begins with the discretization of measure $m[S] \in \mathcal{BM}(\mathbb{R})$ to histogram $m^S \in \mathcal{BM}_d$, where $\mathcal{BM}_d$ is the collection of bounded histograms with d bars. Then we discuss how to parameterize set operators with embeddings in the $\mathcal{BM}_d$ and efficiently compute the scoring function in $\mathcal{BM}_d$. Finally, we introduce how to learn set embeddings and operators.

4.1 Discretize BM1Ds into Histograms

We discretize each $m[S] \in \mathcal{BM}(\mathbb{R})$ as a histogram on a uniform mesh on $\mathbb{R}$. Without loss of generality, the maximum length of bars in the histogram is one, and the mesh spacing is Δ. In this way, each $m[S] = \sum_{i=1}^d m_i^S \delta_{i\Delta}$, where $m_i^S \in [0,1]$ for $1 \le i \le d$. Therefore, it is sufficient to store the discretized mass vector $\mathbf{m}^S = [m_1^S, \dots, m_d^S] \in \mathcal{BM}_d$ because the support set $\{i\Delta\}_{i=1}^d$ is fixed for all $m[S] \in \mathcal{BM}(\mathbb{R})$. Then we discuss set operators on $\mathcal{BM}_d$

4.2 Set Operators on $\mathcal{BM}_d$

Non-parametric Set Operations It should be stressed that the mass vector $\mathbf{m}^S \in \mathcal{BM}_d$ can be interpreted *locally*, where each element of $\mathbf{m}^S$ is regarded the continuous truth value in fuzzy logic. Therefore, set operations **intersection** $\cap$, **union** $\cup$, and **complement** on the $\mathcal{BM}_d$ are modeled by the element-wise t-norm on the mass vector $\mathbf{m}^S$. For the i-th element of the mass vector, $1 \le i \le d$,

$$\text{Intersection} \quad m_i^{S_1 \cap S_2} = m_i^{S_1} \top m_i^{S_2}, \tag{14}$$

$$\text{Union} \quad m_i^{S_1 \cup S_2} = m_i^{S_1} \bot m_i^{S_2}, \tag{15}$$

$$\text{Complement} \quad m_i^{S^C} = 1 - m_i^S, \tag{16}$$

where $\top$ is a t-norm and $\bot$ is the corresponding t-conorm.

Neural Set Projections Each set **projection** is modeled as functions from one mass vector to another given a relation r. We adopt base decomposition (Schlichtkrull et al., 2018) to define a Multi-Layer Perceptron (MLP) from $[0,1]^d$ to $[0,1]^d$. For each fully-connected layer with input $\mathbf{m}^{S,(l)} \in [0,1]^{d_l}$, the output $\mathbf{m}^{S,(l+1)} \in [0,1]^{d_{l+1}}$ through relation r is computed by

$$\mathbf{m}^{S,(l+1)} = \sigma(W_r^{(l)} \mathbf{m}^{S,(l)} + b_r^{(l)}), \tag{17}$$

where σ is an activation function, and $W_r^{(l)}$ and $b_r^{(l)}$ are the weight matrix and bias vector for relation r at the l-th layer. Specifically,

$$W_r^{(l)} = \sum_{j=1}^K V_j^{(l)} r_j, \quad b_r^{(l)} = \sum_{j=1}^K a_j^{(l)} r_j. \tag{18}$$

K is the number of bases, $\mathbf{r} \in \mathbb{R}^K$ is the relation embedding. $V_j^{(l)} \in \mathbb{R}^{d_{l+1} \times d_l}$ and $a_j^{(l)} \in \mathbb{R}^{d_{l+1}}$ the are the base weight matrices and base bias vectors at the l-th layer, respectively.

Dropout on Set Complement Inspired by the dropout for neural networks that improves the generalizability, we propose to apply dropout to the set complement operation. The idea is to randomly alter the elements in mass vectors before the complement operation by randomly setting their values to $\frac{1}{2}$. In this way, the complemented elements are also $\frac{1}{2}$. This technique improves the generalizability of the set complement operator.

4.3 Scoring function for $\mathcal{BM}_d$

Consider $\mathbf{m}^{S_1}, \mathbf{m}^{S_2} \in \mathcal{BM}_d$. It is straight forward to score this pair by $\mathcal{WFR}(\mathbf{m}^{S_1}, \mathbf{m}^{S_2}; \eta)$. However, direct applying the Sinkhorn algorithm requires a $O(d^2)$ time, which hinders the large-scale computation of the WFR metric. In this part, we introduce (1) convolution-based Sinkhorn to reduce the complexity within $O(d)$ time and (2) block diagonal transport as an additional mechanism for the *local* and *global* tradeoff besides the transport radius η. We note that our contribution does not coincide with the recent linear-time "fast" Sinkhorn algorithms (Liao et al., 2022a,b), which do not apply to unbalanced optimal transport in $\mathcal{BM}_d$.

Convolution-based Sinkhorn The computational bottleneck for the Sinkhorn update shown in Equation (11) and (12) is the matrix-vector multiplication. When comparing the discretized measures in $\mathcal{BM}_d$, K_ϵ exhibits a symmetric and diagonal structure.

$$K_{\epsilon,ij} = \begin{cases} \cos\left(\frac{\pi}{2}\frac{|i-j|}{\eta/\Delta}\right)^{\frac{2}{\epsilon}} & |i-j| < \frac{\eta}{\Delta} \\ 0 & \text{o.w.} \end{cases} \quad (19)$$

Let $\omega = \lfloor \frac{\eta}{\Delta} \rfloor$ be the window size, the matrix-vector multiplication $K_\epsilon \mathbf{v} = K_\epsilon^\top \mathbf{v}$ could be simplified as a discrete convolution $H(\beta, \omega) * \mathbf{v}$, where the kernel $[H(\beta,\omega)]_k = \cos\left(\frac{\pi\beta}{2\omega}k\right), -\omega \le k \le \omega$ and $\beta := \lfloor \frac{\eta}{\Delta} \rfloor / \frac{\eta}{\Delta} \in (1 - \frac{1}{\omega+1}, 1]$ is another hyperparameter. Specifically, the i-th element of $H * \mathbf{v}$ is

$$[H(\beta,\omega) * \mathbf{v}]_i = \sum_{k=-\omega}^{+\omega} H_k v_{i+k} \mathbf{1}_{1 \le i+k \le d}, \quad (20)$$

where $\mathbf{1}_{1\le i+k \le d} = 1$ if and only if $1 \le i+k \le d$. Then the Sinkhorn algorithm could be simplified as

$$\phi^{(l+1)} \leftarrow \left[\mathbf{u} \oslash \left(H(\beta,\omega) * \psi^{(l)}\right)\right]^{\frac{1}{1+\epsilon}}, \quad (21)$$

$$\psi^{(l+1)} \leftarrow \left[\mathbf{v} \oslash \left(H(\beta,\omega) * \phi^{(l+1)}\right)\right]^{\frac{1}{1+\epsilon}}. \quad (22)$$

Hence, the time complexity of the Sinkhorn algorithm could be reduced to $O(\omega d)$. In our setting, ω is the window size that interpolates the global transport and local comparison, and β is chosen to be 1 in every setting.

