Stability and Generalization Capability of Subgraph Reasoning Models for Inductive Knowledge Graph Completion

Minsung Hwang [1] Jaejun Lee [1] Joyce Jiyoung Whang [1]

Abstract

Inductive knowledge graph completion aims to predict missing triplets in an incomplete knowledge graph that differs from the one observed during training. While subgraph reasoning models have demonstrated empirical success in this task, their theoretical properties, such as stability and generalization capability, remain unexplored. In this work, we present the first theoretical analysis of the relationship between the stability and the generalization capability for subgraph reasoning models. Specifically, we define stability as the degree of consistency in a subgraph reasoning model's outputs in response to differences in input subgraphs and introduce the Relational Tree Mover's Distance as a metric to quantify the differences between the subgraphs. We then show that the generalization capability of subgraph reasoning models, defined as the discrepancy between the performance on training data and test data, is proportional to their stability. Furthermore, we empirically analyze the impact of stability on generalization capability using real-world datasets, validating our theoretical findings.

1. Introduction

Knowledge graphs (KGs) represent real-world knowledge by modeling relationships between entities as triplets (Wang et al., 2017). Due to their inherent incompleteness, numerous studies have focused on knowledge graph completion (KGC), which aims to predict missing triplets within KGs (Bordes et al., 2013; Schlichtkrull et al., 2018). Conventional models for KGC operate under a transductive setting, where a KG to be completed during inference is identical to the one observed during training (Bordes et al., 2013). These models typically learn representations of the observed entities and utilize the learned representations to predict missing links (Chung et al., 2023; Lee et al., 2023a).

As real-world KGs continually expand, recent research has shifted its focus to inductive KGC, where the KG that appears during inference differs from the one used for training (Teru et al., 2020). Most models designed for transductive KGC are unsuitable for inductive inference because they cannot handle unobserved entities, as these entities lack learned representations. To overcome this limitation, recent studies have adopted subgraph reasoning approaches (Teru et al., 2020; Zhu et al., 2021; Zhang & Yao, 2022). Unlike models that learn representations for individual entities and use the learned representations for link prediction, subgraph reasoning models determine the validity of a triplet by utilizing the structure of the subgraph extracted around the triplet. Consequently, these models can effectively perform inductive link prediction using the subgraph structures extracted from the inference KG.

To theoretically explain the empirical success of KGC models, several analyses have been conducted. Most of these studies focus on understanding what these models are capable of learning. For example, some investigate the graph structures that the models can distinguish (Barcelo et al., 2022; Huang et al., 2023), while others examine the rules within a KG that the models can infer (Qiu et al., 2024). In contrast, relatively little attention has been given to two critical theoretical properties of subgraph reasoning models: (1) generalization capability, which refers to the discrepancy between a model's performance on training and test data, and (2) stability, defined as the degree to which a model's output varies in response to the changes in input. Although a recent study examined the generalization capability of knowledge graph representation learning models in a transductive setting (Lee et al., 2024), the stability and the generalization capability of subgraph reasoning models for inductive KGC have not been studied.

In this paper, we present the first theoretical analysis of the relationship between stability and generalization capability of subgraph reasoning models designed for KGC. To comprehensively analyze existing subgraph reasoning models, we provide the framework that can represent existing subgraph reasoning models by decomposing subgraph rea-

[1] School of Computing, KAIST, Daejeon, South Korea. Correspondence to: Joyce Jiyoung Whang <jjwhang@kaist.ac.kr>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025. Copyright 2025 by the author(s).

soning models into two components: a subgraph extractor that extracts a subgraph associated with an input triplet from a KG and a Subgraph Message-Passing Neural Network (SMPNN) that computes the score of the input triplet using the extracted subgraph. The proposed framework can represent well-known subgraph reasoning models, such as GraIL (Teru et al., 2020), NBFNet (Zhu et al., 2021), and RED-GNN (Zhang & Yao, 2022). Within this framework, we show that the differences in output scores computed by subgraph reasoning models are bounded by the Relational Tree Mover's Distance (RTMD) between the input subgraphs, where RTMD is introduced to quantify the differences between subgraphs of KGs. We define the stability of a subgraph reasoning model as the ratio of the RTMD between the subgraphs to the difference between the scores of the subgraphs computed by its SMPNN. This measure is then used to derive a generalization bound of the subgraph reasoning models for inductive KGC.

Our theoretical analysis reveals that the stability of subgraph reasoning models is a key factor influencing the generalization capability. Furthermore, we empirically validate our theoretical findings by demonstrating that RTMD can be used to infer the labels of triplets in real-world datasets, that the differences between the output scores of SMPNN are bounded by RTMD of the input subgraphs, and that stable models tend to exhibit superior generalization capability.

Our contributions can be summarized as follows:

- We propose a general framework for subgraph reasoning models and derive their stability with respect to the perturbations of the subgraph structures.
- We introduce a pseudo-metric[1], RTMD, specifically designed for subgraph reasoning models, and use it to compute the stability of subgraph reasoning models.
- We theoretically analyze the generalization bound of subgraph reasoning models and discuss the impact of the stability on their generalization capability.
- We empirically show that RTMD is a suitable metric for subgraph reasoning models and examine how stability impacts generalization error on real-world KGs.

2. Related Work

Subgraph Reasoning Models for Inductive KGC Subgraph reasoning models have been proposed to predict missing links in graphs without relying on predefined rules (Zhang & Chen, 2018). These models extract a subgraph associated with a target link, relabel the nodes within the subgraph, and compute a score of the relabeled subgraph. Since these methods do not require the entities encountered during inference to have been observed during training, subgraph reasoning has been widely adopted for inductive KGC (Teru et al., 2020; Chen et al., 2021; Mai et al., 2021; Lin et al., 2022; Liu et al., 2023; Zhu et al., 2021; Zhang & Yao, 2022; Zhu et al., 2023; Zhang et al., 2023). For instance, GraIL (Teru et al., 2020) extracts an enclosing subgraph for a given triplet and labels the entities within the subgraph based on the shortest distance from the head and tail entities of the triplet. The model then computes a score for the subgraph using edge attention and an R-GCN encoder (Schlichtkrull et al., 2018).

Theoretical Analysis on KGs Various studies have been conducted to theoretically understand KGC models, focusing on their expressivity (Barcelo et al., 2022; Huang et al., 2023; Qiu et al., 2024). For instance, Barcelo et al. (2022) extended the WL test (Weisfeiler & Lehman, 1968) to multi-relational graphs to explore the expressivity of GNN-based KGC models. While the expressivity of KGC models is actively studied, the generalization capability has received less attention. Recently, Lee et al. (2024) introduced a framework called ReED that generalizes diverse knowledge graph representation learning methods and computed the generalization bounds of the ReED framework for the transductive setting, which is not applicable to the subgraph reasoning models for inductive KGC. On the other hand, we analyze subgraph reasoning models in the inductive setting, focusing on the stability and generalization capability of subgraph reasoning models.

Generalization Capability of Graph Neural Networks Some research has focused on the generalization capability of Graph Neural Networks (GNNs) using various approaches (Scarselli et al., 2018; Garg et al., 2020; Oono & Suzuki, 2020; Ma et al., 2021; Maskey et al., 2022; Zhou et al., 2022; Ju et al., 2023; Karczewski et al., 2024; Aminian et al., 2024). For example, Liao et al. (2021) computed the PAC-Bayesian generalization bound of GNNs for graph classification tasks. Furthermore, recent studies have analyzed the relationship between the generalization capability and other theoretical properties of GNNs (Morris et al., 2023; Franks et al., 2024; Chuang & Jegelka, 2022; Huang et al., 2024; Dong et al., 2024). For instance, Chuang & Jegelka (2022) derived the generalization bound of GIN (Xu et al., 2019) using the stability assessed by the Lipschitz continuity under the Tree Mover's Distance (TMD). In contrast, we extend TMD into a form applicable to subgraphs extracted from KGs and provide analysis applicable to various subgraph reasoning models.

3. Inductive KGC by Subgraph Reasoning

To provide a general analysis applicable to existing subgraph reasoning approaches, we formally define the subgraph rea-

[1] While RTMD is a pseudo-metric since there exists a pair of distinct subgraphs for which RTMD between them is 0, we refer to it as a metric for convenience in this paper.

soning models for inductive KGC.

3.1. Inductive Knowledge Graph Completion

A knowledge graph $G = (\mathcal{V}, \mathcal{R}, \mathcal{F} \cup \mathcal{T})$ consists of a set of entities $\mathcal{V}$, a set of relations $\mathcal{R}$, a set of triplets $\mathcal{F} \subseteq \mathcal{V} \times \mathcal{R} \times \mathcal{V}$ assumed to be known, and another set of triplets $\mathcal{T} \subseteq \mathcal{V} \times \mathcal{R} \times \mathcal{V}$ for prediction, where $\mathcal{F} \cap \mathcal{T} = \emptyset$. Each triplet (h, r, t) is associated with a label $y_{hrt} \in \{-1, +1\}$. A label of $+1$ denotes a positive triplet, indicating the given triplet is true, whereas a label of -1 denotes a negative triplet, indicating the given triplet is false.

KGC is the task of predicting the labels of triplets in $\mathcal{T}$ based on the triplets in $\mathcal{F}$. In inductive KGC, a model is trained using a training knowledge graph $G_{\text{tr}} = (\mathcal{V}_{\text{tr}}, \mathcal{R}, \mathcal{F}_{\text{tr}} \cup \mathcal{T}_{\text{tr}})$ and conducts inference on an inference knowledge graph $G_{\text{inf}} = (\mathcal{V}_{\text{inf}}, \mathcal{R}, \mathcal{F}_{\text{inf}} \cup \mathcal{T}_{\text{inf}})$. Note that transductive KGC is a special case of inductive KGC, where $\mathcal{V}_{\text{tr}} = \mathcal{V}_{\text{inf}}$ and $\mathcal{F}_{\text{tr}} = \mathcal{F}_{\text{inf}}$.

3.2. Subgraph Reasoning Models for Inductive KGC

A subgraph reasoning model uses a subgraph extractor to extract a subgraph corresponding to a target triplet included in $\mathcal{T}$ and predicts the label of the target triplet using a scoring function that takes the extracted subgraph as input. We formally define a subgraph extractor as follows.

Definition 3.1 (Subgraph Extractor)**.** Given a knowledge graph $G = (\mathcal{V}, \mathcal{R}, \mathcal{F} \cup \mathcal{T})$, a subgraph extractor $g(G, (h, q, t)) = S$ is a non-parametric function that utilizes $\mathcal{F}$ to map a triplet $(h, q, t) \in \mathcal{T}$ into a subgraph S. A subgraph S is denoted as $S = (\mathcal{V}_S, \mathcal{E}_S, \mathcal{R}, \text{INIT}_S, (h, q, t))$, where $\mathcal{V}_S$ and $\mathcal{E}_S$ denote the sets of entities and triplets of the subgraph S, respectively. The function $\text{INIT}_S : \mathcal{V}_S \to \mathbb{R}^{d_0}$ generates an initial entity embedding vector with dimension d_0 for each entity. The triplet (h, q, t) is referred to as the query triplet, and q is referred to as the query relation of the subgraph S. Note that h and t are always included in $\mathcal{V}_S$. For each entity v in $\mathcal{V}_S$, the multiset of v's neighbors is defined as $\mathcal{N}_S(v) = \{\{(r, u) \mid (u, r, v) \in \mathcal{E}_S\}\}$.

A common form of subgraphs extracted by a subgraph extractor is an enclosing subgraph constructed by the intersection of the k-hop neighbor entities of h and t for each query triplet (h, q, t) (Teru et al., 2020). Given a subgraph extracted by a subgraph extractor, a scoring function of a subgraph reasoning model computes a score of the subgraph through a message-passing. In Definition 3.2, we propose a framework called Subgraph Message-Passing Neural Networks (SMPNNs), which generalizes the scoring functions that utilize the message-passing.

Definition 3.2 (Subgraph Message-Passing Neural Networks)**.** Given a subgraph $S = (\mathcal{V}_S, \mathcal{E}_S, \mathcal{R}, \text{INIT}_S, (h, q, t))$, a Subgraph Message-Passing Neural Network (SMPNN) $f_{\mathbf{w}}$ with parameters $\mathbf{w}$ is defined by

$$\mathbf{x}_S^{(0)}(v) = \text{INIT}_S(v)$$

$$\mathcal{M}_S^{(l)}(v) = \{\{\text{MSG}^{(l)}(\mathbf{x}_S^{(l-1)}(u), \mathbf{x}_S^{(l-1)}(v), r, q) \mid (r, u) \in \mathcal{N}_S(v)\}\}$$

$$\mathbf{x}_S^{(l)}(v) = \text{UPD}^{(l)}\left(\mathbf{x}_S^{(\theta(l))}(v), \text{AGG}^{(l)}(\mathcal{M}_S^{(l)}(v))\right)$$

$$f_{\mathbf{w}}(S) = \text{RD}\left(\mathbf{x}_S^{(L)}(h), \mathbf{x}_S^{(L)}(t), \text{GRD}(\{\{\mathbf{x}_S^{(L)}(u) \mid u \in \mathcal{V}_S\}\}), q\right)$$

where $\mathbf{x}_S^{(l)}(v) \in \mathbb{R}^d$ is an embedding vector of an entity $v \in \mathcal{V}_S$, $\text{MSG}^{(l)} : \mathbb{R}^d \times \mathbb{R}^d \times \mathcal{R} \times \mathcal{R} \to \mathbb{R}^d$ is a *message* function, $\mathcal{M}_S^{(l)}(v)$ is a multiset of messages propagated to the entity v, $\text{AGG}^{(l)} : 2^{\mathbb{R}^d} \to \mathbb{R}^d$ is an *aggregation* function, $\text{UPD}^{(l)} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d$ is an *update* function, $\text{GRD} : 2^{\mathbb{R}^d} \to \mathbb{R}^d$ is a *global-readout* function, $\text{RD} : \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}^d \times \mathcal{R} \to \mathbb{R}$ is a *readout* function that computes the score of the subgraph, and θ is a *history* function with $\theta(k) = k - 1$ or $\theta(k) = 0$.

A subgraph reasoning model consists of a subgraph extractor g and an SMPNN $f_{\mathbf{w}}$. By appropriately configuring the subgraph extractor and the functions within the SMPNN, well-known subgraph reasoning models such as GraIL (Teru et al., 2020), NBFNet (Zhu et al., 2021), and RED-GNN (Zhang & Yao, 2022) can be subsumed within our framework. Detailed explanations are provided in Appendix A.

4. Stability of Subgraph Reasoning Models

The stability of a model refers to the degree of consistency in its outputs with respect to the changes in its inputs (Bousquet & Elisseeff, 2002; Shalev-Shwartz et al., 2010). Since the subgraph extractor of a subgraph reasoning model is non-parametric and discrete, we define the stability of a subgraph reasoning model in terms of the stability of its SMPNN. To quantify the stability of SMPNNs, it is necessary to measure the differences between subgraphs, as direct comparisons between subgraphs are hard to quantify. Therefore, we introduce the Relational Tree Mover's Distance (RTMD) which measures differences between subgraphs while reflecting the message-passing mechanism of SMPNNs. Specifically, the RTMD between two subgraphs is computed as the optimal transport distance between the sets of relational computation trees of the subgraphs, where the relational computation tree represents how SMPNNs perform message-passing on the subgraphs.

4.1. Optimal Transport

Optimal transport distance (Villani et al., 2009; Cuturi, 2013) is a measure used to quantify the discrepancy between two probability distributions. For two multisets $\mathcal{A} = \{\{a_i\}\}_{i=1}^m, \mathcal{B} = \{\{b_j\}\}_{j=1}^m$, a matrix $\boldsymbol{P}_{\mathcal{A},\mathcal{B}} \in \mathbb{R}_+^{m \times m}$ is referred to as a transportation plan if it satisfies

$$\boldsymbol{P}_{\mathcal{A},\mathcal{B}} \mathbf{1}_m = \boldsymbol{P}_{\mathcal{A},\mathcal{B}}^\top \mathbf{1}_m = \mathbf{1}_m$$

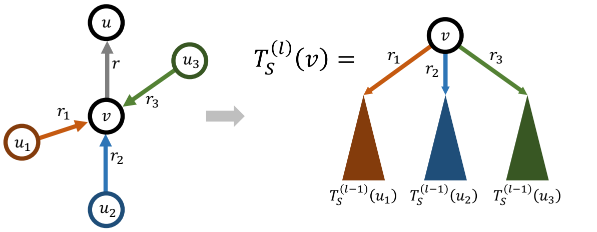


Figure 1. A visualization of the construction of a depth-l relational computation tree $T_S^{(l)}(v)$ for the entity v. The root entity of each subtree and the relation pointing to that subtree are derived from the multiset of v's incoming neighbors $\mathcal{N}_S(v)$.

where $\mathbf{1}_m \in \mathbb{R}^m$ is a one vector. Given a cost function $\phi : \mathcal{A} \times \mathcal{B} \to \mathbb{R}$ that maps each pair $(a_i, b_j) \in \mathcal{A} \times \mathcal{B}$ to the transportation cost between a_i and b_j, the optimal transport distance between $\mathcal{A}$ and $\mathcal{B}$ is defined as:

$$\mathrm{OT}_\phi(\mathcal{A}, \mathcal{B}) = \min_{\boldsymbol{P}_{\mathcal{A},\mathcal{B}}} \sum_{i,j}^{m} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i, j]\phi[i, j] \qquad (1)$$

The transportation plan that provides the optimal solution for Eq. 1 is referred to as the optimal transportation plan.

4.2. Relational Tree Mover's Distance

Tree Mover's Distance (TMD) (Chuang & Jegelka, 2022) is a metric designed to quantify differences between graphs, particularly for analyzing the stability of message-passing neural networks. However, TMD cannot be directly applied to analyze SMPNNs, as it only accounts for node features and graph structure, whereas the input subgraphs of SMPNNs are associated with triplets and contain edges labeled by relations. To address this issue, we propose Relational Tree Mover's Distance (RTMD), which incorporates initial entity embeddings, edge labels, subgraph structures, and the triplets associated with the input subgraphs to compute the differences between subgraphs. Specifically, RTMD captures subgraph structures by modeling how SMPNNs process these structures, which is formalized through the concept of a relational computation tree, as defined in Definition 4.1.

Definition 4.1 (Relational Computation Tree)**.** Given a subgraph $S = (\mathcal{V}_S, \mathcal{E}_S, \mathcal{R}, \mathrm{INIT}_S, (h, q, t))$ and $l > 0$, the depth-l relational computation tree $T_S^{(l)}(v)$ of an entity $v \in \mathcal{V}_S$ is defined by

$$T_S^{(0)}(v) = (v, \emptyset)$$
$$\mathrm{SUB}(T_S^{(l)}(v)) = \{\{(r, T_S^{(l-1)}(u)) | (r, u) \in \mathcal{N}_S(v)\}\}$$
$$T_S^{(l)}(v) = \left(v, \mathrm{SUB}(T_S^{(l)}(v))\right)$$

where $\mathrm{SUB}(T_S^{(l)}(v))$ is a multiset of subtrees of $T_S^{(l)}(v)$. The entity v is referred to as the root entity of $T_S^{(l)}(v)$.

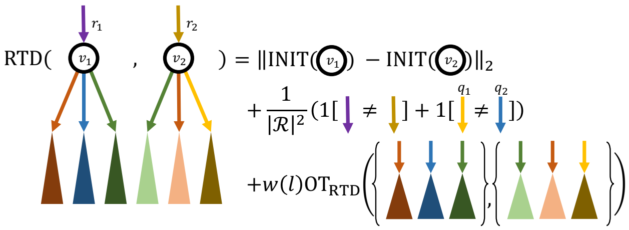

RTD(
r_1
v_1
,
r_2
v_2
) = ||INIT(v_1) − INIT(v_2)||₂
+ $\frac{1}{|\mathcal{R}|^2}$(1[≠] + 1[≠])
q_1
q_2
+w(l)OT_RTD

Figure 2. A visualization of computation of the Relational Tree Distance (RTD) between two subtrees of relational computation trees. The RTD is the sum of the difference between initial entity embeddings, the penalties for different relations and queries, and the optimal transport distance between the multisets of subtrees.

As illustrated in Figure 1, the relational computation tree is constructed by recursively adding neighboring relations and entities to the leaf nodes, which aligns with the process of updating entity embedding vectors in SMPNNs. Therefore, the relational computation tree of an entity v represents the computation tree of SMPNNs for v.

To define the difference between the relational computation trees of two entities, we consider three factors: (i) the difference between the initial embedding vectors of their root entities, (ii) the difference between the multisets of their subtrees, and (iii) whether their query relations differ. However, to measure the difference between the multisets of subtrees, it is necessary to first define the difference between individual subtrees. Since each subtree consists of a relation and a relational computation tree, the difference between two subtrees, $(r_1, T_{S_1}^{(l_1)}(v_1))$ and $(r_2, T_{S_2}^{(l_2)}(v_2))$, is determined by two factors: (i) the difference between the relational computation trees, $T_{S_1}^{(l_1)}(v_1)$ and $T_{S_2}^{(l_2)}(v_2)$, and (ii) whether their relations, r_1 and r_2, differ. This difference is referred to as the relational tree distance, which is formally defined in Definition 4.2. Note that the difference between the relational computation trees is a special case of relational tree distance, obtained by introducing a virtual relation r_{root} shared across all relational computation trees.