Once the convolution-based Sinkhorn algorithm converged, we could approximate the WFR metric via the D_ϵ with optimal ϕ^* and ψ^*. For complex query-answering, the final answers are ranked by their distances (the smaller, the better). This process could be accelerated by the primal-dual pruning for WFR-based k-nearest neighbors (Wang et al., 2020) or the Wasserstein Dictionary Learning (Schmitz et al., 2018).

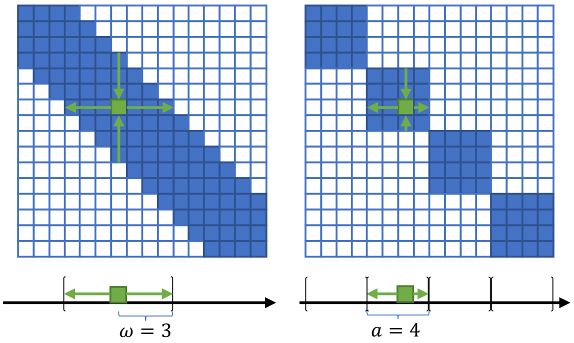


Figure 3: Example of 16 × 16 transport plan matrices by two mechanisms. The zero elements are indicated by white blocks while the (possible) non-zero elements are colored. The transport scope of a sample mass point (green block) is illustrated by the arrows. Left: *Relative* scope by the WFR transport of window size $\omega = 3$; Right: *Absolute* scope by the block diagonal kernel of block size $a = 4$.

Block Diagonal Transport Besides the window size ω that controls the scope of transport *relative* to each mass point, we provide another mechanism to restrict the scope of the transport by the *absolute* position of each mass point. Specifically, we consider the block diagonal kernel matrix K_ϵ^b of b blocks, and $a = d/b$ is the size of each diagonal block. We could see from Equation (13) that the block diagonal kernel leads to the block diagonal transport plan. Figure 3 illustrates the differences between the two mechanisms for restricting global transport in terms of possible transport plans.

Computing the Scoring Function We propose to define the scoring function Dist computed by a convolution-based Sinkhorn with a block diagonal kernel. It should be stressed that a Problem (9) of size $d \times d$ could be regarded as solving b independent problems of size $a \times a$ under the block diagonal problem. This behavior encourages a greater parallelization of the Sinkhorn iterations (21) and (22). We assume $a > \omega$ to ensure each block contains at least a window size of WFR transport so that those two mechanisms could work together. Given the parallel nature of 1D convolution, the entire distance can be highly parallelized with GPU. Specifically, the scoring function Dist is given in Algorithm 1.

4.4 Learning Embeddings in $\mathcal{BM}_d$

Let $\mathbf{m}^Q \in \mathcal{BM}_d$ be a query embedding of query $Q[V_?]$ and $\mathbf{m}^e \in \mathcal{BM}_d$ be the set embedding for unitary set $\{e\}$ with element e. We follow the prac-

Algorithm 1 Scoring function on $\mathcal{BM}_d$ (PyTorch-like style)

```
Require: two bounded measures $\mathbf{m}^{S_1}, \mathbf{m}^{S_2} \in \mathcal{BM}_d$, entropic regularization $\epsilon$, window size $\omega$, number of blocks $b$ such that the block size $a = d/b \geq \omega$, number of iteration $L$.
procedure Dist($\mathbf{m}^{S_1}, \mathbf{m}^{S_2}, \epsilon, \omega, a, b, L$)
    $M_1 \leftarrow \mathbf{m}^{S_1}$.reshape$(1, b, a)$.
    $M_2 \leftarrow \mathbf{m}^{S_2}$.reshape$(1, b, a)$.
    Initialize $H \leftarrow H(1, \omega)$.
    Initialize $\psi \leftarrow$ ones$(1, b, a)$.
    for $l = 1, ..., L$ do
        $\phi \leftarrow M_1/$conv1d$(\psi, H)$.
        $\psi \leftarrow M_2/$conv1d$(\phi, H)$.
    end for
    $\phi^* \leftarrow \phi$.reshape$(d))$
    $\psi^* \leftarrow \psi$.reshape$(d))$
    return $D_\epsilon(\phi^*, \psi^*; \mathbf{m}^{S_1}, \mathbf{m}^{S_2}, K_\epsilon^b)$.
end procedure
```

tice in Ren and Leskovec (2020) to train the parameterized projections and embeddings with negative sampling. For a query Q, we sample one answer a and K_{neg} negative samples $\{v_k\}_{k=1}^{K_{\text{neg}}}$. The objective function is

$$L = -\log \sigma \left(\gamma - \rho \text{Dist} \left(\mathbf{m}^a, \mathbf{m}^Q\right)\right) - \sum_{k=1}^{K_{\text{neg}}} \frac{1}{K_{\text{neg}}} \log \sigma \left(\rho \text{Dist} \left(\mathbf{m}^{v_k}, \mathbf{m}^Q\right) - \gamma\right), \tag{23}$$

where γ is the margin, and ρ is the scale, and σ is the sigmoid function.

5 Experiments

In this section, we evaluate the performance of WFRE on complex query answering in three aspects: (1) we compare WFRE with other SOTA query embedding methods over commonly used datasets on three knowledge graphs (Ren and Leskovec, 2020); (2) we evaluate WFRE on 301 query types to justify its combinatorial generalizability (Wang et al., 2021); (3) we train and evaluate WFRE on a complex query answering datasets on WordNet (Miller, 1995), a lexical KG whose relations are typically hierarchical (Huang et al., 2022). Aspects (2) and (3) emphasize on different query types and the underlying KG, respectively. These results provide empirical evidence for WFRE's strong capability for applying to various query types and KGs. Moreover, we also investigate the *local* and *global* tradeoff of WFRE on ω and a in the ablation study. Other results are listed in the Appendix.

5.1 Experimental Settings

For all experiments, we follow the practice of training and evaluation in Ren and Leskovec (2020). We train query embeddings on train data, select hyperparameters on valid data, and report the scores on test data. Details about the training and evaluation protocol are described in Appendix B. For WFRE, the hyperparameters are listed and discussed in Appendix C. All experiments are conducted on one V100 GPU of 32G memory with PyTorch (Paszke et al., 2019).

5.2 Benchmark Datasets

Datasets on FB15k-237 (Bordes et al., 2013a), FB15k (Toutanova and Chen, 2015), and NELL (Xiong et al., 2017b) proposed by (Ren and Leskovec, 2020) are commonly used to evaluate the effectiveness of query embedding methods. WFRE is compared with baselines with local comparison and global transport, including BetaE (Ren and Leskovec, 2020), ConE (Zhang et al., 2021) MLP-MIX (Alivanistos et al., 2022), Q2P (Bai et al., 2022), and GammaE (Yang et al., 2022). For fairness, we compare the union operators with the DNF treatment introduced by Ren and Leskovec (2020) where scores of answers are merged from those scores of the containing conjunctive queries. Other treatments about union operators are discussed in Appendix E Detailed discussions about the datasets and baselines are listed in Appendix D.1. Table 1 shows how WFRE outperforms existing methods by a large margin in terms of the scores averaged from queries with and without logic negation.

5.3 Combinatorial Generalization on Queries

We also explore how WFRE generalizes on the combinatorial space of complex queries on a benchmark targeting the combinatorial generalizability of query embedding methods (Wang et al., 2021). Details of datasets are presented in Appendix D.2

Results of 301 different query types are averaged by the number of anchor nodes and the maximum depth of the operator tree and are visualized in Figure 4. To illustrate the combinatorial generalizability of complex queries, we normalize scores on query types with the scores on BetaE, as indicated in the axis labels in Figure 4. Then we plot the results into lines by the number of anchor nodes

Table 1: MRR scores for answering all tasks on FB15k, FB15k-237, and NELL. Scores of baselines are taken from their original paper. The boldface indicates the best scores. A_{P} is the average score for queries without negation (EPFO queries). A_{N} is the average score for queries with negation.

Dataset	QE	1P	2P	3P	2I	3I	PI	IP	2U	UP	2IN	3IN	INP	PIN	PNI	A_{P}	A_{N}
FB15k	BetaE	65.1	25.7	24.7	55.8	66.5	43.9	28.1	40.1	25.2	14.3	14.7	11.5	6.5	12.4	41.6	11.8
	ConE	75.3	33.8	29.2	64.4	73.7	50.9	35.7	55.7	31.4	17.9	18.7	12.5	9.8	15.1	49.8	14.8
	MLPMIX	69.7	27.7	23.9	58.7	69.9	46.7	30.8	38.2	24.8	17.2	17.8	13.5	9.1	15.2	43.4	14.8
	Q2P	82.6	30.8	25.5	65.1	74.7	49.5	34.9	32.1	26.2	21.9	20.8	12.5	8.9	17.1	46.8	16.4
	GammaE	76.5	36.9	**31.4**	65.4	75.1	53.9	39.7	**53.5**	30.9	20.1	20.5	13.5	11.8	17.1	51.3	16.6
	WFRE	**81.1**	**37.7**	30.5	**68.5**	**78.0**	**56.3**	**41.8**	48.0	**33.1**	**26.1**	**26.5**	**15.6**	**13.7**	**19.4**	**52.8**	**20.2**
FB15k-237	BetaE	39.0	10.9	10.0	28.8	42.5	22.4	12.6	12.4	9.7	5.1	7.9	7.4	3.6	3.4	20.9	5.4
	ConE	41.8	12.8	11.0	32.6	47.3	25.5	14.0	14.5	10.8	5.4	8.6	7.8	4.0	3.6	23.4	5.9
	MLPMIX	42.4	11.5	9.9	33.5	46.8	25.4	14.0	14.0	9.2	6.6	10.7	8.1	4.7	4.4	22.9	6.9
	Q2P	39.1	11.4	10.1	32.3	47.7	24.0	14.3	8.7	9.1	4.4	9.7	7.5	4.6	3.8	21.9	6.0
	GammaE	43.2	13.2	11.0	33.5	47.9	27.2	15.9	13.9	10.3	6.7	9.4	**8.6**	4.8	**4.4**	24.0	6.8
	WFRE	**44.1**	**13.4**	**11.1**	**35.1**	**50.1**	**27.4**	**17.2**	**13.9**	**10.9**	**6.9**	**11.2**	**8.5**	**5.0**	4.3	**24.8**	**7.2**
NELL	BetaE	53.0	13.0	11.5	37.6	47.5	24.1	14.3	12.2	8.5	5.1	7.8	10.0	3.1	3.5	24.6	5.9
	ConE	53.1	16.1	13.9	40.0	50.8	26.3	17.5	15.3	11.3	5.7	8.1	10.8	3.5	3.9	27.2	6.4
	MLPMIX	55.4	16.5	13.9	39.5	51.0	25.7	18.3	14.7	11.2	5.1	8.0	10.0	3.1	3.5	27.4	5.9
	Q2P	56.5	15.2	12.5	35.8	48.7	22.6	16.1	11.1	10.4	5.1	7.4	10.2	3.3	3.4	25.5	6.0
	GammaE	55.1	17.3	14.2	**41.9**	51.1	26.9	18.3	15.1	11.2	6.3	8.7	11.4	4.0	**4.5**	27.9	7.0
	WFRE	**58.6**	**18.6**	**16.0**	41.2	**52.7**	**28.4**	**20.7**	**16.1**	**13.2**	**6.9**	**8.8**	**12.5**	**4.1**	4.4	**29.5**	**7.3**

Table 2: MRR scores of different query embedding methods on WN18RR. A_{p} is the average of scores from 1P, 2P, and 3P queries; A_{ℓ} is the average of scores from other queries without negation; A_{N} is the average of scores from queries with negation. Scores are taken from Huang et al. (2022).

QE	1P	2P	3P	2I	3I	IP	PI	2IN	3IN	INP	PIN	PNI	2U	UP	A_{p}	A_{ℓ}	A_{N}
BetaE	44.13	9.85	3.86	57.19	76.26	17.97	32.59	12.77	59.98	5.07	4.04	7.48	7.57	5.39	19.28	32.83	17.87
LinE	45.12	12.35	6.70	47.11	67.13	14.73	24.87	12.50	60.81	7.34	5.20	7.74	8.49	6.93	21.39	28.21	18.72
WFRE	52.78	21.00	15.18	68.23	88.15	26.50	40.97	18.99	69.07	14.29	11.06	11.01	15.14	15.81	29.65	42.46	24.88

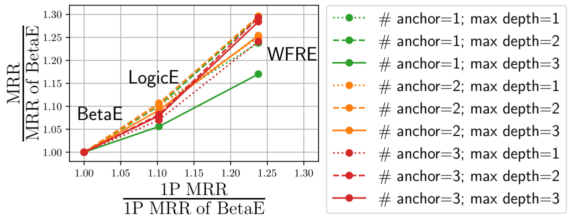


Figure 4: Visualization of different query embedding methods on combinatorial generalizability benchmark (Wang et al., 2021). Results of BetaE and LogicE are taken from Wang et al. (2021). The slopes of lines indicate how the performance of a complex query grows as the performance of the one-hop query grows.

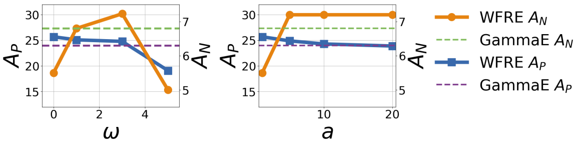
and the max depths. Scores from the same model are located at the same vertical line. We find that WFRE not only improves the performance significantly but also generalizes better in combinatorial complex queries with a larger slope compared to LogicE (Luus et al., 2021).