Definition 4.2 (Relational Tree Distance)**.** Given two subgraphs $S_1 = (\mathcal{V}_{S_1}, \mathcal{E}_{S_1}, \mathcal{R}, \mathrm{INIT}_{S_1}, (h_1, q_1, t_1))$, $S_2 = (\mathcal{V}_{S_2}, \mathcal{E}_{S_2}, \mathcal{R}, \mathrm{INIT}_{S_2}, (h_2, q_2, t_2))$ and $l_1, l_2 > 0$, for $v_1 \in \mathcal{V}_{S_1}$, $v_2 \in \mathcal{V}_{S_2}$ and $r_1, r_2 \in \mathcal{R}$, the relational tree distance between $(r_1, T_{S_1}^{(l_1)}(v_1))$ and $(r_2, T_{S_2}^{(l_2)}(v_2))$ is defined as

$$\mathrm{RTD}\left((r_1, T_{S_1}^{(l_1)}(v_1)), (r_2, T_{S_2}^{(l_2)}(v_2))\right) =$$
$$\|\mathrm{INIT}_{S_1}(v_1) - \mathrm{INIT}_{S_2}(v_2)\|_2 + \frac{1}{|\mathcal{R}|^2}\left(\mathbb{1}[r_1 \neq r_2] + \mathbb{1}[q_1 \neq q_2]\right)$$
$$+ w(\max(l_1, l_2))\mathrm{OT}_{\mathrm{RTD}}\left(\rho(\mathrm{SUB}(T_{S_1}^{(l_1)}(v_1)), \mathrm{SUB}(T_{S_2}^{(l_2)}(v_2)))\right)$$

where $T_S^{(l)}(v)$ is the depth-l relational computation tree of entity v for subgraph S, $w(l)$ is a weight function for the distance between multisets of subtrees based on the depth of the subtrees, and ρ denotes the blank tree augmentation defined in Definition B.1.

To quantify the difference between two multisets, we use the optimal transport distance. However, since the optimal transport distance requires the multisets to have equal sizes, and the sizes of the two multisets of neighbors may differ, we augment the smaller set by adding blank trees to match the size. This process is referred to as blank tree augmentation, detailed in Appendix B. Figure 2 illustrates how the relational tree distance between two subtrees is computed.

Finally, we define the RTMD between two subgraphs by considering (i) the difference between the head entities of the query triplets, (ii) the difference between the tail entities of the query triplets, and (iii) the difference between the multisets of relational computation trees of all entities in the subgraphs. The differences between the head and tail entities are measured using the relational tree distances between their relational computation trees, while the difference between the multisets of relational computation trees is quantified by the optimal transport distance. RTMD is formally defined in Definition 4.3.

Definition 4.3 (Relational Tree Mover's Distance)**.** Given two subgraphs $S_1 = (\mathcal{V}_{S_1}, \mathcal{E}_{S_1}, \mathcal{R}, \text{INIT}_{S_1}, (h_1, q_1, t_1))$, $S_2 = (\mathcal{V}_{S_2}, \mathcal{E}_{S_2}, \mathcal{R}, \text{INIT}_{S_2}, (h_2, q_2, t_2))$ and $L > 0$, Relational Tree Mover's Distance (RTMD) between S_1 and S_2 is defined by

$$\begin{aligned}\text{RTMD}(S_1, S_2) = {} & \text{RTD}((r_{\text{root}}, T_{S_1}^{(L)}(h_1)), (r_{\text{root}}, T_{S_2}^{(L)}(h_2)))+ \\ & \text{RTD}((r_{\text{root}}, T_{S_1}^{(L)}(t_1)), (r_{\text{root}}, T_{S_2}^{(L)}(t_2)))+ \\ & \text{OT}_{\text{RTD}}(\{\{(r_{\text{root}}, T_{S_1}^{(L)}(v_1)) | v_1 \in \mathcal{V}_{S_1}\}\}, \\ & \{\{(r_{\text{root}}, T_{S_2}^{(L)}(v_2)) | v_2 \in \mathcal{V}_{S_2}\}\})\end{aligned}$$

where $T_S^{(L)}(v)$ is the depth-L relational computation tree of entity v for subgraph S, and r_{root} is a virtual relation.

While the query relation is taken into account when computing the difference between relational computation trees, the head and tail entities of the query triplet are considered during the computation of RTMD. This distinction arises because SMPNNs incorporate the query relations during message-passing, whereas the head and tail entities serve solely as the identities of the subgraphs and are not directly utilized in the message-passing.

4.3. Stability of SMPNNs

Lipschitz continuity is a property of a function that holds when the difference between the outputs of a function is bounded by the difference between the inputs. Therefore, Lipschitz-continuous functions are also referred to as stable functions (Huang et al., 2024; Dong et al., 2024; Wang et al., 2022; Chuang & Jegelka, 2022). To define the stable SMPNNs, we define the Lipschitz continuity of SMPNNs with respect to the RTMD as follows.

Definition 4.4 (Lipschitz Continuity of SMPNNs)**.** Given $G_{\text{tr}} = (\mathcal{V}_{\text{tr}}, \mathcal{R}, \mathcal{F}_{\text{tr}} \cup \mathcal{T}_{\text{tr}})$ and $G_{\text{inf}} = (\mathcal{V}_{\text{inf}}, \mathcal{R}, \mathcal{F}_{\text{inf}} \cup \mathcal{T}_{\text{inf}})$, an SMPNN $f_{\mathbf{w}}$ with L layers is Lipschitz continuous if there exists a constant $\eta \geq 0$ such that $|f_{\mathbf{w}}(S_1) - f_{\mathbf{w}}(S_2)| \leq \eta\, \text{RTMD}(S_1, S_2)$ for any subgraphs $S_1, S_2 \in \{\{g(G_{\text{tr}}, e_1) | e_1 \in \mathcal{T}_{\text{tr}}\}\} \cup \{\{g(G_{\text{inf}}, e_2) | e_2 \in \mathcal{T}_{\text{inf}}\}\}$.

Since Lipschitz-continuous functions have a non-zero Lipschitz constant, we denote η_f as the Lipschitz constant of the $f_{\mathbf{w}}$. Then, a smaller η_f indicates that the score computed by the SMPNN is less affected by the distance between subgraphs, implying that the model is more stable. To ease of analysis, we quantify how stable an SMPNN is by defining its stability as $C_f = \frac{1}{\eta_f}$. Therefore, the higher C_f implies the lower Lipschitz constant and the more stable SMPNN.

Note that we assume the *message*, *aggregation*, *update*, *global-readout*, and *readout* functions of the SMPNNs to be Lipschitz continuous, which holds for the SMPNNs of many existing subgraph reasoning models such as GraIL, NBFNet, and RED-GNN, as detailed in Appendix A. Using the Lipschitz constant of each function, we derive the upper bound of the Lipschitz constant of the SMPNNs.

Theorem 4.5 (Lipschitz Constant of SMPNNs)**.** *Given $G_{\text{tr}} = (\mathcal{V}_{\text{tr}}, \mathcal{R}, \mathcal{F}_{\text{tr}} \cup \mathcal{T}_{\text{tr}})$, $G_{\text{inf}} = (\mathcal{V}_{\text{inf}}, \mathcal{R}, \mathcal{F}_{\text{inf}} \cup \mathcal{T}_{\text{inf}})$, and an* SMPNN *$f_{\mathbf{w}}$ with L layers, if the message, aggregation, update, global-readout, and readout functions of $f_{\mathbf{w}}$ are Lipschitz continuous, then $f_{\mathbf{w}}$ is Lipschitz continuous with the Lipschitz constant η_f and the following holds:*

$$\eta_f \leq \begin{cases} \left(\prod_{l=1}^{L+1} \eta^{(l)}\right) & \theta(k) = k - 1 \\ (L+1)\left(\prod_{l=1}^{L+1} \eta^{(l)}\right) & \theta(k) = 0 \end{cases}$$

$$\begin{aligned}\eta^{(l)} = {} & \max(A_{\text{upd}}^{(l)} + d_{\max} B_{\text{upd}}^{(l)} A_{\text{agg}}^{(l)} B_{\text{msg}}^{(l)}, B_{\text{upd}}^{(l)} A_{\text{agg}}^{(l)} A_{\text{msg}}^{(l)}, \\ & |\mathcal{R}|^2 B_{\text{upd}}^{(l)} A_{\text{agg}}^{(l)} C_{\text{msg}}^{(l)}, |\mathcal{R}|^2 B_{\text{upd}}^{(l)} A_{\text{agg}}^{(l)} D_{\text{msg}}^{(l)}, 1), \\ \eta^{(L+1)} = {} & \max(A_{\text{rd}}, B_{\text{rd}}, C_{\text{rd}} A_{\text{grd}}, \frac{|\mathcal{R}|^2 D_{\text{rd}}}{2 + \max(|\mathcal{V}_{\text{tr}}|, |\mathcal{V}_{\text{inf}}|)})\end{aligned}$$

where $1 \leq l \leq L$, $A_{msg}^{(l)}, B_{msg}^{(l)}, C_{msg}^{(l)}, D_{msg}^{(l)}$ are the Lipschitz constants of the message function, $A_{agg}^{(l)}$ is the Lipschitz constant of the aggregation function, $A_{upd}^{(l)}, B_{upd}^{(l)}$ are the Lipschitz constants of the update function, A_{grd} is the Lipschitz constant of the global-readout function, $A_{rd}, B_{rd}, C_{rd}, D_{rd}$ are the Lipschitz constants of the readout function, and $d_{\max}$ is the maximum degree of G_{tr} and G_{inf}.

We provide a proof for Theorem 4.5 in Appendix C. According to Theorem 4.5 and Definition 4.4, the stability increases as the Lipschitz constants of the functions in the SMPNNs become smaller. Therefore, we can compare the stability of subgraph reasoning models by comparing the Lipschitz constants of these functions. For instance, while the Lipschitz constant of the *aggregation* function that uses a sum aggregator is 1, the Lipschitz constant of the *aggregation* function that uses a mean aggregator is reciprocal of the maximum degree of the KG. This indicates that the subgraph reasoning model using a mean aggregator as the *aggregation* function is more stable than the model using a sum aggregator as the *aggregation* function. Detailed calculations of the Lipschitz constants for each function are provided in Appendix A.

5. PAC-Bayesian Generalization Bound of Stable Subgraph Reasoning Models for Inductive KGC

We present the first PAC-Bayesian generalization bound of subgraph reasoning models for inductive KGC and relate their stability to their generalization capability.

5.1. Assumptions

We make the following assumptions to derive the generalization bound of the subgraph reasoning models.

A.1 Label y of a triplet e in a knowledge graph G is drawn from a distribution conditional to the subgraph corresponding to e, i.e., $y \sim \mathbb{P}(Y|g(G,e))$, where Y is a random variable that follows a binomial distribution.

A.2 In a knowledge graph $G = (\mathcal{V}, \mathcal{R}, \mathcal{F} \cup \mathcal{T})$, the set of triplets $\mathcal{F}$ contains sufficient information for predicting the labels of the triplets in $\mathcal{T}$.

Assumption 1 follows from the underlying assumption of the subgraph reasoning models, and Assumption 2 ensures that the subgraph reasoning model can predict the label of the subgraph based solely on the given KG.

5.2. PAC-Bayesian Generalization Bound for Inductive Knowledge Graph Completion

To measure the risk of a subgraph reasoning model, we employ a γ-margin risk. The γ-margin risk increases when a score for a positive triplet is less than or equal to γ or when a score for a negative triplet is greater than or equal to $-\gamma$. Note that the γ-margin loss, a differentiable approximation of a γ-margin risk, is a commonly used loss function for inductive KGC (Teru et al., 2020).

Definition 5.1 (γ-margin Risk)**.** Given $G = (\mathcal{V}, \mathcal{R}, \mathcal{F} \cup \mathcal{T})$ and a subgraph reasoning model with a subgraph extractor g and an SMPNN $f_\mathbf{w}$ with parameters $\mathbf{w}$, for any $\gamma \geq 0$, its empirical γ-margin risk on G is defined by

$$\widehat{\mathcal{L}}_G(f_\mathbf{w}, \gamma) = \frac{1}{|\mathcal{T}|} \sum_{(h,r,t)\in\mathcal{T}} \mathbb{1}\left[y_{\text{hrt}} f_\mathbf{w}(g(G,(h,r,t))) \leq \gamma\right]$$

where $\mathbb{1}[\cdot]$ is an indicator function. The expected γ-margin risk is defined as the expectation of the empirical risk:

$$\mathcal{L}_G(f_\mathbf{w}, \gamma) = \mathbb{E}_{y_{\text{hrt}} \sim \mathbb{P}(Y|g(G,(h,r,t)))}\left[\widehat{\mathcal{L}}_G(f_\mathbf{w}, \gamma)\right]$$

To assess the generalization capability of a subgraph reasoning model, we derive a generalization bound which is defined as the upper bound of a generalization error. The generalization error of a subgraph reasoning model with an SMPNN $f_\mathbf{w}$ is defined by $\mathcal{L}_{G_\text{inf}}(f_\mathbf{w}, 0) - \widehat{\mathcal{L}}_{G_\text{tr}}(f_\mathbf{w}, \gamma)$ where G_tr is a training KG and G_inf is an inference KG. We use the expected 0-margin risk since we are interested in the classification accuracy on the inference KG. Additionally, since the training KG and the inference KG are different in the inductive setting, the generalization error depends on the expected risk discrepancy in Definition 5.2.

Definition 5.2 (Expected Risk Discrepancy)**.** Given $G_\text{tr}, G_\text{inf}$, a subgraph reaoning model with a subgraph extractor g and an SMPNN $f_\mathbf{w}$, and a distribution $\mathcal{P}$ on the parameter space of $f_\mathbf{w}$, for any $\lambda > 0$ and $\gamma \geq 0$, the expected risk discrepancy between G_tr and G_inf with respect to $\mathcal{P}$, λ, and γ is defined by

$$\begin{aligned} &D(\mathcal{P}, \lambda, \gamma) \\ =&\ln\left(\mathbb{E}_{\mathbf{w}\sim\mathcal{P}}\left[\exp\left(\lambda\left(\mathcal{L}_{G_\text{tr}}(f_\mathbf{w}, \frac{\gamma}{2}) - \mathcal{L}_{G_\text{inf}}(f_\mathbf{w}, \gamma)\right)\right)\right]\right) \end{aligned}$$

where $\exp$ is an exponential function.

We utilize the PAC-Bayesian approach which computes the generalization bound based on the KL divergence between the posterior distribution $\mathcal{Q}$ and the prior distribution $\mathcal{P}$ over the parameter space. The prior distribution is defined independently of the training dataset, while the posterior distribution is the distribution after training. Originally, the PAC-Bayesian approach was used to derive generalization bounds for stochastic models, while most subgraph reasoning models are deterministic. Therefore, we present the PAC-Bayesian generalization bound of deterministic subgraph reasoning models in Theorem 5.3 by extending the PAC-Bayesian generalization bound for stochastic classifiers introduced by Ma et al. (2021).

Theorem 5.3 (PAC-Bayesian Generalization Bound of Deterministic Subgraph Reasoning Models)**.** *Given* G_tr= $(\mathcal{V}_\text{tr}, \mathcal{R}, \mathcal{F}_\text{tr} \cup \mathcal{T}_\text{tr})$, $G_\text{inf} = (\mathcal{V}_\text{inf}, \mathcal{R}, \mathcal{F}_\text{inf} \cup \mathcal{T}_\text{inf})$, *and a subgraph reasoning model with a subgraph extractor* g *and an* SMPNN $f_\mathbf{w}$*, for any prior distribution* $\mathcal{P}$ *and posterior distribution* $\mathcal{Q}$ *on the parameter space of* $f_\mathbf{w}$ *constructed by adding random noise* $\ddot{\mathbf{w}}$ *to* $\mathbf{w}$ *such that* $\mathbb{P}\big(\max(\max_{e\in\mathcal{T}_\text{tr}} |f_{\widetilde{\mathbf{w}}}(g(G_\text{tr}, e)) - f_\mathbf{w}(g(G_\text{tr}, e))|, \max_{e\in\mathcal{T}_\text{inf}} |f_{\widetilde{\mathbf{w}}}(g(G_\text{inf}, e)) - f_\mathbf{w}(g(G_\text{inf}, e))|) < \frac{\gamma}{4}\big) > \frac{1}{2}$*, and* $\gamma > 0$, $\lambda > 0$*, the following holds with proba-*

bility at least $1-\delta$*:*

$$\mathcal{L}_{G_{\text{inf}}}(f_{\mathbf{w}},0) \le \widehat{\mathcal{L}}_{G_{\text{tr}}}(f_{\mathbf{w}},\gamma) + \frac{1}{\lambda}\left(2KL(\mathcal{Q}\|\mathcal{P}) + \ln\frac{4}{\delta} + \frac{\lambda^2}{4|\mathcal{T}_{\text{tr}}|} + D\left(\mathcal{P},\lambda,\frac{\gamma}{2}\right)\right)$$

where $D\big(\mathcal{P},\lambda,\frac{\gamma}{2}\big)$ *is the expected risk discrepancy between* G_{tr} *and* G_{inf} *and* $KL(\mathcal{Q}\|\mathcal{P})$ *is a KL divergence of* $\mathcal{Q}$ *from* $\mathcal{P}$*.*

We derive Theorem 5.3 by approximating a deterministic model by introducing a stochastic model generated by adding random perturbations $\ddot{\mathbf{w}}$ to the fixed parameter $\mathbf{w}$, which is a common technique for translating a deterministic model into a stochastic model (Liao et al., 2021; Ma et al., 2021; Neyshabur et al., 2018; Lee et al., 2024). A detailed proof is provided in Appendix D.1.

5.3. Upper Bound of the Expected Risk Discrepancy

To discuss the generalization capability of subgraph reasoning models, we analyze the generalization bound presented in Theorem 5.3 which depends on both the KL divergence and the expected risk discrepancy $D(\mathcal{P},\lambda,\gamma)$. The KL divergence represents how far the learned distribution is from the prior distribution on the parameter space, and it increases as the complexity of the model grows, e.g., increasing the norm of learnable weight matrices or the number of layers. This term arises from the definition of the PAC-Bayesian approach (McAllester, 2003) and is also present in other PAC-Bayesian generalization bounds (Liao et al., 2021; Lee et al., 2024). In contrast, the expected risk discrepancy $D(\mathcal{P},\lambda,\gamma)$ is a key factor that affects our generalization bound for inductive KGC, where the training KG and inference KG are separately defined. Therefore, to focus on the discrepancy between the training KG and the inference KG, we derive the upper bound of the expected risk discrepancy in Theorem 5.4 and provide its implications.

Theorem 5.4 (Bound of $D(\mathcal{P},\lambda,\gamma)$)**.** *Given* $G_{\text{tr}} = (\mathcal{V}_{\text{tr}},\mathcal{R},\mathcal{F}_{\text{tr}}\cup\mathcal{T}_{\text{tr}})$*,* $G_{\text{inf}} = (\mathcal{V}_{\text{inf}},\mathcal{R},\mathcal{F}_{\text{inf}}\cup\mathcal{T}_{\text{inf}})$*, and a subgraph reasoning model with a subgraph extractor* g *and an* SMPNN $f_{\mathbf{w}}$ *with stability* C_f*, for any prior distribution* $\mathcal{P}$ *and posterior distribution* $\mathcal{Q}$ *on the parameter space of* $f_{\mathbf{w}}$*, and* $\lambda>0$*, the following holds:*

$$D(\mathcal{P},\lambda,\gamma) \le \lambda\left(\max(0, \frac{|\mathcal{T}_{\text{tr}}|}{|\mathcal{T}_{\text{inf}}|} - 1) + \frac{2\ \mathrm{OT}_{\mathrm{RTMD}}(\psi(\mathcal{T}_{\text{inf}},\mathcal{T}_{\text{tr}}))}{\gamma C_f \max(|\mathcal{T}_{\text{inf}}|,|\mathcal{T}_{\text{tr}}|)}\right)$$

where ψ *is the empty subgraph augmentation defined in Definition D.2.*

The term $\mathrm{OT}_{\mathrm{RTMD}}(\psi(\mathcal{T}_{\text{inf}},\mathcal{T}_{\text{tr}}))$ is the optimal transport distance between the multisets $\{\{g(G_{\text{tr}},e_i)|e_i \in \mathcal{T}_{\text{tr}}\}\}$ and $\{\{g(G_{\text{inf}},e_i)|e_i \in \mathcal{T}_{\text{inf}}\}\}$. Since the sizes of two sets of triplets differ, empty subgraph augmentation ψ adds empty subgraphs to the smaller multiset of subgraphs. We provide the proof for Theorem 5.4 in Appendix D.2

In Theorem 5.4, the upper bound of the expected risk discrepancy increases as the optimal transport distance between the multisets of subgraphs extracted from the training KG and the inference KG increases. A large optimal transport distance between the multisets of subgraphs indicates a more significant difference between the distributions of subgraph structures. This is consistent with existing analyses, which show that the generalization capability decreases as the difference between source and target distribution increases (Chuang & Jegelka, 2022; Shen et al., 2018). Furthermore, as the stability C_f of the SMPNN increases, the expected risk discrepancy decreases. A large C_f implies a smaller difference in the scores calculated for subgraphs with small distances, indicating a more stable model. Thus, a subgraph reasoning model with a more stable SMPNN tends to exhibit a higher generalization capability.