5.4 Complex Queries on Hierarchical KG

Evaluations above are restricted to three commonly used knowledge graphs. Then, we turn to another type of the underlying knowledge graph, which is characterized by the hierarchy of its relation. We train and evaluate WFRE on a complex query dataset proposed by Huang et al. (2022) on WordNet (Miller, 1995). Details of this dataset are shown in Appendix D.3. We compare WFRE to LinE (Huang et al., 2022), another histogram-based query embedding proposed to solve queries on hierarchical KG without global transport. Table 2 shows the results on WR18RR. We could see that WFRE significantly outperforms LinE and BetaE. In particular, WFRE significantly improved the performance of BetaE and LinE on longer multi-hop queries, i.e., 1P, 2P, and 3P queries. It should be stressed that LinE also used histograms as WFRE but trained with the scoring function motivated only by local comparison. This result shows that WFRE is suitable for modeling hierarchical relations because the *local* and *global* tradeoff on the scoring function learns better embeddings WFRE. It also confirms that Wasserstein spaces make the embeddings more efficient (Frogner et al., 2018).

5.5 Local and Global Trade-off

We further investigate how two mechanisms to restrict the transport, i.e., transport window size ω and block size a affect the performance. Experiments are conducted on queries on FB15k-237 sampled by Ren and Leskovec (2020). We alter one value and fix another one. The default choice is $(\omega, a) = (3, 5)$. Figure 5 demonstrates the effect of these two hyperparameters.



(i) (ii)

Figure 5: The effect of hyperparameter ω and a. The default choice is $(\omega, a) = (3, 5)$.

Compared to the most recent SOTA query embedding GammaE (Yang et al., 2022), the result confirms the importance of the trade-off between local comparison and global transport. When the block size $a = 5$, we find that larger window size ω hurts the performance of negation. Meanwhile, the performance of queries without negation (EPFO queries) reaches their maximum when properly choosing $\omega = 3$. When the window size is fixed $\omega = 3$ and a is small, we see that the performance of EPFO and negation queries follows our observation for window size. Further increasing the block size a only has little impact on the EPFO queries but also hurts the performance of negation queries. It indicates that a proper a is necessary for performance when ω is fixed. This observation could help to improve the degree of parallelization of the convolution-based Sinkhorn algorithm.

6 Conclusion

In this paper, we propose WFRE, a new query embedding method for complex queries on knowledge graphs. The key feature of WFRE against to previous methods is its scoring function that balances local comparison and global transport. Empirical results show that WFRE is the state-of-the-art query embedding method for complex query answering, and has good generalizability to combinatorially complex queries and hierarchical knowledge graphs. The ablation study justifies the importance of the local and global trade-off.

7 Limitation

WFRE suffers common drawbacks from the existing query embedding methods. The queries that can be solved by such methods are a limited subclass of first-order queries. It is also not clear how to apply WFRE to unseen entities and relations in an inductive setting.

8 Ethics Statement

As a query embedding method, WFRE has stronger generalizability to different query types and knowledge graphs. Experiments and evaluations in this paper involve no ethical issues and are not even related to any human entities. WFRE could be potentially used to efficiently infer private information from an industrial-level knowledge graph. This is a common potential risk for approaches targeting data incompleteness and link prediction.

Acknowledgement

The authors of this paper were supported by the NSFC Fund (U20B2053) from the NSFC of China, the RIF (R6020-19 and R6021-20) and the GRF (16211520 and 16205322) from RGC of Hong Kong, the MHKJFS (MHP/001/19) from ITC of Hong Kong and the National Key R&D Program of China (2019YFE0198200) with special thanks to HKMAAC and CUSBLT. We also thank the support from NVIDIA AI Technology Center (NVAITC) and the UGC Research Matching Grants (RMGS20EG01-D, RMGS20CR11, RMGS20CR12, RMGS20EG19, RMGS20EG21, RMGS23CR05, RMGS23EG08).

References

Dimitrios Alivanistos, Max Berrendorf, Michael Cochez, and Mikhail Galkin. 2022. Query Embedding on Hyper-Relational Knowledge Graphs. In *International Conference on Learning Representations*.

David Alvarez-Melis and Tommi Jaakkola. 2018. Gromov-Wasserstein alignment of word embedding spaces. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1881–1890, Brussels, Belgium. Association for Computational Linguistics.

Alfonso Amayuelas, Shuai Zhang, Susie Xi Rao, and Ce Zhang. 2022. Neural Methods for Logical Reasoning over Knowledge Graphs. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Erik Arakelyan, Daniel Daza, Pasquale Minervini, and Michael Cochez. 2021. Complex Query Answering with Neural Link Predictors. In *International Conference on Learning Representations*.

Jiaxin Bai, Zihao Wang, Hongming Zhang, and Yangqiu Song. 2022. Query2Particles: Knowledge Graph Reasoning with Particle Embeddings. In *Findings of the Association for Computational Linguistics:*

NAACL 2022, pages 2703–2714, Seattle, United States. Association for Computational Linguistics.

Jiaxin Bai, Tianshi Zheng, and Yangqiu Song. 2023. Sequential query encoding for complex query answering on knowledge graphs. *arXiv preprint arXiv:2302.13114*.

Kurt D. Bollacker, Colin Evans, Praveen K. Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: A collaboratively created graph database for structuring human knowledge. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2008, Vancouver, BC, Canada, June 10-12, 2008*, pages 1247–1250. ACM.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013a. Translating Embeddings for Modeling Multi-relational Data. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. 2013b. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 2787–2795.

Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka Jr, and Tom M. Mitchell. 2010. Toward an Architecture for Never-Ending Language Learning. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010*. AAAI Press.

Mathieu Carriere, Marco Cuturi, and Steve Oudot. 2017. Sliced wasserstein kernel for persistence diagrams. In *International conference on machine learning*, pages 664–673. PMLR.

Xuelu Chen, Ziniu Hu, and Yizhou Sun. 2022. Fuzzy Logic Based Logical Query Answering on Knowledge Graphs. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, pages 3939–3948. AAAI Press.

Lenaic Chizat, Gabriel Peyré, Bernhard Schmitzer, and François-Xavier Vialard. 2018a. An interpolating distance between optimal transport and fisher–rao metrics. *Foundations of Computational Mathematics*, 18(1):1–44.

Lenaic Chizat, Gabriel Peyré, Bernhard Schmitzer, and François-Xavier Vialard. 2018b. Scaling algorithms for unbalanced optimal transport problems. *Mathematics of Computation*, 87(314):2563–2609.

Nurendra Choudhary, Nikhil Rao, Sumeet Katariya, Karthik Subbian, and Chandan Reddy. 2021a. Probabilistic Entity Representation Model for Reasoning over Knowledge Graphs. In *Advances in Neural Information Processing Systems*, volume 34, pages 23440–23451. Curran Associates, Inc.

Nurendra Choudhary, Nikhil Rao, Sumeet Katariya, Karthik Subbian, and Chandan K. Reddy. 2021b. Self-Supervised Hyperboloid Representations from Logical Queries over Knowledge Graphs. In *Proceedings of the Web Conference 2021*, WWW '21, pages 1373–1384, New York, NY, USA. Association for Computing Machinery.

Daniel Daza and Michael Cochez. 2020. Message Passing Query Embedding.