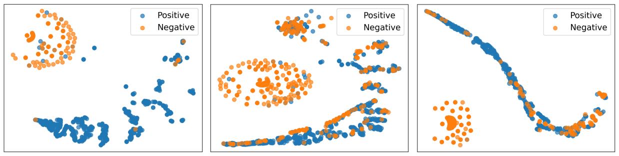
6. Experiments

In the previous section, we provided our theoretical findings on the stability and generalization capability of subgraph reasoning models for inductive KGC. To empirically validate our findings, we conduct experiments on real-world KGs using the inductive KGC datasets provided in Teru et al. (2020). Specifically, we use v3 of WN18RR (WNv3), v1 of FB15K-237 (FBv1), and v2 of NELL-995 (NLv2). For the triplets in $\mathcal{T}_{\text{tr}}$ and $\mathcal{T}_{\text{inf}}$, we extract 2-hop enclosing subgraphs using $\mathcal{F}_{\text{tr}}$ and $\mathcal{F}_{\text{inf}}$, respectively. Further experimental details are provided in Appendix E.

6.1. Label Classification using RTMD

We demonstrate that RTMD is a valid metric for quantifying differences between subgraphs extracted from real-world KGs. Figure 3 presents t-SNE visualizations of subgraphs based on RTMD with 3 layers. Specifically, the distance between points in the plot is proportional to the RTMD between subgraphs corresponding to the points. The results show that positive and negative triplets are well clustered according to their labels in WNv3. While the clustering is less distinct in FBv1 and NLv2, triplets in these datasets still exhibit a tendency to form label-based clusters.

Furthermore, for each dataset, we evaluate the classification accuracy of subgraphs corresponding to positive and negative triplets using a support vector machine (SVM) classifier with an indefinite kernel $\exp(-0.1\times\mathrm{RTMD})$ (Luss & d'Aspremont, 2007), following Chuang & Jegelka (2022). We set the number of layers in RTMD as $L=2$ or $L=3$. Table 1 reports the label classification accuracies of the SVM classifiers. Since the datasets contain equal numbers of positive and negative triplets, the baseline accuracy of random classification is 0.5. The classification results indicate that an SVM classifier can effectively distinguish subgraphs with different labels using RTMD, which demonstrates that



(a) WNv3 (b) FBv1 (c) NLv2

Figure 3. t-SNE visualizations of subgraphs using RTMD. Blue and orange points represent positive and negative subgraphs, respectively. The points tend to be clustered according to their labels.

Table 1. Label classification accuracies of support vector machine classifiers on WNv3, FBv1, and NLv2 using RTMD with 2 and 3 layers. Random classification accuracy is 0.5.

	WNv3	FBv1	NLv2
$L = 2$	0.8204	0.7743	0.8652
$L = 3$	0.8205	0.7739	0.8654

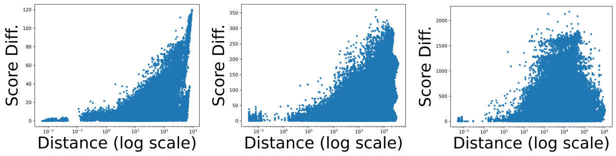
RTMD is an appropriate metric for quantifying the distance between subgraphs.

6.2. Comparing RTMD with Scores

In Section 4.3, we defined the Lipschitz continuity of SMPNNs using RTMD and proved that existing SMPNNs are Lipschitz continuous. To further support our theoretical findings, we empirically demonstrate that SMPNNs are indeed Lipschitz continuous with respect to RTMD. Specifically, we use GraIL (Teru et al., 2020) as a representative of SMPNN model, denoted as $f_{\mathbf{w}}$, and train $f_{\mathbf{w}}$ using $\mathcal{T}_{\text{tr}}$ of each dataset. For each pair of subgraphs $S_1, S_2 \in \{\{g(G_{\text{tr}}, e)|e \in \mathcal{T}_{\text{tr}}\}\} \cup \{\{g(G_{\text{inf}}, e)|e \in \mathcal{T}_{\text{inf}}\}\}$, we compute $\text{RTMD}(S_1, S_2)$ and the score difference $|f_{\mathbf{w}}(S_1) - f_{\mathbf{w}}(S_2)|$. In this computation, the number of layers in both the RTMD and the subgraph reasoning model is set to 3. Figure 4 displays the scatter plots of $\text{RTMD}(S_1, S_2)$ against $|f_{\mathbf{w}}(S_1) - f_{\mathbf{w}}(S_2)|$ in the WNv3, FBv1, and NLv2 datasets. Across all datasets, we observe that the maximum score difference increases as RTMD increases, demonstrating the Lipschitz continuity of $f_{\mathbf{w}}$. Moreover, the results indicate that the bound defined by RTMD approximates this trend, showing that RTMD is a valid metric for defining the Lipschitz continuity of SMPNNs.

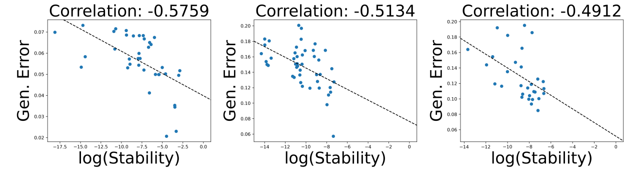
6.3. Comparing Stability with Generalization Errors

As described in Section 5.3, we have theoretically shown that a more stable model tends to exhibit better generalization capability. To empirically validate this claim, we train 48 different subgraph reasoning models, including GraIL, NBFNet, RED-GNN, and their variations, generated by permuting the candidate functions of each component of the SMPNN as detailed in Appendix E. For each trained model, we compute its empirical Lipschitz constant (Chuang & Jegelka, 2022), defined as $\max_{S_1,S_2} \frac{|f_{\mathbf{w}}(S_1)-f_{\mathbf{w}}(S_2)|}{\text{RTMD}(S_1,S_2)}$, where $S_1, S_2 \in \{\{g(G_{\text{tr}}, e)|e \in \mathcal{T}_{\text{tr}}\}\} \cup \{\{g(G_{\text{inf}}, e)|e \in \mathcal{T}_{\text{inf}}\}\}$. We then utilize the reciprocal of the empirical Lipschitz constant of a model as its stability. To assess the generalization capability of SMPNNs, we compute the generalization error defined in Section 5. Figure 5 presents the generalization errors of different subgraph reasoning models and their stability, along with the average Pearson correlation coefficients between generalization errors and stability. Across all datasets, we observe negative average Pearson correlation coefficients, which indicates that generalization error decreases as the stability increases, i.e., a positive correlation between generalization capability and stability. Also, Table 2 provides correlation values, p-values, and 95% confidence intervals regarding the correlation between stability and generalization errors on WNv3, FBv1, and NLv2. Since the p-values are all below 0.01 and the 95% confidence intervals do not include zero, we can conclude that the observed correlations are statistically significant. These results confirm that our theoretical finding, i.e., stable subgraph reasoning models exhibit high generalization capability, also holds in real-world datasets.



(a) WNv3 (b) FBv1 (c) NLv2

Figure 4. Scatter plots of RTMD versus the score differences on WNv3, FBv1, and NLv2. The maximum score difference increases as RTMD between subgraphs increases.



(a) WNv3 (b) FBv1 (c) NLv2

Figure 5. Comparisons of stability and generalization error on WNv3, FBv1, and NLv2. The dashed lines illustrate the negative correlations between stability and generalization error.

7. Conclusion and Future Works

We establish the relationship between stability and the generalization capability of subgraph reasoning models for inductive KGC. To facilitate a comprehensive analysis, we

Table 2. The correlation values, p-values, and 95% confidence intervals regarding the correlation between stability and generalization errors on WNv3, FBv1, and NLv2. If the p-value is lower than 0.01 and the 95% confidence interval does not include zero, we conclude that the observed correlation is statistically significant.

Dataset	Corr.	p-value	95% Confidence Interval
WNv3	-0.5759	0.00019	(-0.7584, -0.3097)
FBv1	-0.5134	0.00031	(-0.7013, -0.2589)
NLv2	-0.4912	0.00584	(-0.7235, -0.1591)

provide a framework that can represent various existing subgraph reasoning models. In this framework, we define the stability of subgraph reasoning models with respect to RTMD. Using the PAC-Bayesian approach, we derive the first generalization bound for subgraph reasoning models in the inductive setting and show that a more stable subgraph reasoning model exhibits a better generalization capability. On real-world KGs, we validate our theoretical findings and ensure they are aligned with empirical observations. Our analysis highlights the importance of designing stable subgraph reasoning models to enhance generalization capability in inductive KGC.

We will further explore additional theoretical properties of subgraph reasoning models, such as their expressive power and how they relate to stability and generalization capability. Also, we plan to extend our analyses to scenarios where both unobserved entities and unobserved relations appear during inference (Lee et al., 2023b; Geng et al., 2023).

Impact Statement

This paper provides a theoretical foundation for understanding stability and generalization capability in subgraph reasoning models. Our primary contributions are theoretical, aiming to advance the fundamental understanding of inductive knowledge graph completion. By establishing theoretical insights into how stability influences generalization, our work provides guidance for designing more robust and reliable subgraph reasoning models for knowledge graph completion. Knowledge graph completion models are widely used in various applications such as information retrieval, recommendation systems, and biomedical research, where ensuring model stability and generalization capability is crucial for reliable decision-making. Our theoretical findings contribute to developing methodologies that improve model robustness, potentially leading to more reliable predictions in real-world applications. We encourage future research to further explore the implications of stability and generalization in safety-critical domains.

Acknowledgements

This work was supported by the NRF of Korea grant funded by MSIT (65% from RS-2025-00559066, 5% from RS-2022-NR068758) and the IITP grant funded by MSIT (30% from RS-2022-II220369, (Part 4) Development of AI Technology to support Expert Decision-making that can Explain the Reasons/Grounds for Judgment Results based on Expert Knowledge).

References

Aminian, G., He, Y., Reinert, G., Szpruch, L., and Cohen, S. N. Generalization error of graph neural networks in the mean-field regime. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 1359–1391, 2024.

Barcelo, P., Galkin, M., Morris, C., and Orth, M. R. Weisfeiler and leman go relational. In *Proceedings of the 1st Learning on Graphs Conference*, 2022.

Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., and Yakhnenko, O. Translating embeddings for modeling multi-relational data. In *Proceedings of the 27th Conference on Neural Information Processing Systems*, pp. 2787–2795, 2013.

Bousquet, O. and Elisseeff, A. Stability and generalization. *Journal of Machine Learning Research*, 2:499–526, 2002.

Chen, J., He, H., Wu, F., and Wang, J. Topology-aware correlations between relations for inductive link prediction in knowledge graphs. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, pp. 6271–6278, 2021.

Chuang, C.-Y. and Jegelka, S. Tree mover's distance: Bridging graph metrics and stability of graph neural networks. In *Proceedings of the 36th Conference on Neural Information Processing Systems*, pp. 2944–2957, 2022.

Chung, C., Lee, J., and Whang, J. J. Representation learning on hyper-relational and numeric knowledge graphs with transformers. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 310–322, 2023.

Cuturi, M. Sinkhorn distances: Lightspeed computation of optimal transport. In *Proceedings of the 27th Conference on Neural Information Processing Systems*, pp. 2292–2300, 2013.

Dong, Z., Zhang, M., Payne, P., Province, M. A., Cruchaga, C., Zhao, T., Li, F., and Chen, Y. Rethinking the power of graph canonization in graph representation learning with stability. In *Proceedings of the 12th International Conference on Learning Representations*, 2024.

Feydy, J., Roussillon, P., Trouvé, A., and Gori, P. Fast and scalable optimal transport for brain tractograms. *Medical Image Computing and Computer Assisted Intervention –MICCAI 2019*, 2019.

Flamary, R., Courty, N., Gramfort, A., Alaya, M. Z., Boisbunon, A., Chambon, S., Chapel, L., Corenflos, A., Fatras, K., Fournier, N., Gautheron, L., Gayraud, N. T., Janati, H., Rakotomamonjy, A., Redko, I., Rolet, A., Schutz, A., Seguy, V., Sutherland, D. J., Tavenard, R., Tong, A., and Vayer, T. POT: Python optimal transport. *Journal of Machine Learning Research*, 22(78):1–8, 2021.

Franks, B. J., Morris, C., Velingker, A., and Geerts, F. Weisfeiler-Leman at the margin: When more expressivity matters. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 13885–13926, 2024.

Garg, V. K., Jegelka, S., and Jaakkola, T. Generalization and representational limits of graph neural networks. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 3419–3430, 2020.

Geng, Y., Chen, J., Pan, J. Z., Chen, M., Jiang, S., Zhang, W., and Chen, H. Relational message passing for fully inductive knowledge graph completion. In *Proceedings of the 2023 IEEE 39th International Conference on Data Engineering*, pp. 1221–1233, 2023.

Huang, X., Romero, M., Ceylan, I., and Barceló, P. A theory of link prediction via relational weisfeiler-leman on knowledge graphs. In *Proceedings of the 37th Conference on Neural Information Processing Systems*, pp. 19714–19748, 2023.

Huang, Y., Lu, W., Robinson, J., Yang, Y., Zhang, M., Jegelka, S., and Li, P. On the stability of expressive positional encodings for graphs. In *Proceedings of the 12th International Conference on Learning Representations*, 2024.

Ju, H., Li, D., Sharma, A., and Zhang, H. R. Generalization in graph neural networks: Improved PAC-Bayesian bounds on graph diffusion. In *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics*, pp. 6314–6341, 2023.

Karczewski, R., Souza, A. H., and Garg, V. On the generalization of equivariant graph neural networks. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 23159–23186, 2024.

Lee, J., Chung, C., Lee, H., Jo, S., and Whang, J. J. VISTA: Visual-textual knowledge graph representation learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 7314–7328, 2023a.

Lee, J., Chung, C., and Whang, J. J. InGram: Inductive knowledge graph embedding via relation graphs. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 18796–18809, 2023b.

Lee, J., Hwang, M., and Whang, J. J. PAC-Bayesian generalization bounds for knowledge graph representation learning. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 26589–26620, 2024.

Liao, R., Urtasun, R., and Zemel, R. A PAC-Bayesian approach to generalization bounds for graph neural networks. In *Proceedings of the 9th International Conference on Learning Representations*, 2021.

Lin, Q., Liu, J., Xu, F., Pan, Y., Zhu, Y., Zhang, L., and Zhao, T. Incorporating context graph with logical reasoning for inductive relation prediction. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 893–903, 2022.

Liu, T., Lv, Q., Wang, J., Yang, S., and Chen, H. Learning rule-induced subgraph representations for inductive relation prediction. In *Proceedings of the 37th Conference on Neural Information Processing Systems*, pp. 3517–3535, 2023.

Luss, R. and d'Aspremont, A. Support vector machine classification with indefinite kernels. In *Proceedings of the 21st Conference on Neural Information Processing Systems*, 2007.

Ma, J., Deng, J., and Mei, Q. Subgroup generalization and fairness of graph neural networks. In *Proceedings of the 35th Conference on Neural Information Processing Systems*, pp. 1048–1061, 2021.

Mai, S., Zheng, S., Yang, Y., and Hu, H. Communicative message passing for inductive relation reasoning. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, pp. 4294–4302, 2021.

Maskey, S., Levie, R., Lee, Y., and Kutyniok, G. Generalization analysis of message passing neural networks on large random graphs. In *Proceedings of the 36th Conference on Neural Information Processing Systems*, pp. 4805–4817, 2022.

McAllester, D. Simplified PAC-Bayesian margin bounds. In *Proceedings of the 16th Annual Conference on Computational Learning Theory and 7th Kernel Workshop*, pp. 203–215, 2003.

Morris, C., Geerts, F., Tönshoff, J., and Grohe, M. WL meet VC. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 25275–25302, 2023.

Neyshabur, B., Bhojanapalli, S., and Srebro, N. A PAC-Bayesian approach to spectrally-normalized margin bounds for neural networks. In *Proceedings of the 6th International Conference on Learning Representations*, 2018.

Oono, K. and Suzuki, T. Optimization and generalization analysis of transduction through gradient boosting and application to multi-scale graph neural networks. In *Proceedings of the 34th Conference on Neural Information Processing Systems*, pp. 18917–18930, 2020.

Qiu, H., Zhang, Y., Li, Y., and Yao, Q. Understanding expressivity of GNN in rule learning. In *Proceedings of the 12th International Conference on Learning Representations*, 2024.

Scarselli, F., Tsoi, A. C., and Hagenbuchner, M. The Vapnik-Chervonenkis dimension of graph and recursive neural networks. *Neural Networks*, 108:248–259, 2018.

Schlichtkrull, M., Kipf, T. N., Bloem, P., van den Berg, R., Titov, I., and Welling, M. Modeling relational data with graph convolutional networks. In *Proceedings of the 15th Extended Semantic Web Conference*, pp. 593–607, 2018.

Shalev-Shwartz, S., Shamir, O., Srebro, N., and Sridharan, K. Learnability, stability and uniform convergence. *Journal of Machine Learning Research*, 11(90):2635–2670, 2010.

Shen, J., Qu, Y., Zhang, W., and Yu, Y. Wasserstein distance guided representation learning for domain adaptation. In *Proceedings of the 32th AAAI Conference on Artificial Intelligence*, pp. 4058–4065, 2018.

Socher, R., Chen, D., Manning, C. D., and Ng, A. Y. Reasoning with neural tensor networks for knowledge base completion. In *Proceedings of the 27th Conference on Neural Information Processing Systems*, pp. 926–934, 2013.

Sun, Z., Deng, Z.-H., Nie, J.-Y., and Tang, J. RotatE: Knowledge graph embedding by relational rotation in complex space. In *Proceedings of the 7th International Conference on Learning Representations*, 2019.

Teru, K. K., Denis, E. G., and Hamilton, W. L. Inductive relation prediction by subgraph reasoning. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 9448–9457, 2020.

Villani, C. et al. *Optimal transport: old and new*, volume 338. 2009.

Wang, H., Yin, H., Zhang, M., and Li, P. Equivariant and stable positional encoding for more powerful graph neural networks. In *Proceedings of the 10th International Conference on Learning Representations*, 2022.

Wang, Q., Mao, Z., Wang, B., and Guo, L. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering*, 29(12):2724–2743, 2017.

Weisfeiler, B. and Lehman, A. A. A reduction of a graph to a canonical form and an algebra arising during this reduction. *Nauchno-Technicheskaya Informatsia*, 2(9): 12–16, 1968.

Xu, K., Hu, W., Leskovec, J., and Jegelka, S. How powerful are graph neural networks? In *Proceedings of the 7th International Conference on Learning Representations*, 2019.

Yang, B., Yih, W.-t., He, X., Gao, J., and Deng, L. Embedding entities and relations for learning and inference in knowledge bases. In *Proceedings of the 3rd International Conference on Learning Representations*, 2015.

Zhang, M. and Chen, Y. Link prediction based on graph neural networks. In *Proceedings of the 32nd Conference on Neural Information Processing Systems*, 2018.

Zhang, Y. and Yao, Q. Knowledge graph reasoning with relational digraph. In *Proceedings of the ACM Web Conference 2022*, pp. 912–924, 2022.

Zhang, Y., Zhou, Z., Yao, Q., Chu, X., and Han, B. AdaProp: Learning adaptive propagation for graph neural network based knowledge graph reasoning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 3446–3457, 2023.

Zhou, Y., Kutyniok, G., and Ribeiro, B. OOD link prediction generalization capabilities of message-passing GNNs in larger test graphs. In *Proceedings of the 36th Conference on Neural Information Processing Systems*, pp. 20257–20272, 2022.

Zhu, Z., Zhang, Z., Xhonneux, L. A. C., and Tang, J. Neural Bellman-Ford networks: A general graph neural network framework for link prediction. In *Proceedings of the 35th Conference on Neural Information Processing Systems*, pp. 29476–29490, 2021.

Zhu, Z., Yuan, X., Galkin, M., Xhonneux, L.-P., Zhang, M., Gazeau, M., and Tang, J. A*Net: A scalable path-based reasoning approach for knowledge graphs. In *Proceedings of the 37th Conference on Neural Information Processing Systems*, pp. 59323–59336, 2023.

A. Instantiatation of Existing Subgraph Reasoning Models

A subgraph reasoning model consists of a subgraph extractor and an SMPNN. By appropriately configuring the subgraph extractor and the functions within the SMPNN, we can represent existing well-known models for inductive KGC by subgraph reasoning. Furthermore, under Assumption A.1, each function satisfies Lipschitz continuous.

Assumption A.1. We assume that the L_2-norms of all weight vectors, weight matrices, and relation embedding matrices are upper bounded by κ. Additionally, we assume that the L_2-norms of the entity representation vectors computed at every layer of SMPNNs are upper bounded by β.

The Lipschitz continuity for each function in SMPNNs is defined as follows:

Definition A.2 (Lipschitz Continuity)**.** A *message* function MSG of SMPNNs is Lipschitz continuous if and only if there exist constants $A_{\text{msg}}^{(l)}, B_{\text{msg}}^{(l)}, C_{\text{msg}}^{(l)}, D_{\text{msg}}^{(l)} \geq 0$ that satisfies

$$\begin{aligned}&\|\text{MSG}^{(l)}(\mathbf{a}_1, \mathbf{a}_2, r_1, q_1) - \text{MSG}^{(l)}(\mathbf{b}_1, \mathbf{b}_2, r_2, q_2)\|_2 \\ \leq &A_{\text{msg}}^{(l)}\|\mathbf{a}_1 - \mathbf{b}_1\|_2 + B_{\text{msg}}^{(l)}\|\mathbf{a}_2 - \mathbf{b}_2\|_2 + C_{\text{msg}}^{(l)}\mathbb{1}[r_1 \neq r_2] + D_{\text{msg}}^{(l)}\mathbb{1}[q_1 \neq q_2]\end{aligned}$$

Given a transportation plan $\boldsymbol{P}_{\mathcal{A},\mathcal{B}}$ between $\mathcal{A}$ and $\mathcal{B}$, an *aggregation* function AGG of SMPNNs is Lipschitz continuous if and only if there exists a constant $A_{\text{agg}}^{(l)} \geq 0$ that satisfies

$$\|\text{AGG}^{(l)}(\mathcal{A} = \{\{\mathbf{a}_i\}\}_{i=1}^n) - \text{AGG}^{(l)}(\mathcal{B} = \{\{\mathbf{b}_j\}\}_{j=1}^n)\|_2 \leq A_{\text{agg}}^{(l)} \sum_{i,j} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}^{(l)}[i,j]\|\mathbf{a}_i - \mathbf{b}_i\|_2$$

An *update* function UPD of SMPNNs is Lipschitz continuous if and only if there exist constants $A_{\text{upd}}^{(l)}, B_{\text{upd}}^{(l)} \geq 0$ that satisfies

$$\|\text{UPD}^{(l)}(\mathbf{a}_1, \mathbf{a}_2) - \text{UPD}^{(l)}(\mathbf{b}_1, \mathbf{b}_2)\|_2 \leq A_{\text{upd}}^{(l)}\|\mathbf{a}_1 - \mathbf{b}_1\|_2 + B_{\text{upd}}^{(l)}\|\mathbf{a}_2 - \mathbf{b}_2\|_2$$

Given a transportation plan $\boldsymbol{P}_{\mathcal{A},\mathcal{B}}$ between $\mathcal{A}$ and $\mathcal{B}$, a *global-readout* function GRD of SMPNNs is Lipschitz continuous if and only if there exists a constant $A_{\text{grd}} \geq 0$ that satisfies

$$\|\text{GRD}(\mathcal{A} = \{\{\mathbf{a}_i\}\}_{i=1}^n) - \text{GRD}(\mathcal{B} = \{\{\mathbf{b}_j\}\}_{j=1}^n)\|_2 \leq A_{\text{grd}} \sum_{i,j} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j]\|\mathbf{a}_i - \mathbf{b}_i\|_2$$

A *readout* function RD of SMPNNs is Lipschitz continuous if and only if there exist constants $A_{\text{rd}}, B_{\text{rd}}, C_{\text{rd}}, D_{\text{rd}} \geq 0$ that satisfies

$$\begin{aligned}&|\text{RD}(\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3, q_1) - \text{RD}(\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3, q_2)| \\ \leq &A_{\text{rd}}\|\mathbf{a}_1 - \mathbf{b}_1\|_2 + B_{\text{rd}}\|\mathbf{a}_2 - \mathbf{b}_2\|_2 + C_{\text{rd}}\|\mathbf{a}_3 - \mathbf{b}_3\|_2 + D_{\text{rd}}\mathbb{1}[q_1 \neq q_2]\end{aligned}$$

where $n > 0$, $\mathbf{a}_i, \mathbf{b}_j \in \mathbb{R}^d$ for $1 \leq i, j \leq n$, and $r_1, r_2, q_1, q_2 \in \mathcal{R}$.

Now, we demonstrate how GraIL (Teru et al., 2020), NBFNet (Zhu et al., 2021), and RED-GNN (Zhang & Yao, 2022) can be instantiated as subgraph reasoning models.

A.1. Instantiating GraIL

GraIL (Teru et al., 2020) can be instantiated as follows:

A.1.1. Subgraph Extraction

GraIL extracts the enclosing subgraph for the head and tail entities of a given triplet, which is constructed by the intersection of k-hop neighbors entities from the head and tail entity. Therefore, the subgraph extractor is a function that extracts the enclosing subgraph for h and t in the triplet (h, q, t).

A.1.2. Initialization

GraIL applies a double radius vertex labeling scheme to the entities within the subgraph. This scheme encodes the shortest path distances from each entity to the head and tail entities as one-hot vectors and concatenates these vectors. Therefore, by setting INIT_S to the double radius vertex labeling scheme, the initial labeling of GraIL can be instantiated.

A.1.3. Subgraph Message-Passing Neural Networks

Message Function The *message* function of GraIL is formulated as follows.

$$\text{MSG}^{(l)}(\mathbf{x}_S^{(l-1)}(u), \mathbf{x}_S^{(l-1)}(v), r, q) = \alpha_{r,q,v,u}^{(l)} \boldsymbol{W}_r^{(l)} \mathbf{x}_S^{(l-1)}(u)$$

$$\alpha_{r,q,v,u}^{(l)} = \sigma(\mathbf{a}_5^{(l)} \cdot \text{ReLU}(\boldsymbol{W}_1^{(l)} \mathbf{x}_S^{(l-1)}(u) + \boldsymbol{W}_2^{(l)} \mathbf{x}_S^{(l-1)}(v) + \boldsymbol{W}_3^{(l)} \boldsymbol{R}_a[r] + \boldsymbol{W}_4^{(l)} \boldsymbol{R}_a[q] + \mathbf{b}_1) + b_2)$$

where $\boldsymbol{W}_r^{(l)}, \boldsymbol{W}_1^{(l)}, \boldsymbol{W}_2^{(l)}, \boldsymbol{W}_3^{(l)}, \boldsymbol{W}_4^{(l)}, \boldsymbol{R}_a$ are learnable weight matrices that satisfy the Assumption A.1, $\mathbf{a}_5^{(l)}, \mathbf{b}_1$ are learnable weight vectors that satisfy the Assumption A.1, and b_2 is a learnable parameter, σ is a sigmoid function, and ReLU is a ReLU activation function. The Lipschitz continuity of the *message* function of GraIL can be formally established as follows.

$$\begin{aligned}
&\|\text{MSG}^{(l)}(\mathbf{x}_{S_1}^{(l-1)}(u_1), \mathbf{x}_{S_1}^{(l-1)}(v_1), r_1, q_1) - \text{MSG}^{(l)}(\mathbf{x}_{S_2}^{(l-1)}(u_2), \mathbf{x}_{S_2}^{(l-1)}(v_2), r_2, q_2)\|_2 \\
\leq&\|\alpha_{r_1,q_1,v_1,u_1}^{(l)} \boldsymbol{W}_{r_1}^{(l)} \mathbf{x}_{S_1}^{(l-1)}(u_1) - \alpha_{r_2,q_2,v_2,u_2}^{(l)} \boldsymbol{W}_{r_2}^{(l)} \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 \\
\leq&\|\alpha_{r_1,q_1,v_1,u_1}^{(l)} \left(\boldsymbol{W}_{r_1}^{(l)} \mathbf{x}_{S_1}^{(l-1)}(u_1) - \boldsymbol{W}_{r_2}^{(l)} \mathbf{x}_{S_2}^{(l-1)}(u_2)\right) + \left(\alpha_{r_1,q_1,v_1,u_1}^{(l)} - \alpha_{r_2,q_2,v_2,u_2}^{(l)}\right) \boldsymbol{W}_{r_2}^{(l)} \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 \\
\leq&|\alpha_{r_1,q_1,v_1,u_1}^{(l)}| \|\boldsymbol{W}_{r_1}^{(l)} \mathbf{x}_{S_1}^{(l-1)}(u_1) - \boldsymbol{W}_{r_2}^{(l)} \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + |\alpha_{r_1,q_1,v_1,u_1}^{(l)} - \alpha_{r_2,q_2,v_2,u_2}^{(l)}| \|\boldsymbol{W}_{r_2}^{(l)} \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 \\
\leq&|\alpha_{r_1,q_1,v_1,u_1}^{(l)}| \|\boldsymbol{W}_{r_1}^{(l)} \left(\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\right) + \left(\boldsymbol{W}_{r_1}^{(l)} - \boldsymbol{W}_{r_2}^{(l)}\right) \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + |\alpha_{r_1,q_1,v_1,u_1}^{(l)} - \alpha_{r_2,q_2,v_2,u_2}^{(l)}| \|\boldsymbol{W}_{r_2}^{(l)} \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 \\
\leq&|\alpha_{r_1,q_1,v_1,u_1}^{(l)}| \left(\|\boldsymbol{W}_{r_1}^{(l)}\|_2 \|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + \|\boldsymbol{W}_{r_1}^{(l)} - \boldsymbol{W}_{r_2}^{(l)}\|_2 \|\mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2\right) + |\alpha_{r_1,q_1,v_1,u_1}^{(l)} - \alpha_{r_2,q_2,v_2,u_2}^{(l)}| \|\boldsymbol{W}_{r_2}^{(l)}\|_2 \|\mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 \\
\leq&\kappa \|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + 2\kappa\beta \mathbb{1}[r_1 \neq r_2] + \kappa\beta |\alpha_{r_1,q_1,v_1,u_1}^{(l)} - \alpha_{r_2,q_2,v_2,u_2}^{(l)}|
\end{aligned}$$

For the attention value $\alpha_{r_1,q_1,v_1,u_1}^{(l)}$, the following inequalities hold.

$$\begin{aligned}
&|\alpha_{r_1,q_1,v_1,u_1}^{(l)} - \alpha_{r_2,q_2,v_2,u_2}^{(l)}| \\
\leq&|\sigma(\mathbf{a}_5^{(l)} \cdot \text{ReLU}(\boldsymbol{W}_1^{(l)} \mathbf{x}_{S_1}^{(l-1)}(u_1) + \boldsymbol{W}_2^{(l)} \mathbf{x}_{S_1}^{(l-1)}(v_1) + \boldsymbol{W}_3^{(l)} \boldsymbol{R}_a[r_1] + \boldsymbol{W}_4^{(l)} \boldsymbol{R}_a[q_1] + \mathbf{b}_1) + b_2) - \\
&\sigma(\mathbf{a}_5^{(l)} \cdot \text{ReLU}(\boldsymbol{W}_1^{(l)} \mathbf{x}_{S_2}^{(l-1)}(u_2) + \boldsymbol{W}_2^{(l)} \mathbf{x}_{S_2}^{(l-1)}(v_2) + \boldsymbol{W}_3^{(l)} \boldsymbol{R}_a[r_2] + \boldsymbol{W}_4^{(l)} \boldsymbol{R}_a[q_2] + \mathbf{b}_1) + b_2)| \\
\leq&|\mathbf{a}_5^{(l)} \cdot \text{ReLU}(\boldsymbol{W}_1^{(l)} \mathbf{x}_{S_1}^{(l-1)}(u_1) + \boldsymbol{W}_2^{(l)} \mathbf{x}_{S_1}^{(l-1)}(v_1) + \boldsymbol{W}_3^{(l)} \boldsymbol{R}_a[r_1] + \boldsymbol{W}_4^{(l)} \boldsymbol{R}_a[q_1] + \mathbf{b}_1) - \\
&\mathbf{a}_5^{(l)} \cdot \text{ReLU}(\boldsymbol{W}_1^{(l)} \mathbf{x}_{S_2}^{(l-1)}(u_2) + \boldsymbol{W}_2^{(l)} \mathbf{x}_{S_2}^{(l-1)}(v_2) + \boldsymbol{W}_3^{(l)} \boldsymbol{R}_a[r_2] + \boldsymbol{W}_4^{(l)} \boldsymbol{R}_a[q_2] + \mathbf{b}_1)| \\
\leq&\|\mathbf{a}_5^{(l)}\|_2 \|\text{ReLU}(\boldsymbol{W}_1^{(l)} \mathbf{x}_{S_1}^{(l-1)}(u_1) + \boldsymbol{W}_2^{(l)} \mathbf{x}_{S_1}^{(l-1)}(v_1) + \boldsymbol{W}_3^{(l)} \boldsymbol{R}_a[r_1] + \boldsymbol{W}_4^{(l)} \boldsymbol{R}_a[q_1] + \mathbf{b}_1) - \\
&\text{ReLU}(\boldsymbol{W}_1^{(l)} \mathbf{x}_{S_2}^{(l-1)}(u_2) + \boldsymbol{W}_2^{(l)} \mathbf{x}_{S_2}^{(l-1)}(v_2) + \boldsymbol{W}_3^{(l)} \boldsymbol{R}_a[r_2] + \boldsymbol{W}_4^{(l)} \boldsymbol{R}_a[q_2] + \mathbf{b}_1)\|_2 \\
\leq&\|\mathbf{a}_5^{(l)}\|_2 \|\boldsymbol{W}_1^{(l)} \mathbf{x}_{S_1}^{(l-1)}(u_1) + \boldsymbol{W}_2^{(l)} \mathbf{x}_{S_1}^{(l-1)}(v_1) + \boldsymbol{W}_3^{(l)} \boldsymbol{R}_a[r_1] + \boldsymbol{W}_4^{(l)} \boldsymbol{R}_a[q_1] - \\
&\left(\boldsymbol{W}_1^{(l)} \mathbf{x}_{S_2}^{(l-1)}(u_2) + \boldsymbol{W}_2^{(l)} \mathbf{x}_{S_2}^{(l-1)}(v_2) + \boldsymbol{W}_3^{(l)} \boldsymbol{R}_a[r_2] + \boldsymbol{W}_4^{(l)} \boldsymbol{R}_a[q_2]\right) \|_2 \\
\leq&\|\mathbf{a}_5^{(l)}\|_2 \left(\|\boldsymbol{W}_1^{(l)}\|_2 \|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + \|\boldsymbol{W}_2^{(l)}\|_2 \|\mathbf{x}_{S_1}^{(l-1)}(v_1) - \mathbf{x}_{S_2}^{(l-1)}(v_2)\|_2 + \right. \\
&\left. \|\boldsymbol{W}_3^{(l)}\|_2 \|\boldsymbol{R}_a[r_1] - \boldsymbol{R}_a[r_2]\|_2 + \|\boldsymbol{W}_4^{(l)}\|_2 \|\boldsymbol{R}_a[q_1] - \boldsymbol{R}_a[q_2]\|_2\right) \\
\leq&\|\mathbf{a}_5^{(l)}\|_2 \left(\|\boldsymbol{W}_1^{(l)}\|_2 \|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + \|\boldsymbol{W}_2^{(l)}\|_2 \|\mathbf{x}_{S_1}^{(l-1)}(v_1) - \mathbf{x}_{S_2}^{(l-1)}(v_2)\|_2 + \right. \\
&\left. \|\boldsymbol{W}_3^{(l)}\|_2 \|\boldsymbol{R}_a \cdot \text{One-Hot}(r_1) - \boldsymbol{R}_a \cdot \text{One-Hot}(r_2)\|_2 + \|\boldsymbol{W}_4^{(l)}\|_2 \|\boldsymbol{R}_a \cdot \text{One-Hot}(q_1) - \boldsymbol{R}_a \cdot \text{One-Hot}(q_2)\|_2\right) \\
\leq&\|\mathbf{a}_5^{(l)}\|_2 \left(\|\boldsymbol{W}_1^{(l)}\|_2 \|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + \|\boldsymbol{W}_2^{(l)}\|_2 \|\mathbf{x}_{S_1}^{(l-1)}(v_1) - \mathbf{x}_{S_2}^{(l-1)}(v_2)\|_2 + \right. \\
&\left. \|\boldsymbol{W}_3^{(l)}\|_2 \|\boldsymbol{R}_a\|_2 \|\text{One-Hot}(r_1) - \text{One-Hot}(r_2)\|_2 + \|\boldsymbol{W}_4^{(l)}\|_2 \|\boldsymbol{R}_a\|_2 \|\text{One-Hot}(q_1) - \text{One-Hot}(q_2)\|_2\right) \\
\leq&\|\mathbf{a}_5^{(l)}\|_2 \left(\|\boldsymbol{W}_1^{(l)}\|_2 \|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + \|\boldsymbol{W}_2^{(l)}\|_2 \|\mathbf{x}_{S_1}^{(l-1)}(v_1) - \mathbf{x}_{S_2}^{(l-1)}(v_2)\|_2 + \right. \\
&\left. + \sqrt{2} \|\boldsymbol{W}_3^{(l)}\|_2 \|\boldsymbol{R}_a\|_2 \mathbb{1}[r_1 \neq r_2] + \sqrt{2} \|\boldsymbol{W}_4^{(l)}\|_2 \|\boldsymbol{R}_a\|_2 \mathbb{1}[q_1 \neq q_2]\right) \\
\leq&\kappa \left(\kappa \|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + \kappa \|\mathbf{x}_{S_1}^{(l-1)}(v_1) - \mathbf{x}_{S_2}^{(l-1)}(v_2)\|_2 \sqrt{2}\kappa^2 \mathbb{1}[r_1 \neq r_2] + \sqrt{2}\kappa^2 \mathbb{1}[q_1 \neq q_2]\right) \\
=&\kappa^2 \|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + \kappa^2 \|\mathbf{x}_{S_1}^{(l-1)}(v_1) - \mathbf{x}_{S_2}^{(l-1)}(v_2)\|_2 + \sqrt{2}\kappa^3 \mathbb{1}[r_1 \neq r_2] + \sqrt{2}\kappa^3 \mathbb{1}[q_1 \neq q_2]
\end{aligned}$$

Therefore,

$$\begin{aligned}
&\|\text{MSG}^{(l)}(\mathbf{x}_{S_1}^{(l-1)}(u_1), \mathbf{x}_{S_1}^{(l-1)}(v_1), r_1, q_1) - \text{MSG}^{(l)}(\mathbf{x}_{S_2}^{(l-1)}(u_2), \mathbf{x}_{S_2}^{(l-1)}(v_2), r_2, q_2)\|_2 \\
\leq& \kappa \|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + 2\kappa\beta \mathbb{1}[r_1 \neq r_2] + \\
&\kappa\beta \left(\kappa^2 \|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + \kappa^2 \|\mathbf{x}_{S_1}^{(l-1)}(v_1) - \mathbf{x}_{S_2}^{(l-1)}(v_2)\|_2 + \sqrt{2}\kappa^3 \mathbb{1}[r_1 \neq r_2] + \sqrt{2}\kappa^3 \mathbb{1}[q_1 \neq q_2]\right) \\
\leq& (\kappa + \beta\kappa^3) \|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + \beta\kappa^3 \|\mathbf{x}_{S_1}^{(l-1)}(v_1) - \mathbf{x}_{S_2}^{(l-1)}(v_2)\|_2 + \\
&(2\kappa\beta + \sqrt{2}\beta\kappa^4) \mathbb{1}[r_1 \neq r_2] + \sqrt{2}\beta\kappa^4 \mathbb{1}[q_1 \neq q_2]
\end{aligned}$$

Finally, the Lipschitz constants of the message function of GraIL are computed as follows.

$$A_{\text{msg}}^{(l)} = \kappa + \beta\kappa^3, B_{\text{msg}}^{(l)} = \beta\kappa^3, C_{\text{msg}}^{(l)} = 2\kappa\beta + \sqrt{2}\beta\kappa^4, D_{\text{msg}}^{(l)} = \sqrt{2}\beta\kappa^4$$

History Function The *history* function of GraIL is $\theta(k) = k - 1$.

Aggregation Function GraIL uses sum aggregation as the *aggregation* function, which is formulated as follows.

$$\text{AGG}^{(l)}(\mathcal{A} = \{\{\mathbf{a}_i\}\}_{i=1}^n) = \sum_{i=1}^n \mathbf{a}_i$$

We can justify the Lipschitz continuity of the *aggregation* function of GraIL as follows.

$$\begin{aligned}
\|\text{AGG}^{(l)}(\mathcal{A} = \{\{\mathbf{a}_i\}\}_{i=1}^n) - \text{AGG}^{(l)}(\mathcal{B} = \{\{\mathbf{b}_j\}\}_{j=1}^n)\|_2 =& \|\sum_{i=1}^n \mathbf{a}_i - \sum_{j=1}^n \mathbf{b}_j\|_2 \\
=& \|\sum_{i,j} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j](\mathbf{a}_i - \mathbf{b}_j)\|_2 \\
\leq& \sum_{i,j} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j] \|\mathbf{a}_i - \mathbf{b}_j\|_2
\end{aligned}$$

where $\boldsymbol{P}_{\mathcal{A},\mathcal{B}}$ is a transportation plan between $\mathcal{A}, \mathcal{B}$. Finally, the Lipschitz constants of the aggregation function of GraIL are computed as follows.

$$A_{\text{agg}}^{(l)} = 1$$

Also, since adding zero vectors to the sum of vectors does not change the resulting vector, the embedding vector Φ_{agg} is a zero vector.