Charlie Frogner, Farzaneh Mirzazadeh, and Justin Solomon. 2018. Learning embeddings into entropic wasserstein spaces. In *International Conference on Learning Representations*.

Albert Gu, Frederic Sala, Beliz Gunel, and Christopher Ré. 2018. Learning mixed-curvature representations in product spaces. In *International Conference on Learning Representations*.

Petr Hájek. 1998. *Metamathematics of Fuzzy Logic*, volume 4 of *Trends in Logic*. Springer Netherlands, Dordrecht.

William L. Hamilton, Payal Bajaj, Marinka Zitnik, Dan Jurafsky, and Jure Leskovec. 2018. Embedding Logical Queries on Knowledge Graphs. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 2030–2041.

Zijian Huang, Meng-Fen Chiang, and Wang-Chien Lee. 2022. LinE: Logical Query Reasoning over Hierarchical Knowledge Graphs. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '22, pages 615–625, New York, NY, USA. Association for Computing Machinery.

Soheil Kolouri, Kimia Nadjahi, Umut Simsekli, Roland Badeau, and Gustavo Rohde. 2019. Generalized sliced wasserstein distances. *Advances in neural information processing systems*, 32.

Bhushan Kotnis, Carolin Lawrence, and Mathias Niepert. 2021. Answering Complex Queries in Knowledge Graphs with Bidirectional Sequence Encoders. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(6):4968–4977.

Manling Li, Tengfei Ma, Mo Yu, Lingfei Wu, Tian Gao, Heng Ji, and Kathleen McKeown. 2021. Timeline summarization based on event graph compression via time-aware optimal transport. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6443–6456, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xinhang Li, Zhaopeng Qiu, Xiangyu Zhao, Zihao Wang, Yong Zhang, Chunxiao Xing, and Xian Wu. 2022. Gromov-wasserstein guided representation learning for cross-domain recommendation. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 1199–1208.

Qichen Liao, Jing Chen, Zihao Wang, Bo Bai, Shi Jin, and Hao Wu. 2022a. Fast sinkhorn i: An $o(n)$ algorithm for the wasserstein-1 metric. *Communications in Mathematical Sciences*, 20(7):2053–2067.

Qichen Liao, Zihao Wang, Jing Chen, Bo Bai, Shi Jin, and Hao Wu. 2022b. Fast sinkhorn ii: Collinear triangular matrix and linear time accurate computation of optimal transport. *arXiv preprint arXiv:2206.09049*.

Leonid Libkin and Cristina Sirangelo. 2009. Open and Closed World Assumptions in Data Exchange. In *Proceedings of the 22nd International Workshop on Description Logics (DL 2009), Oxford, UK, July 27-30, 2009*, volume 477 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Xuan Lin, Zhe Quan, Zhi-Jie Wang, Tengfei Ma, and Xiangxiang Zeng. 2020. Kgnn: Knowledge graph neural network for drug-drug interaction prediction. In *IJCAI*, volume 380, pages 2739–2745.

Xiao Liu, Shiyu Zhao, Kai Su, Yukuo Cen, Jiezhong Qiu, Mengdi Zhang, Wei Wu, Yuxiao Dong, and Jie Tang. 2022. Mask and Reason: Pre-Training Knowledge Graph Transformers for Complex Logical Queries. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '22, pages 1120–1130, New York, NY, USA. Association for Computing Machinery.

Francois Luus, Prithviraj Sen, Pavan Kapanipathi, Ryan Riegel, Ndivhuwo Makondo, Thabang Lebese, and Alexander Gray. 2021. Logic Embeddings for Complex Query Answering. *arXiv:2103.00418 [cs]*.

D. Marker. 2002. *Model Theory: An Introduction*. Number 217 in Graduate Texts in Mathematics. Springer, New York.

George A. Miller. 1995. WordNet: A lexical database for English. *Communications of the ACM*, 38(11):39–41.

Boris Muzellec and Marco Cuturi. 2018. Generalizing point embeddings using the wasserstein space of elliptical distributions. *Advances in Neural Information Processing Systems*, 31.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Gabriel Peyré, Marco Cuturi, et al. 2019. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607.

Hongyu Ren, Hanjun Dai, Bo Dai, Xinyun Chen, Michihiro Yasunaga, Haitian Sun, Dale Schuurmans, Jure Leskovec, and Denny Zhou. 2021. LEGO: Latent Execution-Guided Reasoning for Multi-Hop Question Answering on Knowledge Graphs. In *Proceedings of the 38th International Conference on Machine Learning*, pages 8959–8970. PMLR.

Hongyu Ren, Hanjun Dai, Bo Dai, Xinyun Chen, Denny Zhou, Jure Leskovec, and Dale Schuurmans. 2022. SMORE: Knowledge Graph Completion and Multi-hop Reasoning in Massive Knowledge Graphs. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '22, pages 1472–1482, New York, NY, USA. Association for Computing Machinery.

Hongyu Ren, Mikhail Galkin, Michael Cochez, Zhaocheng Zhu, and Jure Leskovec. 2023. Neural graph reasoning: Complex logical query answering meets graph databases. *arXiv preprint arXiv:2303.14617*.

Hongyu Ren, Weihua Hu, and Jure Leskovec. 2020. Query2box: Reasoning over Knowledge Graphs in Vector Space Using Box Embeddings. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Hongyu Ren and Jure Leskovec. 2020. Beta Embeddings for Multi-Hop Logical Reasoning in Knowledge Graphs. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, Virtual*.

Michael Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. 2018. Modeling Relational Data with Graph Convolutional Networks. In *The Semantic Web*, Lecture Notes in Computer Science, pages 593–607, Cham. Springer International Publishing.

Morgan A Schmitz, Matthieu Heitz, Nicolas Bonneel, Fred Ngole, David Coeurjolly, Marco Cuturi, Gabriel Peyré, and Jean-Luc Starck. 2018. Wasserstein dictionary learning: Optimal transport-based unsupervised nonlinear dictionary learning. *SIAM Journal on Imaging Sciences*, 11(1):643–678.

Zhiquan Tan, Zihao Wang, and Yifan Zhang. 2023. Seal: Simultaneous label hierarchy exploration and learning. *arXiv preprint arXiv:2304.13374*.

Peggy Tang, Kun Hu, Rui Yan, Lei Zhang, Junbin Gao, and Zhiyong Wang. 2022. OTExtSum: Extractive Text Summarisation with Optimal Transport. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1128–1141, Seattle, United States. Association for Computational Linguistics.

Kristina Toutanova and Danqi Chen. 2015. Observed versus latent features for knowledge base and text inference. In *Proceedings of the 3rd Workshop on Continuous Vector Space Models and Their Compositionality*, pages 57–66, Beijing, China. Association for Computational Linguistics.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: A free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. 2019. Kgat: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 950–958.

Zihao Wang, Jiaheng Dou, and Yong Zhang. 2022a. Unsupervised sentence textual similarity with compositional phrase semantics. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4976–4995.

Zihao Wang, Yangqiu Song, Ginny Y Wong, and Simon See. 2023. Logical message passing networks with one-hop inference on atomic formulas. *arXiv preprint arXiv:2301.08859*.