Update Function The *update* function of GraIL is formulated as follows

$$\text{UPD}^{(l)}(\mathbf{a}_1, \mathbf{a}_2) = \text{ReLU}(\boldsymbol{W}_{\text{self}}^{(l)} \mathbf{a}_1 + \mathbf{a}_2)$$

where $\boldsymbol{W}_{\text{self}}^{(l)}$ is a learnable weight matrix that satisfies Assumption A.1. We can justify the Lipschitz continuity of the *update* function of GraIL as follows.

$$\begin{aligned}
\|\text{UPD}^{(l)}(\mathbf{a}_1, \mathbf{a}_2) - \text{UPD}^{(l)}(\mathbf{b}_1, \mathbf{b}_2)\|_2 =& \|\text{ReLU}(\boldsymbol{W}_{\text{self}}^{(l)} \mathbf{a}_1 + \mathbf{a}_2) - \text{ReLU}(\boldsymbol{W}_{\text{self}}^{(l)} \mathbf{b}_1 + \mathbf{b}_2)\|_2 \\
\leq& \|\boldsymbol{W}_{\text{self}}^{(l)} \mathbf{a}_1 + \mathbf{a}_2 - \boldsymbol{W}_{\text{self}}^{(l)} \mathbf{b}_1 - \mathbf{b}_2\|_2 \\
\leq& \|\boldsymbol{W}_{\text{self}}^{(l)} (\mathbf{a}_1 - \mathbf{b}_1) + \mathbf{a}_2 - \mathbf{b}_2\|_2 \\
\leq& \|\boldsymbol{W}_{\text{self}}^{(l)}\|_2 \|\mathbf{a}_1 - \mathbf{b}_1\|_2 + \|\mathbf{a}_2 - \mathbf{b}_2\|_2 \\
\leq& \kappa \|\mathbf{a}_1 - \mathbf{b}_1\|_2 + \|\mathbf{a}_2 - \mathbf{b}_2\|_2
\end{aligned}$$

Finally, the Lipschitz constants of the update function of GraIL are computed as follows.

$$A_{\text{upd}}^{(l)} = \kappa, B_{\text{upd}}^{(l)} = 1$$

Global Readout Function GraIL uses mean aggregation as the *global-readout* function, which is formulated as follows.

$$\mathrm{GRD}(\mathcal{A} = \{\{\mathbf{a}_i\}\}_{i=1}^{n}) = \frac{1}{n}\sum_{i=1}^{n}\mathbf{a}_i$$

We can justify the Lipschitz continuity of the *global-readout* function of GraIL as follows.

$$\begin{aligned}\|\mathrm{GRD}(\mathcal{A} = \{\{\mathbf{a}_i\}\}_{i=1}^{n}) - \mathrm{GRD}(\mathcal{B} = \{\{\mathbf{b}_j\}\}_{j=1}^{n})\|_2 =& \|\frac{1}{n}\sum_{i=1}^{n}\mathbf{a}_i - \frac{1}{n}\sum_{j=1}^{n}\mathbf{b}_j\|_2 \\ =& \frac{1}{n}\|\sum_{i,j}\boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j](\mathbf{a}_i - \mathbf{b}_j)\|_2 \\ \leq& \frac{1}{n}\sum_{i,j}\boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j]\|\mathbf{a}_i - \mathbf{b}_j\|_2\end{aligned}$$

where $\boldsymbol{P}_{\mathcal{A},\mathcal{B}}$ is a transportation plan between $\mathcal{A},\mathcal{B}$ and $n = \max(|\mathcal{V}_{S_1}|, |\mathcal{V}_{S_2}|)$. Finally, the Lipschitz constants of the global readout function of GraIL are computed as follows.

$$A_{\text{grd}} = \frac{1}{\max(|\mathcal{V}_{S_1}|, |\mathcal{V}_{S_2}|)}$$

Also, since $\mathrm{GRD}(\{\{\mathbf{a}_i\}\}_{i=1}^{n}) = \mathrm{GRD}(\{\{\mathbf{a}_i\}\}_{i=1}^{n} \cup \{\{\frac{1}{n}\sum_{i=1}^{n}\mathbf{a}_i\}\})$, the embedding vector $\Phi_{\text{grd}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{a}_i$.

Readout Function The *readout* function of GraIL is formulated as follows.

$$\mathrm{RD}(\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3, q) = \boldsymbol{W}_{\text{h}}\mathbf{a}_1 + \boldsymbol{W}_{\text{t}}\mathbf{a}_2 + \boldsymbol{W}_{\text{g}}\mathbf{a}_3 + \boldsymbol{W}_{\text{q}}\boldsymbol{R}[q]$$

where $\boldsymbol{W}_{\text{h}}, \boldsymbol{W}_{\text{t}}, \boldsymbol{W}_{\text{g}}, \boldsymbol{W}_{\text{q}}$ are learnable weight matrices that satisfy Assumption A.1. We can justify the Lipschitz continuity of the *readout* function of GraIL as follows.

$$\begin{aligned}&|\mathrm{RD}(\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3, q_1) - \mathrm{RD}(\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3, q_2)| \\ \leq& |\boldsymbol{W}_{\text{h}}\mathbf{a}_1 + \boldsymbol{W}_{\text{t}}\mathbf{a}_2 + \boldsymbol{W}_{\text{g}}\mathbf{a}_3 + \boldsymbol{W}_{\text{q}}\boldsymbol{R}[q_1] - \boldsymbol{W}_{\text{h}}\mathbf{b}_1 - \boldsymbol{W}_{\text{t}}\mathbf{b}_2 - \boldsymbol{W}_{\text{g}}\mathbf{b}_3 - \boldsymbol{W}_{\text{q}}\boldsymbol{R}[q_2]| \\ \leq& |\boldsymbol{W}_{\text{h}}(\mathbf{a}_1 - \mathbf{b}_1) + \boldsymbol{W}_{\text{t}}(\mathbf{a}_2 - \mathbf{b}_2) + \boldsymbol{W}_{\text{g}}(\mathbf{a}_3 - \mathbf{b}_3) + \boldsymbol{W}_{\text{q}}(\boldsymbol{R}[q_1] - \boldsymbol{R}[q_2])| \\ =& |\boldsymbol{W}_{\text{h}}(\mathbf{a}_1 - \mathbf{b}_1) + \boldsymbol{W}_{\text{t}}(\mathbf{a}_2 - \mathbf{b}_2) + \boldsymbol{W}_{\text{g}}(\mathbf{a}_3 - \mathbf{b}_3) + \boldsymbol{W}_{\text{q}}(\boldsymbol{R} \cdot \text{One-Hot}(q_1) - \boldsymbol{R} \cdot \text{One-Hot}(q_1))| \\ \leq& \|\boldsymbol{W}_{\text{h}}\|_2\|\mathbf{a}_1 - \mathbf{b}_1\|_2 + \|\boldsymbol{W}_{\text{t}}\|_2\|\mathbf{a}_2 - \mathbf{b}_2\|_2 + \|\boldsymbol{W}_{\text{g}}\|_2\|\mathbf{a}_3 - \mathbf{b}_3\|_2 + \|\boldsymbol{W}_{\text{q}}\|_2\|\boldsymbol{R}\|_2\|\text{One-Hot}(q_1) - \text{One-Hot}(q_1)\|_2 \\ \leq& \kappa\|\mathbf{a}_1 - \mathbf{b}_1\|_2 + \kappa\|\mathbf{a}_2 - \mathbf{b}_2\|_2 + \kappa\|\mathbf{a}_3 - \mathbf{b}_3\|_2 + \sqrt{2}\kappa^2\mathbb{1}[q_1 \neq q_2]\end{aligned}$$

Finally, the Lipschitz constants of the readout function of GraIL are computed as follows.

$$A_{\text{rd}} = B_{\text{rd}} = C_{\text{rd}} = \kappa, D_{\text{rd}} = \sqrt{2}\kappa^2$$

A.2. Instantiation of NBFNet

NBFNet (Zhu et al., 2021) can be instantiated as follows:

A.2.1. Subgraph Extraction

NBFNet computes a score for a given triplet based on the paths from the head entity to the tail entity using the entities' conditional representation with respect to the head entity and the query relation. Thus, it does not explicitly extract subgraphs for scoring. However, from the perspective of calculating the score for a specific triplet, NBFNet considers only the entities included in the union of L-hop neighbor entities from the head and tail entity. This approach is equivalent to scoring the subgraph constructed by the union of L-hop neighbor entities from the two entities.

A.2.2. Initialization

In NBFNet, only the initial embedding vector of the head entity h is initialized with the embedding vector of the query relation, while the initial representations of all other entities are initialized as zero vectors.

A.2.3. Subgraph Message-Passing Neural Networks

Message Function NBFNet utilizes TransE (Bordes et al., 2013), DistMult (Yang et al., 2015), and RotatE (Sun et al., 2019) as the *message* function. First, the *message* function with TransE is formulated as follows.

$$\mathrm{MSG}^{(l)}(\mathbf{x}_S^{(l-1)}(u), \mathbf{x}_S^{(l-1)}(v), r, q) = \mathbf{x}_S^{(l-1)}(u) + \boldsymbol{W}_r^{(l)}\boldsymbol{R}[q] + \mathbf{b}_r^{(l)}$$

where $\boldsymbol{W}_r^{(l)}$ is a learnable weight matrix that satisfies the Assumption A.1, and $\mathbf{b}_r^{(l)}$ is a learnable weight vector that satisfies the Assumption A.1. We can show the Lipschitz continuity of the *message* function with TransE as follows.

$$\begin{aligned}
&\|\mathrm{MSG}^{(l)}(\mathbf{x}_{S_1}^{(l-1)}(u_1), \mathbf{x}_{S_1}^{(l-1)}(v_1), r_1, q_1) - \mathrm{MSG}^{(l)}(\mathbf{x}_{S_2}^{(l-1)}(u_2), \mathbf{x}_{S_2}^{(l-1)}(v_2), r_2, q_2)\|_2 \\
\leq&\|\mathbf{x}_{S_1}^{(l-1)}(u_1) + \boldsymbol{W}_{r_1}^{(l)}\boldsymbol{R}[q_1] + \mathbf{b}_{r_1}^{(l)} - \mathbf{x}_{S_2}^{(l-1)}(u_2) - \boldsymbol{W}_{r_2}^{(l)}\boldsymbol{R}[q_2] - \mathbf{b}_{r_2}^{(l)}\|_2 \\
\leq&\|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + \|\boldsymbol{W}_{r_1}\|_2^{(l)}\|\boldsymbol{R}[q_1] - \boldsymbol{R}[q_2]\|_2 + \|\boldsymbol{W}_{r_1}^{(l)} - \boldsymbol{W}_{r_2}^{(l)}\|_2\boldsymbol{R}[q_2] + \|\mathbf{b}_{r_1}^{(l)} - \mathbf{b}_{r_2}^{(l)}\|_2 \\
\leq&\|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + \sqrt{2}\|\boldsymbol{W}_{r_1}^{(l)}\|_2\|\boldsymbol{R}\|_2\mathbb{1}[q_1 \neq q_2] + \|\boldsymbol{W}_{r_1}^{(l)} - \boldsymbol{W}_{r_2}^{(l)}\|_2\boldsymbol{R}[q_2] + \|\mathbf{b}_{r_1}^{(l)} - \mathbf{b}_{r_2}^{(l)}\|_2 \\
\leq&\|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + \sqrt{2}\kappa^2\mathbb{1}[q_1 \neq q_2] + 2\kappa^2\mathbb{1}[r_1 \neq r_2] + 2\kappa\mathbb{1}[r_1 \neq r_2]
\end{aligned}$$

Finally, the Lipschitz constants of the message function with TransE are computed as follows.

$$A_{\text{msg}}^{(l)} = 1, B_{\text{msg}}^{(l)} = 0, C_{\text{msg}}^{(l)} = 2\kappa^2 + \kappa, D_{\text{msg}}^{(l)} = \sqrt{2}\kappa^2$$

The *message* function with DistMult is formulated as follows.

$$\mathrm{MSG}^{(l)}(\mathbf{x}_S^{(l-1)}(u), \mathbf{x}_S^{(l-1)}(v), r, q) = \mathrm{diag}\Big(\mathbf{x}_S^{(l-1)}(u)\Big)(\boldsymbol{W}_r^{(l)}\boldsymbol{R}[q] + \mathbf{b}_r^{(l)})$$

where $\boldsymbol{W}_r^{(l)}$ is a learnable weight matrix that satisfies the Assumption A.1, and $\mathbf{b}_r^{(l)}$ is a learnable weight vector that satisfies the Assumption A.1. We can justify the Lipschitz continuity of the *message* function with DistMult as follows.

$$\begin{aligned}
&\|\mathrm{MSG}^{(l)}(\mathbf{x}_{S_1}^{(l-1)}(u_1), \mathbf{x}_{S_1}^{(l-1)}(v_1), r_1, q_1) - \mathrm{MSG}^{(l)}(\mathbf{x}_{S_2}^{(l-1)}(u_2), \mathbf{x}_{S_2}^{(l-1)}(v_2), r_2, q_2)\|_2 \\
=&\|\mathrm{diag}\Big(\mathbf{x}_{S_1}^{(l-1)}(u_1)\Big)(\boldsymbol{W}_{r_1}^{(l)}\boldsymbol{R}[q_1] + \mathbf{b}_{r_1}^{(l)}) - \mathrm{diag}\Big(\mathbf{x}_{S_2}^{(l-1)}(u_2)\Big)(\boldsymbol{W}_{r_2}^{(l)}\boldsymbol{R}[q_2] + \mathbf{b}_{r_2}^{(l)})\|_2 \\
=&\|\mathrm{diag}\Big(\mathbf{x}_{S_1}^{(l-1)}(u_1)\Big)(\boldsymbol{W}_{r_1}^{(l)}\boldsymbol{R}[q_1] + \mathbf{b}_{r_1}^{(l)} - \boldsymbol{W}_{r_2}^{(l)}\boldsymbol{R}[q_2] - \mathbf{b}_{r_2}^{(l)}) + \Big(\mathrm{diag}\Big(\mathbf{x}_{S_1}^{(l-1)}(u_1)\Big) - \mathrm{diag}\Big(\mathbf{x}_{S_2}^{(l-1)}(u_2)\Big)\Big)(\boldsymbol{W}_{r_2}^{(l)}\boldsymbol{R}[q_2] + \mathbf{b}_{r_2}^{(l)})\|_2 \\
=&\|\mathrm{diag}\Big(\mathbf{x}_{S_1}^{(l-1)}(u_1)\Big)(\boldsymbol{W}_{r_1}^{(l)}(\boldsymbol{R}[q_1] - \boldsymbol{R}[q_2]) + (\boldsymbol{W}_{r_1}^{(l)} - \boldsymbol{W}_{r_2}^{(l)})\boldsymbol{R}[q_2] + \mathbf{b}_{r_1}^{(l)} - \mathbf{b}_{r_2}^{(l)}) + \\
&\Big(\mathrm{diag}\Big(\mathbf{x}_{S_1}^{(l-1)}(u_1)\Big) - \mathrm{diag}\Big(\mathbf{x}_{S_2}^{(l-1)}(u_2)\Big)\Big)(\boldsymbol{W}_{r_2}^{(l)}\boldsymbol{R}[q_2] + \mathbf{b}_{r_2}^{(l)})\|_2 \\
\leq&\|\mathrm{diag}\Big(\mathbf{x}_{S_1}^{(l-1)}(u_1)\Big)\|_2\left(\|\boldsymbol{W}_{r_1}^{(l)}\|_2\|\boldsymbol{R}[q_1] - \boldsymbol{R}[q_2]\|_2 + \|\boldsymbol{W}_{r_1}^{(l)} - \boldsymbol{W}_{r_2}^{(l)}\|_2\|\boldsymbol{R}[q_2]\|_2 + \|\mathbf{b}_{r_1}^{(l)} - \mathbf{b}_{r_2}^{(l)}\|_2\right) + \\
&\|\mathrm{diag}\Big(\mathbf{x}_{S_1}^{(l-1)}(u_1)\Big) - \mathrm{diag}\Big(\mathbf{x}_{S_2}^{(l-1)}(u_2)\Big)\|_2\|(\boldsymbol{W}_{r_2}^{(l)}\boldsymbol{R}[q_2] + \mathbf{b}_{r_2}^{(l)})\|_2 \\
\leq&\|\mathrm{diag}\Big(\mathbf{x}_{S_1}^{(l-1)}(u_1)\Big)\|_F\left(\|\boldsymbol{W}_{r_1}^{(l)}\|_2\|\boldsymbol{R}[q_1] - \boldsymbol{R}[q_2]\|_2 + \|\boldsymbol{W}_{r_1}^{(l)} - \boldsymbol{W}_{r_2}^{(l)}\|_2\|\boldsymbol{R}[q_2]\|_2 + \|\mathbf{b}_{r_1}^{(l)} - \mathbf{b}_{r_2}^{(l)}\|_2\right) + \\
&\|\mathrm{diag}\Big(\mathbf{x}_{S_1}^{(l-1)}(u_1)\Big) - \mathrm{diag}\Big(\mathbf{x}_{S_2}^{(l-1)}(u_2)\Big)\|_F\|(\boldsymbol{W}_{r_2}^{(l)}\boldsymbol{R}[q_2] + \mathbf{b}_{r_2}^{(l)})\|_2 \\
\leq&\beta\left(\sqrt{2}\kappa^2\mathbb{1}[q_1 \neq q_2] + 2\kappa^2\mathbb{1}[r_1 \neq r_2] + 2\kappa\mathbb{1}[r_1 \neq r_2]\right) + \|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2(\kappa^2 + \kappa)
\end{aligned}$$

Finally, the Lipschitz constants of the *message* function with DistMult are computed as follows.

$$A_{\text{msg}}^{(l)} = \kappa^2 + \kappa, B_{\text{msg}}^{(l)} = 0, C_{\text{msg}}^{(l)} = 2\beta\kappa^2 + 2\beta\kappa, D_{\text{msg}}^{(l)} = \sqrt{2}\beta\kappa^2$$

The *message* function with RotatE is formulated as follows.

$$\begin{aligned}
&\mathrm{MSG}^{(l)}(\mathbf{x}_S^{(l-1)}(u), \mathbf{x}_S^{(l-1)}(v), r, q) = \\
&\begin{bmatrix}
\begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix}\mathbf{x}_S^{(l-1)}(u) \circ \begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix}(\boldsymbol{W}_r\boldsymbol{R}[q] + \mathbf{b}_r) - \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix}\mathbf{x}_S^{(l-1)}(u) \circ \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix}(\boldsymbol{W}_r\boldsymbol{R}[q] + \mathbf{b}_r) \\
\begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix}\mathbf{x}_S^{(l-1)}(u) \circ \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix}(\boldsymbol{W}_r\boldsymbol{R}[q] + \mathbf{b}_r) + \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix}\mathbf{x}_S^{(l-1)}(u) \circ \begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix}(\boldsymbol{W}_r\boldsymbol{R}[q] + \mathbf{b}_r)
\end{bmatrix}
\end{aligned}$$

where $\circ$ is an element-wise vector multiplication, $\boldsymbol{W}_r^{(l)}$ is a learnable weight matrix that satisfies the Assumption A.1, and $\mathbf{b}_r^{(l)}$ is a learnable weight vector that satisfies the Assumption A.1. Note that the embedding dimension d should be an even number to use the RotatE. Let us show that *message* function with RotatE is Lipschitz continuous. Let

$$\mathbf{x}_1 = \begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix} \mathbf{x}_{S_1}^{(l-1)}(u_1) \circ \begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix} (\boldsymbol{W}_{r_1}\boldsymbol{R}[q_1] + \mathbf{b}_{r_1}) - \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix} \mathbf{x}_{S_1}^{(l-1)}(u_1) \circ \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix} (\boldsymbol{W}_{r_1}\boldsymbol{R}[q_1] + \mathbf{b}_{r_1})$$

$$\mathbf{y}_1 = \begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix} \mathbf{x}_{S_1}^{(l-1)}(u_1) \circ \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix} (\boldsymbol{W}_{r_1}\boldsymbol{R}[q_1] + \mathbf{b}_{r_1}) + \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix} \mathbf{x}_{S_1}^{(l-1)}(u_1) \circ \begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix} (\boldsymbol{W}_{r_1}\boldsymbol{R}[q_1] + \mathbf{b}_{r_1})$$

$$\mathbf{x}_2 = \begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix} \mathbf{x}_{S_2}^{(l-1)}(u_2) \circ \begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix} (\boldsymbol{W}_{r_2}\boldsymbol{R}[q_2] + \mathbf{b}_{r_2}) - \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix} \mathbf{x}_{S_2}^{(l-1)}(u_1) \circ \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix} (\boldsymbol{W}_{r_1}\boldsymbol{R}[q_1] + \mathbf{b}_{r_2})$$

$$\mathbf{y}_2 = \begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix} \mathbf{x}_{S_2}^{(l-1)}(u_2) \circ \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix} (\boldsymbol{W}_{r_2}\boldsymbol{R}[q_2] + \mathbf{b}_{r_2}) + \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix} \mathbf{x}_{S_2}^{(l-1)}(u_1) \circ \begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix} (\boldsymbol{W}_{r_1}\boldsymbol{R}[q_1] + \mathbf{b}_{r_2})$$

By applying the method for deriving the Lipschitz constant for DistMult and

$$\| \begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix} \mathbf{x}\|_2 \leq \|\mathbf{x}\|_2, \quad \| \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix} \mathbf{x}\|_2 \leq \|\mathbf{x}\|_2$$

for a vector $\mathbf{x} \in \mathbb{R}^d$, we get the following inequalities.