Zihao Wang, Hang Yin, and Yangqiu Song. 2021. Benchmarking the Combinatorial Generalizability of Complex Query Answering on Knowledge Graphs. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, Virtual*.

Zihao Wang, Hang Yin, and Yangqiu Song. 2022b. Logical queries on knowledge graphs: Emerging interface of incomplete relational data. *Data Engineering*, page 3.

Zihao Wang, Datong Zhou, Ming Yang, Yong Zhang, Chenglong Rao, and Hao Wu. 2020. Robust document distance with wasserstein-fisher-rao metric. In *Asian Conference on Machine Learning*, pages 721–736. PMLR.

Chenyan Xiong, Russell Power, and Jamie Callan. 2017a. Explicit semantic ranking for academic search via knowledge graph embedding. In *Proceedings of the 26th international conference on world wide web*, pages 1271–1279.

Wenhan Xiong, Thien Hoang, and William Yang Wang. 2017b. Deeppath: A reinforcement learning method for knowledge graph reasoning. In *EMNLP*.

Jingjing Xu, Hao Zhou, Chun Gan, Zaixiang Zheng, and Lei Li. 2021. Vocabulary learning via optimal transport for neural machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7361–7373, Online. Association for Computational Linguistics.

Zezhong Xu, Wen Zhang, Peng Ye, Hui Chen, and Huajun Chen. 2022. Neural-Symbolic Entangled Framework for Complex Query Answering. In *Advances in Neural Information Processing Systems*.

Dong Yang, Peijun Qing, Yang Li, Haonan Lu, and Xiaodong Lin. 2022. GammaE: Gamma Embeddings for Logical Queries on Knowledge Graphs.

Hang Yin, Zihao Wang, and Yangqiu Song. 2023. On existential first order queries inference on knowledge graphs. *arXiv preprint arXiv:2304.07063*.

Zhanqiu Zhang, Jie Wang, Jiajun Chen, Shuiwang Ji, and Feng Wu. 2021. ConE: Cone Embeddings for Multi-Hop Reasoning over Knowledge Graphs. In *Advances in Neural Information Processing Systems*, volume 34, pages 19172–19183. Curran Associates, Inc.

Xu Zhao, Zihao Wang, Hao Wu, and Yong Zhang. 2020a. Semi-supervised bilingual lexicon induction with two-way interaction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2973–2984.

Xu Zhao, Zihao Wang, Yong Zhang, and Hao Wu. 2020b. A relaxed matching procedure for unsupervised bli. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3036–3041.

Zhaocheng Zhu, Mikhail Galkin, Zuobai Zhang, and Jian Tang. 2022. Neural-Symbolic Models for Logical Queries on Knowledge Graphs.

A Other Methods for Complex Query Answering

Despite of computing query embedding with neural set operators, other approaches are also proposed to derive answers. Daza and Cochez (2020); Liu et al. (2022) explored the graph representation to answer the logical queries with graph neural networks while Kotnis et al. (2021) discussed the logical queries as sequence representation. Arakelyan et al. (2021) solves the logical queries by solving the continuous optimization problems induced by neural link predictors. However, these discussions are only limited to EPFO queries *without* logical negation. It is not clear how these methods handle first-order queries.

Meanwhile, neural symbolic methods estimate the probability for whether each entity is the answer set (Zhu et al., 2022; Xu et al., 2022) even at each intermediate step. Therefore, it requires $O(|\mathcal{V}| + |\mathcal{T}|)$ space and time to derive answers for a given query, where $\mathcal{V}$ and $\mathcal{T}$ are the entity set and the triple set of a knowledge graph. Compared to the query embedding methods that require only $O(d)$, where d is the fixed dimension of the embedding space, it is challenging to scale neural symbolic methods to logical queries on large-scale knowledge graphs (Ren et al., 2022).

B Training and Evaluation Protocal

We follow the commonly used experiment settings for EFO-1 query answering, which aims to find non-trivial answers in incomplete graphs and generalize to queries of unseen types.

Given an underlying KG $\mathcal{G} = (\mathcal{V}, \mathcal{R})$ and its triple set $\mathcal{T}$, we sample three subgraphs by change the scope of triples $\mathcal{T}_{\text{train}} \subset \mathcal{T}_{\text{valid}} \subset \mathcal{T}_{\text{test}} = \mathcal{T}$. Following the standard evaluation protocol, we aim to find the non-trivial answers which cannot be directly discovered by traversing graphs. We denote $[q]_{\text{train}}$ as the answer set of query q in the train graph, the answer set we focus on is $[q]_{\text{test}} \backslash [q]_{\text{train}}$, and these are easy answers that can only be reasoned or predicted. The hard answers are $[q]_{\text{test}} \setminus [q]_{\text{valid}}$. Then we would rank the easy(hard) answers against all the non-answer sets $\mathcal{V}/[q]_{\text{valid}}(\mathcal{V}/[q]_{\text{test}})$. After getting the rank r, we calculated mean reciprocal rank (MRR): $\frac{1}{r}$ and Hits at K(Hits@K):$1_{r<K}$ as metric to measure the performance of models.

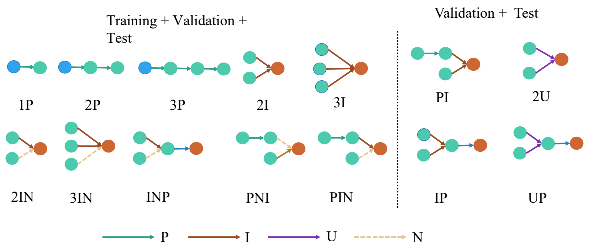


Figure 6: Visualization of logic query structures. The left queries just appear in the training phase, and all the queries are used in the validation and test phases.

C Settings for WFRE

Our framework is implemented with Pytorch. Our code is based on the pipeline for the EFO-1-QA benchmark (Wang et al., 2021) and we use AdamW as the optimizer.

There are also some hyperparameters in code. We apply dropout on projection network and denote the drop probability as Drop$_p$. The Sinkhorn's algorithm's maximum iteration is denoted as K_S. And We just set the layer of Projection MLP as 1 because of the results of the experiment relsults. The hyperparameters and their related information in WFRE are listed in Table 3. We finetune the hyperparameters for four datasets and the results are presented in Table 4. Hope the two tables could help you quickly understand our model's hyperparameters.

D Datasets and Baselines

In this section, we introduce the baselines in three experiments. Table 5 presents the basic statistics of different queries on all the benchmark datasets.

D.1 Benchmark datasets

For commonly used dataset (Ren and Leskovec, 2020), there are ten query types 1P, 2P, 3P, 2I, 3I, 2IN, 3IN, INP, PNI, PIN in the training dataset but also four unseen query structures IP, PI, 2U, and UP in the valid and test datasets. The related query structures are visualized in Figure 6. The purpose of unseen types of the vaild and test queries is to test the combinatorial generalizability of the neural set operator.

In this part, we choose the following complex query embedding methods which support arbitrary EFO1 queries:

BetaE (Ren and Leskovec, 2020) Beta distribution embedding whose scoring function is

Table 3: Hyperparameters used for WFRE

Hyperparameter	Comments	Choices
Learning rate	Model's convergence	$\{0.0001, 0.0005, 0.001\}$
Training steps	Model's convergence	$\{240000, 300000, 360000\}$
Negative sample size K_{neg}	Model's convergence	$\{32\}$
Weight decay	Regulararization for model	$\{0.001, 0.005, 0.01\}$
Drop_p	Regulararization for projection operation	$\{0.05\}$
Drop_n	Regulararization for negation operation	$\{0.05, 0.15, 0.25\}$
Entity dimmension d	Representation of entities	$\{400, 800, 1600\}$
Number of relation bases K	Representation of relations	$\{70, 90, 120\}$
Margin γ	Loss function	$\{37.5\}$
Scale ρ	Loss function	$\{90, 120, 150\}$
Size of diagonal block a	Representation of entities	{5, 10, 20}
Window size ω	Transport area of WFR distance	$\{1, 3, 5\}$
SinkHorn's reg ϵ	Entropy regularization of WFR distance	$\{0.1\}$
Sinkhorn's maximum iteration K_S	Sinkhorn algorithm's convergence	$\{10, 15, 30\}$

Table 4: Best hyperparameters on every dataset

	learning rate	training steps	K_{neg}	weight decay	Drop_p	Drop_n	d	K	γ	ρ	a	ω	ϵ	K_S
FB15k	0.0005	360000	32	0.01	0.05	0.1	1600	90	37.5	150	5	3	0.1	10
FB15k237	0.0005	240000	32	0.01	0.05	0.1	1600	120	37.5	120	5	3	0.1	10
NELL	0.0005	240000	32	0.01	0.05	0.1	1600	70	37.5	180	5	3	0.1	10
WN18RR	0.001	120000	32	0.01	0.05	0.1	800	70	37.5	120	5	3	0.1	15

based on local comparison with KL divergence

GammaE (Yang et al., 2022) GammaE distribution embedding whose scoring function is based on local comparison with KL divergence.

ConE (Zhang et al., 2021) 2D cone embedding whose scoring function is based on global transport with the rotational distance between cones.

MLP-MIX (Amayuelas et al., 2022) Vector embedding whose scoring function is based on local comparison with the Euclidean distance.

Q2P (Bai et al., 2022) Multi-particle embedding whose scoring function is based on global transport by comparing the target particle with the closest particle of a point cloud.

Those scores are directly taken from corresponding papers.

D.2 Combinational generalizability on queries

Wang et al. (2021) propose a new dataset including 301 different query types to benchmark the combinational generalizability of CQA models. Based on the EFO-1 queries represented by OpsTree, EFO-1 formulas are generated with operations including entity, projection, intersection, union, and negation. To make queries more realistic, the maximum length of projection/negation chains and the number of anchor nodes are both limited to no more than 3. The baselines are BetaE and LogicE (Luus et al., 2021). Scores are directly taken from Wang et al. (2021).

D.3 Complex queries on Hierarchical KG

WN18RR is first introduced as a link prediction dataset created from WN18 (Bordes et al., 2013b), which is a subset of WordNet. There are 93,003 triples with 40,943 entities and 11 relation types in WN18RR and most of the relations are hierarchical. In Table 6, we could know seven out of eleven relations have high antisymmetry $Khs_{\mathcal{G}_r}$ Huang et al. (2022) and negative transitive score $\xi_{\mathcal{G}_r}$ (Gu et al., 2018) and are regarded as hierarchical relations. Huang et al. (2022) extends complex logic queries to WN18RR and detaied queries stastics is in Table 5. Huang et al. (2022) generated 14 types of queries from hierarchical KG WN18RR and aimed to investigate the reasoning ability of query embeddings in hierarchical knowledge graphs. We choose BetaE and LinE (Huang et al., 2022) as baselines, their scores are also taken from Huang et al. (2022). Notably, LinE (Huang et al., 2022) is also a histogram-based query embedding method based on the same closed-form set operation. The key difference between LinE and WFRE is that WFRE encourages the local and global trade-offs.

Table 5: Number of training, validation, and test queries generated for different query structures.

Dataset	Training		Validaton		Test	
	1P/2P/3P/2I/3I	2IN/3IN/INP/PIN/PNI	1P	Others	1P	Others
FB15k	273,710	27,371	59,097	8,000	67,016	8,000
FB15k-237	149,689	14,968	20,101	5,000	22,812	5,000
NELL995	107,982	10,798	16,927	4,000	17,034	4,000
WN18RR	103,509	10,350	5,202	1,000	5,356	1,000

Table 6: Hierarchical relations in WN18RR

Relation	$Khs_{\mathcal{G}_r}$	$\xi_{\mathcal{G}_r}$	Hierarchical
memberMeronym	1.00	-2.90	✓
hypernym	1.00	-2.46	✓
hasPart	1.00	-0.82	✓
instance hypernym	1.00	-0.78	✓
memberOfDomainRegion	1.00	-0.78	✓
memberOfDomainUsage	1.00	-0.78	✓
synsetDomainTopicOf	0.99	-0.69	✓
alsoSee	0.36	-0.29	×
derivationally related form	0.07	-3.84	×
SimilarTo	0.07	-1.00	×
verb group	0.07	-0.50	×

E Modeling Union: DNF and DM

There are two ways to deal with union operations. With the De Morgan (DM) Law, it's natural to model union operation $S_1 \cup S_2$ with $\overline{\overline{S_1} \cap \overline{S_2}}$. (Ren et al., 2020) transforms queries into a disjunctive normal form (DNF) and only computes the union operation in the last step. Therefore, CQA models usually train intersection and complement logic operations. Though WFRE has closed union operation, WFRE with DNF has better performance as training queries don't contain union operation.

F Addtional results

Moreover, we further compare with two QE methods FuzzQE (Chen et al., 2022) and GammaE (Yang et al., 2022). Yang et al. (2022) develop a new union operation method with the self-attention mechanism and get better performance than DNF and DM. FuzzQE's result on FB15k is missing, and the suggested hyperparameters setting on FB15k-237 is missing. As we couldn't reproduce FuzzQE's result on NELL, we list the results in the paper and those reproduced by us. In Table 7, WFRE outperforms the two models except for the FuzzQE result in the paper.

Table 8 also provides the mean and standard derivation of the output of our model. All scores are computed from four runs of cases of different random seeds. We could see that the standard derivation is four orders smaller than the mean value. It shows that WFRE is very stable and significantly outperforms previous baselines.

Table 7: Additional benchmark comparison on FB15k, FB15k-237, and NELL(MRR).