$$\begin{aligned}
&\|\mathbf{x}_1 - \mathbf{x}_2\|_2 \\
\leq& 2\|\mathbf{x}_{S_1}^{(l-1)}(u_1)\|_2 \left(\|\boldsymbol{W}_{r_1}^{(l)}\|_2\|\boldsymbol{R}[q_1] - \boldsymbol{R}[q_2]\|_2 + \|\boldsymbol{W}_{r_1}^{(l)} - \boldsymbol{W}_{r_2}^{(l)}\|_2\|\boldsymbol{R}[q_2]\|_2 + \|\mathbf{b}_{r_1}^{(l)} - \mathbf{b}_{r_2}^{(l)}\|_2\right) + \\
&2\|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2\|(\boldsymbol{W}_{r_2}^{(l)}\boldsymbol{R}[q_2] + \mathbf{b}_{r_2}^{(l)})\|_2 \\
\leq& 2\beta\left(\sqrt{2}\kappa^2\mathbb{1}[q_1 \neq q_2] + 2\kappa^2\mathbb{1}[r_1 \neq r_2] + 2\kappa\mathbb{1}[r_1 \neq r_2]\right) + 2\|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2(\kappa^2 + \kappa)
\end{aligned}$$

$$\begin{aligned}
&\|\mathbf{y}_1 - \mathbf{y}_2\|_2 \\
\leq& 2\|\mathbf{x}_{S_1}^{(l-1)}(u_1)\|_2 \left(\|\boldsymbol{W}_{r_1}^{(l)}\|_2\|\boldsymbol{R}[q_1] - \boldsymbol{R}[q_2]\|_2 + \|\boldsymbol{W}_{r_1}^{(l)} - \boldsymbol{W}_{r_2}^{(l)}\|_2\|\boldsymbol{R}[q_2]\|_2 + \|\mathbf{b}_{r_1}^{(l)} - \mathbf{b}_{r_2}^{(l)}\|_2\right) + \\
&2\|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2\|(\boldsymbol{W}_{r_2}^{(l)}\boldsymbol{R}[q_2] + \mathbf{b}_{r_2}^{(l)})\|_2 \\
\leq& 2\beta\left(\sqrt{2}\kappa^2\mathbb{1}[q_1 \neq q_2] + 2\kappa^2\mathbb{1}[r_1 \neq r_2] + 2\kappa\mathbb{1}[r_1 \neq r_2]\right) + 2\|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2(\kappa^2 + \kappa)
\end{aligned}$$

Then,

$$\begin{aligned}
&\|\mathrm{MSG}^{(l)}(\mathbf{x}_{S_1}^{(l-1)}(u_1), \mathbf{x}_{S_1}^{(l-1)}(v_1), r_1, q_1) - \mathrm{MSG}^{(l)}(\mathbf{x}_{S_2}^{(l-1)}(u_2), \mathbf{x}_{S_2}^{(l-1)}(v_2), r_2, q_2)\|_2 \\
=&\| \begin{bmatrix} \begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix} \mathbf{x}_{S_1}^{(l-1)}(u_1) \circ \begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix} (\boldsymbol{W}_{r_1}\boldsymbol{R}[q_1] + \mathbf{b}_{r_1}) - \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix} \mathbf{x}_{S_1}^{(l-1)}(u_1) \circ \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix} (\boldsymbol{W}_{r_1}\boldsymbol{R}[q_1] + \mathbf{b}_{r_1}) \\ \begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix} \mathbf{x}_{S_1}^{(l-1)}(u_1) \circ \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix} (\boldsymbol{W}_{r_1}\boldsymbol{R}[q_1] + \mathbf{b}_{r_1}) + \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix} \mathbf{x}_{S_1}^{(l-1)}(u_1) \circ \begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix} (\boldsymbol{W}_{r_1}\boldsymbol{R}[q_1] + \mathbf{b}_{r_1}) \end{bmatrix} - \\
&\begin{bmatrix} \begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix} \mathbf{x}_{S_2}^{(l-1)}(u_2) \circ \begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix} (\boldsymbol{W}_{r_2}\boldsymbol{R}[q_2] + \mathbf{b}_{r_2}) - \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix} \mathbf{x}_{S_2}^{(l-1)}(u_1) \circ \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix} (\boldsymbol{W}_{r_1}\boldsymbol{R}[q_1] + \mathbf{b}_{r_2}) \\ \begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix} \mathbf{x}_{S_2}^{(l-1)}(u_2) \circ \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix} (\boldsymbol{W}_{r_2}\boldsymbol{R}[q_2] + \mathbf{b}_{r_2}) + \begin{bmatrix}\mathbf{0}_{\frac{d}{2},\frac{d}{2}} & \boldsymbol{I}_{\frac{d}{2}}\end{bmatrix} \mathbf{x}_{S_2}^{(l-1)}(u_1) \circ \begin{bmatrix}\boldsymbol{I}_{\frac{d}{2}} & \mathbf{0}_{\frac{d}{2},\frac{d}{2}}\end{bmatrix} (\boldsymbol{W}_{r_1}\boldsymbol{R}[q_1] + \mathbf{b}_{r_2}) \end{bmatrix} \|_2 \\
\leq&\| \begin{bmatrix}\mathbf{x}_1 \\ \mathbf{y}_1\end{bmatrix} - \begin{bmatrix}\mathbf{x}_2 \\ \mathbf{y}_2\end{bmatrix} \|_2 \\
\leq&\|\mathbf{x}_1 - \mathbf{x}_2\|_2 + \|\mathbf{y}_1 - \mathbf{y}_2\|_2 \\
\leq& 4\beta\left(\sqrt{2}\kappa^2\mathbb{1}[q_1 \neq q_2] + 2\kappa^2\mathbb{1}[r_1 \neq r_2] + 2\kappa\mathbb{1}[r_1 \neq r_2]\right) + 4\|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2(\kappa^2 + \kappa)
\end{aligned}$$

Finally, the Lipschitz constants of the *message* function with RotatE are computed as follows.

$$A_{\text{msg}}^{(l)} = 4\kappa^2 + 4\kappa, B_{\text{msg}}^{(l)} = 0, C_{\text{msg}}^{(l)} = 8\beta\kappa^2 + 8\beta\kappa, D_{\text{msg}}^{(l)} = 4\sqrt{2}\beta\kappa^2$$

History Function The *history* function of NBFNet is $\theta(k) = 0$.

Aggregation Function NBFNet uses sum, mean, and max/min pooling aggregator as the *aggregation* function. First, the sum aggregator of NBFNet is formulated as follows.

$$\text{AGG}^{(l)}(\mathcal{A} = \{\{\mathbf{a}_i\}\}_{i=1}^n) = \sum_{i=1}^n \mathbf{a}_i$$

The Lipschitz continuity of the sum aggregator of NBFNet is formally established as follows.

$$\begin{aligned}\|\text{AGG}^{(l)}(\mathcal{A} = \{\{\mathbf{a}_i\}\}_{i=1}^n) - \text{AGG}^{(l)}(\mathcal{B} = \{\{\mathbf{b}_j\}\}_{j=1}^n)\|_2 =& \|\sum_{i=1}^n \mathbf{a}_i - \sum_{j=1}^n \mathbf{b}_j\|_2 \\ =& \|\sum_{i,j} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j](\mathbf{a}_i - \mathbf{b}_j)\|_2 \\ \leq& \sum_{i,j} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j]\|\mathbf{a}_i - \mathbf{b}_j\|_2\end{aligned}$$

where $\boldsymbol{P}_{\mathcal{A},\mathcal{B}}$ is a transportation plan between $\mathcal{A}, \mathcal{B}$. Finally, the Lipschitz constant of the sum aggregator of NBFNet is computed as follows.

$$A_{\text{agg}}^{(l)} = 1$$

Also, since adding zero vectors to the sum of vectors does not change the resulting vector, the embedding vector Φ_{agg} is a zero vector. The mean aggregator of NBFNet is formulated as follows.

$$\text{AGG}^{(l)}(\mathcal{A} = \{\{\mathbf{a}_i\}\}_{i=1}^n) = \frac{1}{n}\sum_{i=1}^n \mathbf{a}_i$$

We can justify the Lipschitz continuity of the mean aggregator of NBFNet as follows.

$$\begin{aligned}\|\text{AGG}^{(l)}(\mathcal{A} = \{\{\mathbf{a}_i\}\}_{i=1}^n) - \text{AGG}^{(l)}(\mathcal{B} = \{\{\mathbf{b}_j\}\}_{j=1}^n)\|_2 =& \|\frac{1}{n}\sum_{i=1}^n \mathbf{a}_i - \frac{1}{n}\sum_{j=1}^n \mathbf{b}_j\|_2 \\ =& \frac{1}{n}\|\sum_{i,j} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j](\mathbf{a}_i - \mathbf{b}_j)\|_2 \\ \leq& \frac{1}{n}\sum_{i,j} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j]\|\mathbf{a}_i - \mathbf{b}_j\|_2\end{aligned}$$

where $\boldsymbol{P}_{\mathcal{A},\mathcal{B}}$ is a transportation plan between $\mathcal{A}, \mathcal{B}$ and n is the maximum of the degree of the training KG and the inference KG since the blank tree augmentation defined in Definition B.1. Finally, the Lipschitz constant of the mean aggregator of NBFNet is computed as follows.

$$A_{\text{agg}}^{(l)} = \frac{1}{d_{\max}}$$

Also, since $\text{AGG}(\{\{\mathbf{a}_i\}\}_{i=1}^n) = \text{AGG}(\{\{\mathbf{a}_i\}\}_{i=1}^n \cup \{\{\frac{1}{n}\sum_{i=1}^n \mathbf{a}_i\}\})$, the embedding vector $\Phi_{\text{agg}} = \frac{1}{n}\sum_{i=1}^n \mathbf{a}_i$.

The max pooling aggregator of NBFNet is formulated as follows.

$$\text{AGG}^{(l)}(\mathcal{A} = \{\{\mathbf{a}_i\}\}_{i=1}^n) = \Psi_{\mathcal{A}} \quad \text{where } \Psi_{\mathcal{A}}[k] = \max_i \mathbf{a}_i[k]$$

We prove the Lipschitz continuity of the max pooling aggregator of NBFNet as follows.

$$\begin{aligned}&\|\text{AGG}^{(l)}(\mathcal{A} = \{\{\mathbf{a}_i\}\}_{i=1}^n) - \text{AGG}^{(l)}(\mathcal{B} = \{\{\mathbf{b}_j\}\}_{j=1}^n)\|_2 \\ =&\|\Psi_{\mathcal{A}} - \Psi_{\mathcal{B}}\|_2 \\ \leq&\|\Psi_{\mathcal{A}} - \Psi_{\mathcal{B}}\|_1\end{aligned}$$

$$=\sum_{k=1}^{d} |\max_i \mathbf{a}_i[k] - \max_j \mathbf{b}_j[k]|$$

Without loss of generality, we assume $\max_i \mathbf{a}_i[k] \geq \max_j \mathbf{b}_j[k]$. Let $i_k = \text{argmax}_i(\mathbf{a}_i[k])$. Then by the definition of transportation plan, we can ensure the existence of j_k that satisfies $\boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i_k, j_k] > 0$. Also, there always exists a non-zero minimum value of $\boldsymbol{P}_{\mathcal{A},\mathcal{B}}$. Then,

$$\begin{aligned}
\sum_{k=1}^{d} |\max_i \mathbf{a}_i[k] - \max_j \mathbf{b}_j[k]| =& \sum_{k=1}^{d} |\mathbf{a}_{i_k}[k] - \max_j \mathbf{b}_j[k]| \\
\leq& \sum_{k=1}^{d} |\mathbf{a}_{i_k}[k] - \mathbf{b}_{j_k}[k]| \\
=& \sum_{k=1}^{d} \frac{1}{\boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i_k, j_k]} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i_k, j_k] |\mathbf{a}_{i_k}[k] - \mathbf{b}_{j_k}[k]| \\
\leq& \sum_{k=1}^{d} \frac{1}{\boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i_k, j_k]} \sum_{i,j} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i, j] |\mathbf{a}_i[k] - \mathbf{b}_j[k]| \\
\leq& \frac{1}{\min_{i,j,\boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j]>0} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j]} \sum_{k=1}^{d} \sum_{i,j} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i, j] |\mathbf{a}_i[k] - \mathbf{b}_j[k]| \\
=& \frac{1}{\min_{i,j,\boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j]>0} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j]} \sum_{i,j} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i, j] \sum_{k=1}^{d} |\mathbf{a}_i[k] - \mathbf{b}_j[k]| \\
=& \frac{1}{\min_{i,j,\boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j]>0} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j]} \sum_{i,j} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i, j] \|\mathbf{a}_i - \mathbf{b}_i\|_1 \\
\leq& \frac{\sqrt{d}}{\min_{i,j,\boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j]>0} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j]} \sum_{i,j} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i, j] \|\mathbf{a}_i - \mathbf{b}_i\|_2
\end{aligned}$$

where d is the dimension of the vectors. Finally, If we set K as the minimum value among all non-zero values of the optimal transportation plans for l-th layer of all computation trees, the Lipschitz constant of the max pooling function of NBFNet is computed as follows.

$$A_{\text{agg}}^{(l)} = \frac{\sqrt{d}}{K}$$

Also, since $\text{AGG}(\{\{\mathbf{a}_i\}\}_{i=1}^{n}) = \text{AGG}(\{\{\mathbf{a}_i\}\}_{i=1}^{n} \cup \{\{\Psi_{\mathcal{A}}\}\})$, the embedding vector $\Phi_{\text{agg}} = \Psi_{\mathcal{A}}$.

The min pooling aggregator of NBFNet is formulated as follows.

$$\text{AGG}^{(l)}(\mathcal{A} = \{\{\mathbf{a}_i\}\}_{i=1}^{n}) = \Psi_{\mathcal{A}} \quad \text{where } \Psi_{\mathcal{A}}[k] = \min_i \mathbf{a}_i[k]$$

We prove the Lipschitz continuity of the min pooling aggregator of NBFNet as follows.

$$\begin{aligned}
&\|\text{AGG}^{(l)}(\mathcal{A} = \{\{\mathbf{a}_i\}\}_{i=1}^{n}) - \text{AGG}^{(l)}(\mathcal{B} = \{\{\mathbf{b}_j\}\}_{j=1}^{n})\|_2 \\
=&\|\Psi_{\mathcal{A}} - \Psi_{\mathcal{B}}\|_2 \\
\leq&\|\Psi_{\mathcal{A}} - \Psi_{\mathcal{B}}\|_1 \\
=&\sum_{k=1}^{d} |\min_i \mathbf{a}_i[k] - \min_j \mathbf{b}_j[k]|
\end{aligned}$$

Without loss of generality, we assume the $\min_i \mathbf{a}_i[k] \leq \min_j \mathbf{b}_j[k]$. Let $i_k = \text{argmin}_i(\mathbf{a}_i[k])$, then by the definition of the transportation plan, we can ensure that the existence of the j_k that satisfies $\boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i_k, j_k] > 0$. Also, there always exists the non-zero minimum value of $\boldsymbol{P}_{\mathcal{A},\mathcal{B}}$, then

$$\begin{aligned}
\sum_{k=1}^{d} |\min_i \mathbf{a}_i[k] - \min_j \mathbf{b}_j[k]| =& \sum_{k=1}^{d} |\mathbf{a}_{i_k}[k] - \min_j \mathbf{b}_j[k]| \\
\leq& \sum_{k=1}^{d} |\mathbf{a}_{i_k}[k] - \mathbf{b}_{j_k}[k]| \\
=& \sum_{k=1}^{d} \frac{1}{\boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i_k, j_k]} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i_k, j_k] |\mathbf{a}_{i_k}[k] - \mathbf{b}_{j_k}[k]| \\
\leq& \sum_{k=1}^{d} \frac{1}{\boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i_k, j_k]} \sum_{i,j} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i, j] |\mathbf{a}_i[k] - \mathbf{b}_j[k]| \\
\leq& \frac{1}{\min_{i,j,\boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j]>0} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i, j]} \sum_{k=1}^{d} \sum_{i,j} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i, j] |\mathbf{a}_i[k] - \mathbf{b}_j[k]| \\
=& \frac{1}{\min_{i,j,\boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j]>0} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i, j]} \sum_{i,j} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i, j] \sum_{k=1}^{d} |\mathbf{a}_i[k] - \mathbf{b}_j[k]| \\
=& \frac{1}{\min_{i,j,\boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j]>0} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i, j]} \sum_{i,j} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i, j] \|\mathbf{a}_i - \mathbf{b}_i\|_1 \\
=& \frac{\sqrt{d}}{\min_{i,j,\boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j]>0} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i, j]} \sum_{i,j} \boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i, j] \|\mathbf{a}_i - \mathbf{b}_i\|_2
\end{aligned}$$

where d is the dimension of the vectors. Finally, the Lipschitz constant of the min pooling function of NBFNet is computed as follows.

$$A_{\text{agg}}^{(l)} = \frac{\sqrt{d}}{\min_{i,j,\boldsymbol{P}_{\mathcal{A},\mathcal{B}}[i,j]>0}}$$

Also, since $\text{AGG}(\{\{\mathbf{a}_i\}\}_{i=1}^{n}) = \text{AGG}(\{\{\mathbf{a}_i\}\}_{i=1}^{n} \cup \{\{\Psi_{\mathcal{A}}\}\})$, the embedding vector $\Phi_{\text{agg}} = \Psi_{\mathcal{A}}$.

Update Function NBFNet aggregates $\mathbf{x}_S^{(\theta(l))}(v)$ together with the messages from neighboring entities during the aggregation process. Therefore, *update* function of NBFNet can be represented as a linear projection applied to the concatenation of the two input vectors.

$$\text{UPD}^{(l)}(\mathbf{a}_1, \mathbf{a}_2) = \sigma\left(\boldsymbol{W}_1^{(l)}\mathbf{a}_1 + \boldsymbol{W}_2^{(l)}\mathbf{a}_2\right)$$

where σ is an activation function with the Lipschitz constant 1, and $\boldsymbol{W}_1^{(l)}$ is a learnable weight matrix that satisfies the Assumption A.1. The Lipschitz constants of the *update* function of NBFNet are computed as follows.

$$A_{\text{upd}}^{(l)} = \kappa, B_{\text{upd}}^{(l)} = \kappa$$

Global Readout Function The *global-readout* function is not used in NBFNet.

$$\text{GRD}(\mathcal{A} = \{\{\mathbf{a}_i\}\}_{i=1}^{n}) = \mathbf{0}$$

Therefore, the Lipschitz constant of the *global-readout* function is computed as zero.

$$A_{\text{grd}} = 0$$

Readout Function Since NBFNet uses only the final representation of the tail entity t to calculate the final score of the subgraph, the *readout* function of NBFNet is formulated as follows.

$$\text{RD}\left(\mathbf{x}_S^{(L)}(h), \mathbf{x}_S^{(L)}(t), \text{GRD}(\{\{\mathbf{x}_S^{(L)}(u) | u \in \mathcal{V}_S\}\}), q\right) = \text{MLP}(\mathbf{x}_S^{(L)}(t))$$

Note that the MLP is Lipschitz continuous under the Assumption A.1 and let the Lipschitz constant of the MLP be C_{mlp}. Then, we can prove the Lipschitz continuity of the *readout* function of NBFNet as follows.

$$\begin{aligned}
&|\text{RD}\left(\mathbf{x}_{S_1}^{(L)}(h_1), \mathbf{x}_{S_1}^{(L)}(t_1), \text{GRD}(\{\{\mathbf{x}_{S_1}^{(L)}(u_1)|u_1 \in \mathcal{V}_{S_1}\}\}), q_1\right) \\
&\quad - \text{RD}\left(\mathbf{x}_{S_2}^{(L)}(h_2), \mathbf{x}_{S_2}^{(L)}(t_2), \text{GRD}(\{\{\mathbf{x}_{S_2}^{(L)}(u_2)|u_2 \in \mathcal{V}_{S_2}\}\}), q_2\right)| \\
=&|\text{MLP}(\mathbf{x}_{S_1}^{(L)}(t_1)) - \text{MLP}(\mathbf{x}_{S_2}^{(L)}(t_2))| \\
=&C_{\text{mlp}}|\mathbf{x}_{S_1}^{(L)}(t_1) - \mathbf{x}_{S_2}^{(L)}(t_2)|
\end{aligned}$$

Finally, the Lipschitz constants of the *readout* function of NBFNet are computed as follows.

$$A_{\text{rd}} = 0, B_{\text{rd}} = C_{\text{mlp}}, C_{\text{rd}} = 0, D_{\text{rd}} = 0$$

A.3. Instantiation of RED-GNN

RED-GNN (Zhang & Yao, 2022) can be instantiated as follows:

A.3.1. Subgraph Extraction

RED-GNN extracts a relational digraph for a given triplet. However, the final embedding vector of the tail entity is computed using only the embedding vectors of the L-hop neighbor entities of the head entity and the tail entity. Therefore, the subgraph extractor in RED-GNN can be defined as a function that extracts the subgraphs constructed by the union of L-hop neighbor entities from the head entity and tail entity.

A.3.2. Initialization

RED-GNN initializes the embedding vectors of all entities in the subgraph as zero-vectors.

A.3.3. Subgraph Message-Passing Neural Networks

Message Function The *message* function of RED-GNN is formulated as follows.

$$\text{MSG}^{(l)}(\mathbf{x}_S^{(l-1)}(u), \mathbf{x}_S^{(l-1)}(v), r, q) = \alpha_{u,r,q}^{(l)}(\mathbf{x}_S^{(l-1)}(u) + \boldsymbol{R}^{(l)}[r])$$

$$\alpha_{u,r,q}^{(l)} = \sigma(\mathbf{w}_\alpha^{(l)} \cdot \text{ReLU}(\boldsymbol{W}_1^{(l)}\mathbf{x}_S^{(l-1)}(u) + \boldsymbol{W}_2^{(l)}\boldsymbol{R}^{(l)}[r] + \boldsymbol{W}_3^{(l)}\boldsymbol{R}^{(l)}[q]))$$

where $\boldsymbol{R}^{(l)}$ is a relation embedding matrix that satisfies the Assumption A.1, $\boldsymbol{W}_1^{(l)}, \boldsymbol{W}_2^{(l)}, \boldsymbol{W}_3^{(l)}$ are learnable weight matrices that satisfy the Assumption A.1, and $\mathbf{w}_\alpha^{(l)}$ is a learnable weight vector that satisfy the Assumption A.1.