Dataset	QE	1P	2P	3P	2I	3I	PI	IP	2U	UP	2IN	3IN	INP	PIN	PNI	A_{P}	A_{N}
FB15k	GammaE	76.5	36.9	31.4	65.4	75.1	53.9	39.7	57.1	34.5	20.1	20.5	13.5	11.8	17.1	52.3	16.6
	WFRE	81.1	37.7	30.5	68.5	78.0	56.3	41.8	48.0	33.1	26.1	26.5	15.6	13.7	19.4	52.8	20.2
FB15k-237	GammaE	43.2	13.2	11.0	33.5	47.9	27.2	15.9	15.4	11.3	6.7	9.4	8.6	4.8	4.4	24.3	6.8
	WFRE	44.1	13.4	11.1	35.1	50.1	27.4	17.2	13.9	10.9	6.9	11.2	8.5	5.0	4.3	24.8	7.2
NELL	GammaE	55.1	17.3	14.2	41.9	51.1	26.9	18.3	16.5	12.5	6.3	8.7	11.4	4.0	4.5	28.2	7.0
	FuzzQE(our)	55.5	16.8	14.4	37.3	46.9	24.0	19.1	15.0	11.7	7.3	9.1	11.1	4.1	4.9	26.7	7.3
	FuzzQE(reported)	58.1	19.3	15.7	39.8	50.3	28.1	21.8	17.3	13.7	8.3	10.2	11.5	4.6	5.4	29.3	8.0
	WFRE	58.6	18.6	16.0	41.2	52.7	28.4	20.7	16.1	13.2	6.9	8.8	12.5	4.1	4.4	29.5	7.3

Table 8: WFRE: metrics' mean values ($\times 10^{-2}$) and standard deviations ($\times 10^{-6}$, boldface).

Dataset	QE	1P	2P	3P	2I	3I	PI	IP	2U	UP	2IN	3IN	INP	PIN	PNI	A_{P}	A_{N}
FB15k	MRR	81.1 **0.073**	37.7 **0.33**	30.5 **0.39**	68.5 **0.45**	78.0 **1.5**	56.3 **0.37**	41.8 **1.6**	48.0 **1.3**	33.1 **2.5**	26.1 **0.74**	26.5 **1.2**	15.6 **1.8**	13.7 **0.37**	19.4 **2.8**	52.8 **0.16**	20.2 **0.046**
	HITS1	73.0 **0.096**	27.4 **0.35**	21.2 **2.0**	58.6 **1.6**	70.1 **5.4**	45.7 **1.0**	31.1 **1.8**	36.1 **2.5**	23.1 **8.1**	16.5 **1.0**	16.4 **1.4**	8.4 **1.3**	7.1 **0.32**	10.9 **4.8**	42.9 **0.67**	11.9 **0.11**
	HITS3	87.8 **0.18**	42.0 **1.3**	33.8 **0.35**	74.5 **2.0**	83.5 **0.69**	62.0 **1.9**	46.6 **9.8**	54.5 **3.5**	36.6 **2.6**	28.5 **0.18**	28.9 **6.8**	16.4 **3.0**	13.8 **0.16**	20.8 **1.7**	58.0 **0.11**	21.7 **0.32**
	HITS10	93.5 **0.38**	58.0 **6.5**	48.5 **1.2**	86.5 **0.60**	92.2 **0.95**	76.3 **0.062**	62.4 **1.1**	70.4 **0.60**	52.7 **2.9**	45.5 **0.97**	47.3 **0.20**	29.9 **0.91**	26.6 **4.3**	36.2 **0.16**	71.2 **0.0086**	37.1 **0.29**
FB15k237	MRR	44.1 **0.032**	13.4 **2.1**	11.1 **2.4**	35.1 **1.0**	50.1 **3.1**	27.4 **1.6**	17.2 **0.36**	13.9 **0.89**	10.9 **0.12**	6.9 **0.60**	11.2 **1.1**	8.5 **0.20**	5.0 **0.42**	4.3 **0.56**	24.8 **0.016**	7.2 **0.17**
	HITS1	33.7 **0.056**	7.3 **5.7**	5.5 **5.7**	23.9 **1.6**	39.6 **5.5**	18.5 **2.1**	10.8 **0.70**	7.5 **2.3**	5.2 **0.37**	2.8 **0.071**	5.2 **1.2**	3.7 **0.76**	1.6 **2.0**	1.4 **0.56**	16.9 **0.15**	2.9 **0.044**
	HITS3	48.9 **1.2**	13.8 **0.36**	11.3 **0.92**	39.7 **4.1**	55.2 **4.8**	29.9 **0.69**	18.0 **1.8**	14.0 **0.11**	11.0 **0.49**	6.3 **3.7**	10.8 **0.42**	8.2 **5.1**	4.4 **0.20**	3.6 **1.1**	27.0 **0.15**	6.7 **0.73**
	HITS10	64.5 **0.15**	25.6 **3.9**	21.9 **1.0**	57.8 **13**	71.0 **1.6**	45.6 **18**	29.8 **0.53**	23.1 **0.69**	17.5 **0.48**	14.4 **2.5**	23.1 **4.6**	17.5 **0.95**	10.9 **0.37**	9.0 **0.35**	40.5 **0.17**	15.0 **0.24**
NELL	MRR	58.6 **0.74**	18.8 **0.44**	16.0 **0.056**	41.2 **0.72**	52.7 **6.0**	28.4 **2.6**	20.7 **0.96**	16.1 **0.22**	13.2 **0.62**	6.9 **0.0026**	8.8 **0.24**	12.5 **1.6**	4.1 **0.046**	4.4 **0.047**	29.5 **0.52**	7.3 **0.095**
	HITS1	49.1 **1.3**	12.6 **0.27**	10.6 **0.40**	29.5 **12**	41.4 **0.77**	20.6 **5.3**	14.1 **0.34**	9.4 **1.1**	7.8 **0.29**	2.3 **0.12**	3.2 **0.16**	6.1 **1.9**	1.1 **0.24**	1.4 **0.19**	21.7 **0.44**	2.8 **0.25**
	HITS3	64.2 **1.3**	19.9 **8.9**	16.8 **1.6**	46.5 **2.4**	58.3 **2.9**	30.8 **3.5**	22.2 **4.1**	17.2 **5.9**	13.8 **2.1**	6.0 **0.25**	7.6 **0.70**	12.9 **3.2**	3.2 **0.10**	3.7 **0.73**	32.2 **1.71**	6.7 **0.18**
	HITS10	76.1 **1.1**	31.0 **6.2**	26.3 **6.5**	64.6 **5.1**	75.0 **1.6**	43.9 **1.3**	33.7 **6.3**	29.4 **3.0**	24.0 **0.27**	15.6 **0.49**	19.7 **0.062**	24.8 **1.9**	8.6 **1.2**	9.1 **1.5**	44.5 **0.33**	15.5 **0.24**

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Left blank.

☑ A2. Did you discuss any potential risks of your work?
Left blank.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Left blank.

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Left blank.

☑ B1. Did you cite the creators of artifacts you used?
Left blank.

□ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Not applicable. Left blank.

□ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Not applicable. Left blank.

□ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

□ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Not applicable. Left blank.

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Left blank.

C ☑ Did you run computational experiments?

Left blank.

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Left blank.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Left blank.

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Left blank.

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Left blank.

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

□ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Not applicable. Left blank.

□ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Not applicable. Left blank.

□ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Not applicable. Left blank.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.