The Lipschitz continuity of the *message* function of RED-GNN can be shown as follows.

$$\begin{aligned}
&\|\text{MSG}^{(l)}(\mathbf{x}_{S_1}^{(l-1)}(u_1), \mathbf{x}_{S_1}^{(l-1)}(v_1), r_1, q_1) - \text{MSG}^{(l)}(\mathbf{x}_{S_2}^{(l-1)}(u_2), \mathbf{x}_{S_2}^{(l-1)}(v_2), r_2, q_2)\|_2 \\
=&\|\alpha_{u_1,r_1,q_1}^{(l)}(\mathbf{x}_{S_1}^{(l-1)}(u_1) + \boldsymbol{R}^{(l)}[r_1]) - \alpha_{u_2,r_2,q_2}^{(l)}(\mathbf{x}_{S_2}^{(l-1)}(u_2) + \boldsymbol{R}^{(l)}[r_2])\|_2 \\
\le&\|\alpha_{u_1,r_1,q_1}^{(l)}(\mathbf{x}_{S_1}^{(l-1)}(u_1) + \boldsymbol{R}^{(l)}[r_1] - \mathbf{x}_{S_2}^{(l-1)}(u_2) - \boldsymbol{R}^{(l)}[r_2]) + (\alpha_{u_1,r_1,q_1}^{(l)} - \alpha_{u_2,r_2,q_2}^{(l)})(\mathbf{x}_{S_2}^{(l-1)}(u_2) + \boldsymbol{R}^{(l)}[r_2])\|_2 \\
\le&|\alpha_{u_1,r_1,q_1}^{(l)}|(\|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + \|\boldsymbol{R}^{(l)}[r_1] - \boldsymbol{R}^{(l)}[r_2]\|_2) + |\alpha_{u_1,r_1,q_1}^{(l)} - \alpha_{u_2,r_2,q_2}^{(l)}|\|\mathbf{x}_{S_2}^{(l-1)}(u_2) + \boldsymbol{R}^{(l)}[r_2]\|_2 \\
\le&(\|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + \sqrt{2}\kappa\mathbb{1}[r_1 \neq r_2]) + (\kappa + \beta)|\alpha_{u_1,r_1,q_1}^{(l)} - \alpha_{u_2,r_2,q_2}^{(l)}|
\end{aligned}$$

For the attention value $\alpha_{u_1,r_1,q_1}^{(l)}$, the following inequalities hold.

$$\begin{aligned}
&|\alpha_{u_1,r_1,q_1}^{(l)} - \alpha_{u_2,r_2,q_2}^{(l)}| \\
=&|\sigma(\mathbf{w}_\alpha^{(l)} \cdot \text{ReLU}(\boldsymbol{W}_1^{(l)}\mathbf{x}_{S_1}^{(l-1)}(u_1) + \boldsymbol{W}_2^{(l)}\boldsymbol{R}^{(l)}[r_1] + \boldsymbol{W}_3^{(l)}\boldsymbol{R}^{(l)}[q_1])) \\
&\quad - \sigma(\mathbf{w}_\alpha^{(l)} \cdot \text{ReLU}(\boldsymbol{W}_1^{(l)}\mathbf{x}_{S_2}^{(l-1)}(u_2) + \boldsymbol{W}_2^{(l)}\boldsymbol{R}^{(l)}[r_2] + \boldsymbol{W}_3^{(l)}\boldsymbol{R}^{(l)}[q_2]))| \\
\le&|\mathbf{w}_\alpha^{(l)} \cdot \text{ReLU}(\boldsymbol{W}_1^{(l)}\mathbf{x}_{S_1}^{(l-1)}(u_1) + \boldsymbol{W}_2^{(l)}\boldsymbol{R}^{(l)}[r_1] + \boldsymbol{W}_3^{(l)}\boldsymbol{R}^{(l)}[q_1])
\end{aligned}$$

$$
\begin{aligned}
&- \mathbf{w}_\alpha^{(l)} \cdot \mathrm{ReLU}(\boldsymbol{W}_1^{(l)} \mathbf{x}_{S_2}^{(l-1)}(u_2) + \boldsymbol{W}_2^{(l)}\boldsymbol{R}^{(l)}[r_2] + \boldsymbol{W}_3^{(l)}\boldsymbol{R}^{(l)}[q_2])| \\
=&|\mathbf{w}_\alpha^{(l)} \cdot (\mathrm{ReLU}(\boldsymbol{W}_1^{(l)} \mathbf{x}_{S_1}^{(l-1)}(u_1) + \boldsymbol{W}_2^{(l)}\boldsymbol{R}^{(l)}[r_1] + \boldsymbol{W}_3^{(l)}\boldsymbol{R}^{(l)}[q_1]) - \mathrm{ReLU}(\boldsymbol{W}_1^{(l)} \mathbf{x}_{S_2}^{(l-1)}(u_2) + \boldsymbol{W}_2^{(l)}\boldsymbol{R}^{(l)}[r_2] + \boldsymbol{W}_3^{(l)}\boldsymbol{R}^{(l)}[q_2]))| \\
\le&\|\mathbf{w}_\alpha^{(l)}\|_2 \|\mathrm{ReLU}(\boldsymbol{W}_1^{(l)} \mathbf{x}_{S_1}^{(l-1)}(u_1) + \boldsymbol{W}_2^{(l)}\boldsymbol{R}^{(l)}[r_1] + \boldsymbol{W}_3^{(l)}\boldsymbol{R}^{(l)}[q_1]) \\
&- \mathrm{ReLU}(\boldsymbol{W}_1^{(l)} \mathbf{x}_{S_2}^{(l-1)}(u_2) + \boldsymbol{W}_2^{(l)}\boldsymbol{R}^{(l)}[r_2] + \boldsymbol{W}_3^{(l)}\boldsymbol{R}^{(l)}[q_2])\|_2 \\
\le&\|\mathbf{w}_\alpha^{(l)}\|_2 \|\boldsymbol{W}_1^{(l)} \mathbf{x}_{S_1}^{(l-1)}(u_1) + \boldsymbol{W}_2^{(l)}\boldsymbol{R}^{(l)}[r_1] + \boldsymbol{W}_3^{(l)}\boldsymbol{R}^{(l)}[q_1] - \boldsymbol{W}_1^{(l)} \mathbf{x}_{S_2}^{(l-1)}(u_2) - \boldsymbol{W}_2^{(l)}\boldsymbol{R}^{(l)}[r_2] - \boldsymbol{W}_3^{(l)}\boldsymbol{R}^{(l)}[q_2]\|_2 \\
=&\|\mathbf{w}_\alpha^{(l)}\|_2 \|\boldsymbol{W}_1^{(l)} (\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)) + \boldsymbol{W}_2^{(l)}(\boldsymbol{R}^{(l)}[r_1] - \boldsymbol{R}^{(l)}[r_2]) + \boldsymbol{W}_3^{(l)}(\boldsymbol{R}^{(l)}[q_1] - \boldsymbol{R}^{(l)}[q_2])\|_2 \\
\le&\|\mathbf{w}_\alpha^{(l)}\|_2 \left(\|\boldsymbol{W}_1^{(l)}\|_2 \|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + \|\boldsymbol{W}_2^{(l)}\|_2\|\boldsymbol{R}^{(l)}[r_1] - \boldsymbol{R}^{(l)}[r_2]\|_2 + \|\boldsymbol{W}_3^{(l)}\|_2\|\boldsymbol{R}^{(l)}[q_1] - \boldsymbol{R}^{(l)}[q_2]\|_2\right) \\
\le&\kappa \left(\kappa \|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + \sqrt{2}\kappa^2 \mathbb{1}[r_1 \neq r_2] + \sqrt{2}\kappa^2 \mathbb{1}[q_1 \neq q_2]\right) \\
=&\kappa^2 \|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + \sqrt{2}\kappa^3 \mathbb{1}[r_1 \neq r_2] + \sqrt{2}\kappa^3 \mathbb{1}[q_1 \neq q_2]
\end{aligned}
$$

Therefore, we get

$$
\begin{aligned}
&\|\mathrm{MSG}^{(l)}(\mathbf{x}_{S_1}^{(l-1)}(u_1), \mathbf{x}_{S_1}^{(l-1)}(v_1), r_1, q_1) - \mathrm{MSG}^{(l)}(\mathbf{x}_{S_2}^{(l-1)}(u_2), \mathbf{x}_{S_2}^{(l-1)}(v_2), r_2, q_2)\|_2 \\
\le&\|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 + \sqrt{2}\kappa \mathbb{1}[r_1 \neq r_2] + (\kappa + \beta)(\kappa^2 \|\mathbf{x}_{S_1}^{(l-1)}(u_1) - \mathbf{x}_{S_2}^{(l-1)}(u_2)\|_2 \\
&+ \sqrt{2}\kappa^3 \mathbb{1}[r_1 \neq r_2] + \sqrt{2}\kappa^3 \mathbb{1}[q_1 \neq q_2])
\end{aligned}
$$

Finally, the Lipschitz constants of the *message* function of RED-GNN are computed as follows.

$$
A_{\text{msg}}^{(l)} = \kappa^3 + \beta\kappa^2 + 1, B_{\text{msg}}^{(l)} = 0, C_{\text{msg}}^{(l)} = \sqrt{2}\kappa^4 + \sqrt{2}\beta\kappa^3 + \sqrt{2}\kappa, D_{\text{msg}}^{(l)} = \sqrt{2}\kappa^4 + \sqrt{2}\beta\kappa^3
$$

History Function Since RED-GNN does not utilize $\mathbf{x}_S^{(\theta(l))}(v)$ in its *update* function, it can be instantiated by the model with both $\theta(k) = k - 1$ and $\theta(k) = 0$.

Aggregation Function RED-GNN uses sum aggregation as the *aggregation* function. Therefore, the Lipschitz constant of the *aggregation* function of RED-GNN is computed as follows.

$$
A_{\text{agg}}^{(l)} = 1
$$

Update Function RED-GNN does not utilize $\mathbf{x}_S^{(\theta(l))}(v)$ in the *update* function. Therefore, the *update* function of RED-GNN can be formulated as follows.

$$
\mathrm{UPD}^{(l)}\left(\mathbf{x}_S^{(\theta(l))}(v), \mathbf{a}_S^{(l)}(v)\right) = \delta(\boldsymbol{W}^{(l)}\mathbf{a}_S^{(l)}(v))
$$

where δ is an activation function with the Lipschitz constant 1, $\boldsymbol{W}^{(l)}$ is a learnable weight matrix that satisfies the Assumption A.1. The Lipschitz continuity of the *update* function of RED-GNN can be shown as follows.

$$
\begin{aligned}
&\|\mathrm{UPD}^{(l)}\left(\mathbf{x}_{S_1}^{(\theta(l))}(v_1), \mathbf{a}_{S_1}^{(l)}(v_1)\right) - \mathrm{UPD}^{(l)}\left(\mathbf{x}_{S_2}^{(\theta(l))}(v_2), \mathbf{a}_{S_2}^{(l)}(v_2)\right)\|_2 \\
=&\|\boldsymbol{W}^{(l)}\mathbf{a}_{S_1}^{(l)}(v_1) - \boldsymbol{W}^{(l)}\mathbf{a}_{S_2}^{(l)}(v_2)\|_2 \\
=&\|\boldsymbol{W}^{(l)}\|_2\|\mathbf{a}_{S_1}^{(l)}(v_1) - \mathbf{a}_{S_2}^{(l)}(v_2)\|_2 \\
=&\kappa\|\mathbf{a}_{S_1}^{(l)}(v_1) - \mathbf{a}_{S_2}^{(l)}(v_2)\|_2
\end{aligned}
$$

Therefore, the Lipschitz constants of the *update* function of RED-GNN are computed as follows.

$$
A_{\text{upd}}^{(l)} = 0, B_{\text{upd}}^{(l)} = \kappa
$$

Global Readout Function The *global-readout* function is not used in RED-GNN.

$$\text{GRD}(\mathcal{A} = \{\mathbf{a}_i\}_{i=1}^n) = \mathbf{0}$$

Therefore, the Lipschitz constant of the *global-readout* function is computed as zero.

$$A_{\text{grd}} = 0$$

Readout Function Since RED-GNN uses only the final representation of the tail entity t to calculate the final score of the subgraph, the *readout* function of NBFNet is formulated as follows.

$$\text{RD}\left(\mathbf{x}_S^{(L)}(h), \mathbf{x}_S^{(L)}(t), \text{GRD}(\{\{\mathbf{x}_S^{(L)}(u) | u \in \mathcal{V}_S\}\}), q\right) = \mathbf{w} \cdot \mathbf{x}_S^{(L)}(t)$$

where $\mathbf{w}$ is a learnable weight vector that satisfies the Assumption A.1. The Lipschitz constants of the readout function of NBFNet are computed as follows.

$$A_{\text{rd}} = 0, B_{\text{rd}} = \kappa, C_{\text{rd}} = 0, D_{\text{rd}} = 0$$

B. Blank Tree Augmentation

Similar to Chuang & Jegelka (2022), we introduce blank tree augmentation for the relational computation trees. To handle potential variations in the number of the subtrees, we define a blank tree T_0, which consists of a single virtual root entity. This virtual root entity has no connections to any other entities within the subgraph and is assigned a unique label distinct from the labels of all other entities. The blank tree T_0 is used to augment the multisets of subtrees, ensuring that their sizes are equal. The augmentation process is formalized as follows:

Definition B.1 (Blank Tree Augmentation)**.** Given two multisets of subtrees $\text{SUB}(T_{S_1}^{(l)}(v_1))$, $\text{SUB}(T_{S_2}^{(l)}(v_2))$, a blank tree augmentation ρ is defined by

$$\rho(\text{SUB}(T_{S_1}^{(l)}(v_1)), \text{SUB}(T_{S_2}^{(l)}(v_2))) = \begin{cases} \text{SUB}(T_{S_1}^{(l)}(v_1)) \cup \Theta_{n_2 - n_1}, \text{SUB}(T_{S_2}^{(l)}(v_2)) & n_1 < n_2 \\ \text{SUB}(T_{S_1}^{(l)}(v_1)), \text{SUB}(T_{S_2}^{(l)}(v_2)) \cup \Theta_{n_1 - n_2} & n_1 \geq n_2 \end{cases}$$

where $\Theta_n = \bigcup_{k=1}^n \{\{(r_{\text{blank}}, T_0)\}\}$, $n_1 = |\text{SUB}(T_{S_1}^{(l)}(v_1))|$, $n_2 = |\text{SUB}(T_{S_2}^{(l)}(v_2))|$, and r_{blank} is a virtual relation.

C. Proof for Lipschitz Continuity of Subgraph Message-Passing Neural Networks

We provide a proof for Theorem 4.5.

Theorem 4.5 (Lipschitz Constant of SMPNNs)**.** *Given an* SMPNN $f_{\mathbf{w}}$ *with* L *layers,* $G_{\text{tr}} = (\mathcal{V}_{\text{tr}}, \mathcal{R}, \mathcal{F}_{\text{tr}} \cup \mathcal{T}_{\text{tr}})$ *and* $G_{\text{inf}} = (\mathcal{V}_{\text{inf}}, \mathcal{R}, \mathcal{F}_{\text{inf}} \cup \mathcal{T}_{\text{inf}})$*, if the message, aggregation, update, global-readout, and readout functions of* $f_{\mathbf{w}}$ *are Lipschitz continuous, then* $f_{\mathbf{w}}$ *is Lipschitz continuous with the Lipschitz constant* η_f *and the following holds:*

$$\eta_f \leq \begin{cases} \left(\prod_{l=1}^{L+1} \eta^{(l)}\right) & \theta(k) = k - 1 \\ (L+1)\left(\prod_{l=1}^{L+1} \eta^{(l)}\right) & \theta(k) = 0 \end{cases}$$

$$\begin{aligned} \eta^{(l)} &= \max(A_{\text{upd}}^{(l)} + d_{\max} B_{\text{upd}}^{(l)} A_{\text{agg}}^{(l)} B_{\text{msg}}^{(l)}, B_{\text{upd}}^{(l)} A_{\text{agg}}^{(l)} A_{\text{msg}}^{(l)}, \\ &\qquad |\mathcal{R}|^2 B_{\text{upd}}^{(l)} A_{\text{agg}}^{(l)} C_{\text{msg}}^{(l)}, |\mathcal{R}|^2 B_{\text{upd}}^{(l)} A_{\text{agg}}^{(l)} D_{\text{msg}}^{(l)}, 1), \\ \eta^{(L+1)} &= \max(A_{\text{rd}}, B_{\text{rd}}, C_{\text{rd}} A_{\text{grd}}, \frac{|\mathcal{R}|^2 D_{\text{rd}}}{2 + \max(|\mathcal{V}_{\text{tr}}|, |\mathcal{V}_{\text{inf}}|)}) \end{aligned}$$

where $1 \leq l \leq L$*,* $A_{msg}^{(l)}, B_{msg}^{(l)}, C_{msg}^{(l)}, D_{msg}^{(l)}$ *are the Lipschitz constants of the message function,* $A_{agg}^{(l)}$ *is the Lipschitz constant of the aggregation function,* $A_{upd}^{(l)}, B_{upd}^{(l)}$ *are the Lipschitz constants of the update function,* A_{grd} *is the Lipschitz constant of the global-readout function,* $A_{rd}, B_{rd}, C_{rd}, D_{rd}$ *are the Lipschitz constants of the readout function, and* $d_{\max}$ *is the maximum degree of* G_{tr} *and* G_{inf}*.*

Proof. Without loss of generality, we assume $|\mathcal{V}_{S_1}| \geq |\mathcal{V}_{S_2}|$. Let $\mathcal{X}_{S_1} = \{\{\mathbf{x}_{S_1}^{(L)}(v_1)|v_1 \in \mathcal{V}_{S_1}\}\}$ and $\mathcal{X}_{S_2} = \{\{\mathbf{x}_{S_2}^{(L)}(v_2)|u \in \mathcal{V}_{S_2}\}\} \cup \bigcup_{k=1}^{|\mathcal{V}_{S_1}|-|\mathcal{V}_{S_2}|}\{\{\Phi_{\text{grd}}\}\}$ where $\Phi_{\text{grd}} \in \mathbb{R}^d$ is an embedding vector that makes $\text{GRD}(\mathcal{X}_{S_2})$ same as $\text{GRD}(\{\{\mathbf{x}_{S_2}^{(L)}(v_2)|u \in \mathcal{V}_{S_2}\}\})$. We assume that such an embedding vector exists, and justify this for each variation of *global-readout* function in the Appendix A. Also, we set $\boldsymbol{P}_{\mathcal{X}_{S_1},\mathcal{X}_{S_2}}$ as the optimal transportation plan of $\text{OT}_{\text{RTD}}\left(\rho\left(\{\{(r_{\text{root}}, T_{S_1}^{(L)}(v_1))|v_1 \in \mathcal{V}_{S_1}\}\}, \{\{(r_{\text{root}}, T_{S_2}^{(L)}(v_2))|v_2 \in \mathcal{V}_{S_2}\}\}\right)\right)$. Using the Lipschitz continuity of each function in an SMPNN defined in Definition A.2, we deduce the following inequalities:

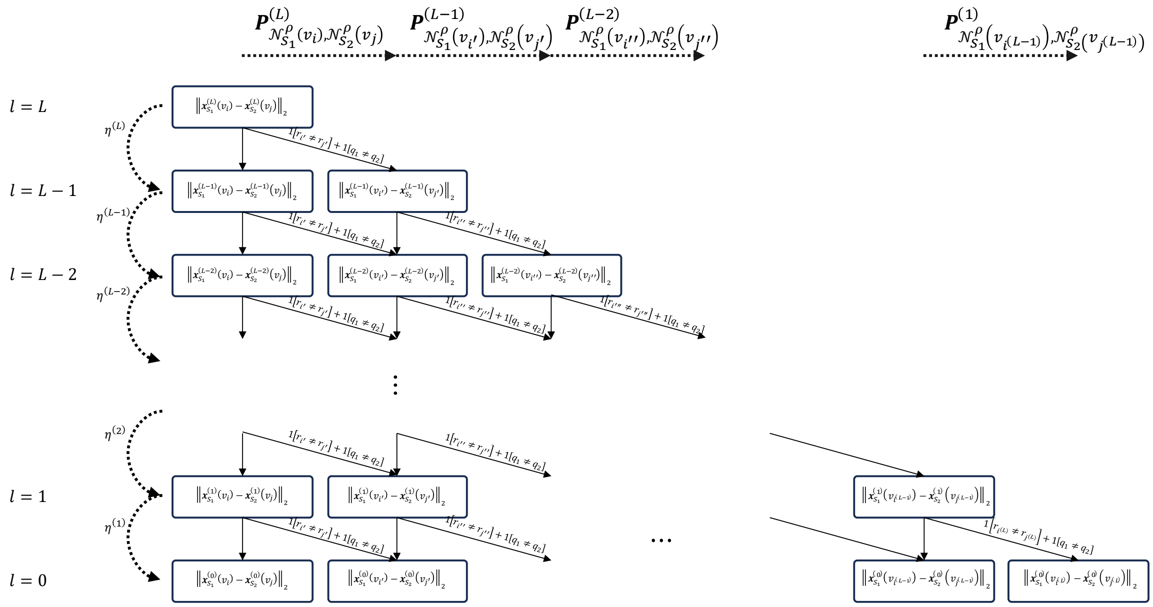
$$
\begin{aligned}
|f_{\mathbf{w}}(S_1) - f_{\mathbf{w}}(S_2)| =& \Big|\text{RD}\left(\mathbf{x}_{S_1}^{(L)}(h_1), \mathbf{x}_{S_1}^{(L)}(t_1), \text{GRD}(\{\{\mathbf{x}_{S_1}^{(L)}(v_1)|v_1 \in \mathcal{V}_{S_1}\}\}), q_1\right) - \\
&\text{RD}\left(\mathbf{x}_{S_2}^{(L)}(h_2), \mathbf{x}_{S_2}^{(L)}(t_2), \text{GRD}(\{\{\mathbf{x}_{S_2}^{(L)}(v_2)|u \in \mathcal{V}_{S_2}\}\}), q_2\right)\Big| \\
\leq& A_{\text{rd}}\|\mathbf{x}_{S_1}^{(L)}(h_1) - \mathbf{x}_{S_2}^{(L)}(h_2)\|_2 + B_{\text{rd}}\|\mathbf{x}_{S_1}^{(L)}(t_1) - \mathbf{x}_{S_2}^{(L)}(t_2)\|_2 + \\
&C_{\text{rd}}\|\text{GRD}(\{\{\mathbf{x}_{S_1}^{(L)}(v_1)|v_1 \in \mathcal{V}_{S_1}\}\}) - \text{GRD}(\{\{\mathbf{x}_{S_2}^{(L)}(v_2)|u \in \mathcal{V}_{S_2}\}\})\|_2 + D_{\text{rd}}\mathbb{1}[q_1 \neq q_2] \\
\leq& A_{\text{rd}}\|\mathbf{x}_{S_1}^{(L)}(h_1) - \mathbf{x}_{S_2}^{(L)}(h_2)\|_2 + B_{\text{rd}}\|\mathbf{x}_{S_1}^{(L)}(t_1) - \mathbf{x}_{S_2}^{(L)}(t_2)\|_2 + D_{\text{rd}}\mathbb{1}[q_1 \neq q_2] + \\
&C_{\text{rd}}A_{\text{grd}}\sum_{i,j}\boldsymbol{P}_{\mathcal{X}_{S_1},\mathcal{X}_{S_2}}[i,j]\|\mathbf{x}_{S_1}^{(L)}(v_i) - \mathbf{x}_{S_2}^{(L)}(v_j)\|_2
\end{aligned}
$$

Similar to blank tree augmentation, pairs of r_{blank} and virtual entities are added to the smaller set among $\mathcal{N}_{S_1}(v_i)$ and $\mathcal{N}_{S_2}(v_j)$, resulting in the augmented multisets $\widetilde{\mathcal{N}}_{S_1}(v_i)$ and $\widetilde{\mathcal{N}}_{S_2}(v_j)$. These augmented multisets are equivalent to the multiset whose elements are pairs of relations and root entities of each subtree in $\rho(\text{SUB}(T_{S_1}^{(l)}(v_i)), \text{SUB}(T_{S_2}^{(l)}(v_j)))$. Note that $\boldsymbol{P}^{(l)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}$ is a transportation plan between $\widetilde{\mathcal{N}}_{S_1}(v_i)$ and $\widetilde{\mathcal{N}}_{S_2}(v_j)$. We set the computed message of the virtual root entity of T_0 and virtual relation r_{blank} as an embedding vector $\Phi_{\text{agg}} \in \mathbb{R}^d$ that makes the aggregated message of the neighbor multiset $\mathcal{N}_S(v)$ same as the aggregated message of the augmented multiset $\widetilde{\mathcal{N}}_S(v)$. We assume that such an embedding vector exists, and justify this for each variation of *aggregation* function in the Appendix A.

$$
\begin{aligned}
&\|\mathbf{x}_{S_1}^{(l)}(v_i) - \mathbf{x}_{S_2}^{(l)}(v_j)\|_2 \\
=&\|\text{UPD}^{(l)}\left(\mathbf{x}_{S_1}^{(\theta(l))}(v_i), \text{AGG}^{(l)}(\{\{\text{MSG}^{(l)}(\mathbf{x}_{S_1}^{(l-1)}(u_{i'}), \mathbf{x}_{S_1}^{(l-1)}(v_i), r_{i'}, q_1)|(r_{i'}, u_{i'}) \in \mathcal{N}_{S_1}(v_i)\}\})\right) - \\
&\text{UPD}^{(l)}\left(\mathbf{x}_{S_2}^{(\theta(l))}(v_j), \text{AGG}^{(l)}(\{\{\text{MSG}^{(l)}(\mathbf{x}_{S_2}^{(l-1)}(u_{j'}), \mathbf{x}_{S_2}^{(l-1)}(v_j), r_{j'}, q_2)|(r_{j'}, u_{j'}) \in \mathcal{N}_{S_2}(v_j)\}\})\right)\|_2 \\
\leq&A_{\text{upd}}^{(l)}\|\mathbf{x}_{S_1}^{(\theta(l))}(v_i) - \mathbf{x}_{S_2}^{(\theta(l))}(v_j)\|_2 + B_{\text{upd}}^{(l)}\|\text{AGG}^{(l)}(\{\{\text{MSG}^{(l)}(\mathbf{x}_{S_1}^{(l-1)}(u_{i'}), \mathbf{x}_{S_1}^{(l-1)}(v_i), r_{i'}, q_1)|(r_{i'}, u_{i'}) \in \mathcal{N}_{S_1}(v_i)\}\}) - \\
&\text{AGG}^{(l)}(\{\{\text{MSG}^{(l)}(\mathbf{x}_{S_2}^{(l-1)}(u_{j'}), \mathbf{x}_{S_2}^{(l-1)}(v_j), r_{j'}, q_2)|(r_{j'}, u_{j'}) \in \mathcal{N}_{S_2}(v_j)\}\})\|_2 \\
=&A_{\text{upd}}^{(l)}\|\mathbf{x}_{S_1}^{(\theta(l))}(v_i) - \mathbf{x}_{S_2}^{(\theta(l))}(v_j)\|_2 + B_{\text{upd}}^{(l)}\|\text{AGG}^{(l)}(\{\{\text{MSG}^{(l)}(\mathbf{x}_{S_1}^{(l-1)}(u_{i'}), \mathbf{x}_{S_1}^{(l-1)}(v_i), r_{i'}, q_1)|(r_{i'}, u_{i'}) \in \widetilde{\mathcal{N}}_{S_1}(v_i)\}\}) - \\
&\text{AGG}^{(l)}(\{\{\text{MSG}^{(l)}(\mathbf{x}_{S_2}^{(l-1)}(u_{j'}), \mathbf{x}_{S_2}^{(l-1)}(v_j), r_{j'}, q_2)|(r_{j'}, u_{j'}) \in \widetilde{\mathcal{N}}_{S_2}(v_j)\}\})\|_2 \\
\leq&A_{\text{upd}}^{(l)}\|\mathbf{x}_{S_1}^{(\theta(l))}(v_i) - \mathbf{x}_{S_2}^{(\theta(l))}(v_j)\|_2 + \\
&B_{\text{upd}}^{(l)}A_{\text{agg}}^{(l)}\sum_{i',j'}\boldsymbol{P}^{(l)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j']\|\text{MSG}^{(l)}(\mathbf{x}_{S_1}^{(l-1)}(u_{i'}), \mathbf{x}_{S_1}^{(l-1)}(v_i), r_{i'}, q_1) - \text{MSG}^{(l)}(\mathbf{x}_{S_2}^{(l-1)}(u_{j'}), \mathbf{x}_{S_2}^{(l-1)}(v_j), r_{j'}, q_2)\|_2 \\
\leq&A_{\text{upd}}^{(l)}\|\mathbf{x}_{S_1}^{(\theta(l))}(v_i) - \mathbf{x}_{S_2}^{(\theta(l))}(v_j)\|_2 + B_{\text{upd}}^{(l)}A_{\text{agg}}^{(l)}\sum_{i',j'}\boldsymbol{P}^{(l)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j']\left(A_{\text{msg}}^{(l)}\|\mathbf{x}_{S_1}^{(l-1)}(u_{i'}) - \mathbf{x}_{S_2}^{(l-1)}(u_{j'})\|_2\right. \\
&\left.+B_{\text{msg}}^{(l)}\|\mathbf{x}_{S_1}^{(l-1)}(v_i) - \mathbf{x}_{S_2}^{(l-1)}(v_j)\|_2 + C_{\text{msg}}^{(l)}\mathbb{1}[r_{i'} \neq r_{j'}] + D_{\text{msg}}^{(l)}\mathbb{1}[q_1 \neq q_2]\right)
\end{aligned}
$$

Let us first consider the case where the *history* function is defined as $\theta(k) = k - 1$. If we define the Lipschitz continuity of each layer of an SMPNN as

$$
\eta^{(l)} = \max(A_{\text{upd}}^{(l)} + d_{\max}B_{\text{upd}}^{(l)}A_{\text{agg}}^{(l)}B_{\text{msg}}^{(l)}, B_{\text{upd}}^{(l)}A_{\text{agg}}^{(l)}A_{\text{msg}}^{(l)}, |\mathcal{R}|^2B_{\text{upd}}^{(l)}A_{\text{agg}}^{(l)}C_{\text{msg}}^{(l)}, |\mathcal{R}|^2B_{\text{upd}}^{(l)}A_{\text{agg}}^{(l)}D_{\text{msg}}^{(l)}, 1), \tag{2}
$$



$P^{(L)}_{\mathcal{N}^{\rho}_{S_1}(v_i),\mathcal{N}^{\rho}_{S_2}(v_j)}$
$P^{(L-1)}_{\mathcal{N}^{\rho}_{S_1}(v_{i'}),\mathcal{N}^{\rho}_{S_2}(v_{j'})}$
$P^{(L-2)}_{\mathcal{N}^{\rho}_{S_1}(v_{i''}),\mathcal{N}^{\rho}_{S_2}(v_{j''})}$
$P^{(1)}_{\mathcal{N}^{\rho}_{S_1}(v_{i^{(L-1)}}),\mathcal{N}^{\rho}_{S_2}(v_{j^{(L-1)}})}$
$l = L$
$l = L-1$
$l = L-2$
$l = 1$
$l = 0$
$\eta^{(L)}$
$\eta^{(L-1)}$
$\eta^{(L-2)}$
$\eta^{(2)}$
$\eta^{(1)}$
$\|x^{(L)}_{S_1}(v_i) - x^{(L)}_{S_2}(v_j)\|_2$
$\|x^{(L-1)}_{S_1}(v_i) - x^{(L-1)}_{S_2}(v_j)\|_2$
$\|x^{(L-1)}_{S_1}(v_{i'}) - x^{(L-1)}_{S_2}(v_{j'})\|_2$
$\|x^{(L-2)}_{S_1}(v_i) - x^{(L-2)}_{S_2}(v_j)\|_2$
$\|x^{(L-2)}_{S_1}(v_{i'}) - x^{(L-2)}_{S_2}(v_{j'})\|_2$
$\|x^{(L-2)}_{S_1}(v_{i''}) - x^{(L-2)}_{S_2}(v_{j''})\|_2$
$\|x^{(1)}_{S_1}(v_i) - x^{(1)}_{S_2}(v_j)\|_2$
$\|x^{(1)}_{S_1}(v_{i'}) - x^{(1)}_{S_2}(v_{j'})\|_2$
$\|x^{(1)}_{S_1}(v_{i^{(L-1)}}) - x^{(1)}_{S_2}(v_{j^{(L-1)}})\|_2$
$\|x^{(0)}_{S_1}(v_i) - x^{(0)}_{S_2}(v_j)\|_2$
$\|x^{(0)}_{S_1}(v_{i'}) - x^{(0)}_{S_2}(v_{j'})\|_2$
$\|x^{(0)}_{S_1}(v_{i^{(L-1)}}) - x^{(0)}_{S_2}(v_{j^{(L-1)}})\|_2$
$\|x^{(0)}_{S_1}(v_{i^{(L)}}) - x^{(0)}_{S_2}(v_{j^{(L)}})\|_2$
$1[r_{i'} \neq r_{j'}] + 1[q_1 \neq q_2]$
$1[r_{i''} \neq r_{j''}] + 1[q_1 \neq q_2]$
$1[r_{i'''} \neq r_{j'''}] + 1[q_1 \neq q_2]$
$1[r_{i^{(L)}} \neq r_{j^{(L)}}] + 1[q_1 \neq q_2]$
⋮
…

Figure 6. Visualization of the iterative process. Each box represents the L_2 norm of the difference between two entity pairs.

$$\eta^{(L+1)} = \max(A_{\text{rd}}, B_{\text{rd}}, C_{\text{rd}}A_{\text{grd}}, \frac{|\mathcal{R}|^2 D_{\text{rd}}}{2+\max(|\mathcal{V}_{\text{tr}}|, |\mathcal{V}_{\text{inf}}|)})$$

, and $\eta^{(0)} = 1$ for $1 \leq l \leq L$, the following inequalities hold.

$$\begin{aligned}
&\|\mathbf{x}^{(l)}_{S_1}(v_i) - \mathbf{x}^{(l)}_{S_2}(v_j)\|_2 \\
\leq & A^{(l)}_{\text{upd}}\|\mathbf{x}^{(l-1)}_{S_1}(v_i) - \mathbf{x}^{(l-1)}_{S_2}(v_j)\|_2 + \\
&B^{(l)}_{\text{upd}}A^{(l)}_{\text{agg}}\sum_{i',j'}\boldsymbol{P}^{(l)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j']\Big(A^{(l)}_{\text{msg}}\|\mathbf{x}^{(l-1)}_{S_1}(u_{i'}) - \mathbf{x}^{(l-1)}_{S_2}(u_{j'})\|_2 + B^{(l)}_{\text{msg}}\|\mathbf{x}^{(l-1)}_{S_1}(v_i) - \mathbf{x}^{(l-1)}_{S_2}(v_j)\|_2! + C^{(l)}_{\text{msg}}\mathbb{1}[r_{i'} \neq r_{j'}] + D^{(l)}_{\text{msg}}\mathbb{1}[q_1 \neq q_2]\Big) \\
\leq & \left(A^{(l)}_{\text{upd}} + d_{\max}B^{(l)}_{\text{upd}}A^{(l)}_{\text{agg}}B^{(l)}_{\text{msg}}\right)\|\mathbf{x}^{(l-1)}_{S_1}(v_i) - \mathbf{x}^{(l-1)}_{S_2}(v_j)\|_2 + \\
&\sum_{i',j'}\boldsymbol{P}^{(l)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j']\left(B^{(l)}_{\text{upd}}A^{(l)}_{\text{agg}}A^{(l)}_{\text{msg}}\|\mathbf{x}^{(l-1)}_{S_1}(u_{i'}) - \mathbf{x}^{(l-1)}_{S_2}(u_{j'})\|_2 + B^{(l)}_{\text{upd}}A^{(l)}_{\text{agg}}C^{(l)}_{\text{msg}}\mathbb{1}[r_{i'} \neq r_{j'}] + B^{(l)}_{\text{upd}}A^{(l)}_{\text{agg}}D^{(l)}_{\text{msg}}\mathbb{1}[q_1 \neq q_2]\right) \\
\leq & \eta^{(l)}\Bigg(\|\mathbf{x}^{(l-1)}_{S_1}(v_i) - \mathbf{x}^{(l-1)}_{S_2}(v_j)\|_2 + \\
&\sum_{i',j'}\boldsymbol{P}^{(l)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j']\left(\|\mathbf{x}^{(l-1)}_{S_1}(u_{i'}) - \mathbf{x}^{(l-1)}_{S_2}(u_{j'})\|_2 + \frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{i'} \neq r_{j'}] + \mathbb{1}[q_1 \neq q_2])\right)\Bigg) \\
\leq & \eta^{(l)}\Bigg(\eta^{(l-1)}\Bigg(\|\mathbf{x}^{(l-2)}_{S_1}(v_i) - \mathbf{x}^{(l-2)}_{S_2}(v_j)\|_2 + \\
&\sum_{i',j'}\boldsymbol{P}^{(l)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j']\left(\|\mathbf{x}^{(l-2)}_{S_1}(u_{i'}) - \mathbf{x}^{(l-2)}_{S_2}(u_{j'})\|_2 + \frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{i'} \neq r_{j'}] + \mathbb{1}[q_1 \neq q_2])\right)\Bigg) + \\
&\sum_{i',j'}\boldsymbol{P}^{(l)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j']\left(\|\mathbf{x}^{(l-1)}_{S_1}(u_{i'}) - \mathbf{x}^{(l-1)}_{S_2}(u_{j'})\|_2 + \frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{i'} \neq r_{j'}] + \mathbb{1}[q_1 \neq q_2])\right)\Bigg)
\end{aligned}$$

The above process shows how a single recursion is unrolled and applied to the next layer. The overall iteration process for L layers is presented in Figure 6. Each box in the figure represents the L_2 norm of the difference between embedding vectors of a specific entity pair at a particular layer. The vertical axis corresponds to layers, and the horizontal axis corresponds to some entity pairs. Hence, boxes within the same column represent the same entity pair across different layers, while boxes in the same row represent the same layer across different entity pairs. From the above inequalities, we can infer that calculating the upper bound of each box requires evaluating the expressions corresponding to the box directly below and the one diagonally below to the right. In Figure 6, a shift to the right column indicates the neighbor sets of the entity pairs in the current column. Thus, the arrows pointing diagonally down to the right require a transportation plan, which must remain consistent for the same entity pair. The process reaches the leaf boxes corresponding to the 0-th layer by recursively applying the iterations described above. The final upper bound for the box at the L-th layer is expressed using the values derived from these leaf boxes. The coefficient for each box is equal to the number of shortest paths from the top box to the corresponding leaf box. Therefore, the coefficients of the leaf boxes correspond to the L-th row of Pascal's triangle. Additionally, the coefficient of the penalties for the relations and queries can be calculated by the summation of Pascal's triangle up to the previous layer in the same column. Therefore, we get the following inequalities.

$$
\begin{aligned}
&\|\mathbf{x}^{(L)}_{S_1}(v_i) - \mathbf{x}^{(L)}_{S_2}(v_j)\|_2 + \frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{\text{root}} \neq r_{\text{root}}] + \mathbb{1}[q_1 \neq q_2]) \\
\leq &\binom{L}{0}\left(\prod_{k=1}^{L}\eta^{(k)}\right)\|\mathbf{x}^{(0)}_{S_1}(v_i) - \mathbf{x}^{(0)}_{S_2}(v_j)\|_2 + \frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{\text{root}} \neq r_{\text{root}}] + \mathbb{1}[q_1 \neq q_2]) + \\
&\sum_{i',j'} \boldsymbol{P}^{(L)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j']\left(\binom{L}{1}\left(\prod_{k=1}^{L}\eta^{(k)}\right)\|\mathbf{x}^{(0)}_{S_1}(u_{i'}) - \mathbf{x}^{(0)}_{S_2}(u_{j'})\|_2 + \frac{1}{|\mathcal{R}|^2}\left(\sum_{r=1}^{L}\binom{L-r}{0}\prod_{k=r}^{L}\eta^{(k)}\right)(\mathbb{1}[r_{i'} \neq r_{j'}] + \mathbb{1}[q_1 \neq q_2]) + \right. \\
&\sum_{i'',j''} \boldsymbol{P}^{(L-1)}_{\widetilde{\mathcal{N}}_{S_1}(v_{i'}),\widetilde{\mathcal{N}}_{S_2}(v_{j'})}[i'',j'']\left(\binom{L}{2}\left(\prod_{k=1}^{L}\eta^{(k)}\right)\|\mathbf{x}^{(0)}_{S_1}(u_{i''}) - \mathbf{x}^{(0)}_{S_2}(u_{j''})\|_2 + \frac{1}{|\mathcal{R}|^2}\left(\sum_{r=1}^{L-1}\binom{L-r}{1}\prod_{k=r}^{L}\eta^{(k)}\right)(\mathbb{1}[r_{i''} \neq r_{j''}] + \mathbb{1}[q_1 \neq q_2]) + \right. \\
&\dots \\
&\sum_{i^{(L-1)},j^{(L-1)}} \boldsymbol{P}^{(2)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-2)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-2)}}\right)}[i^{(L-1)},j^{(L-1)}]\left(\binom{L}{L-1}\left(\prod_{k=1}^{L}\eta^{(k)}\right)\|\mathbf{x}^{(0)}_{S_1}(u_{i^{(L-1)}}) - \mathbf{x}^{(0)}_{S_2}(u_{j^{(L-1)}})\|_2\right. \\
&+ \frac{1}{|\mathcal{R}|^2}\left(\sum_{r=1}^{2}\binom{L-r}{L-2}\prod_{k=r}^{L}\eta^{(k)}\right)\left(\mathbb{1}[r_{i^{(L-1)}} \neq r_{j^{(L-1)}}] + \mathbb{1}[q_1 \neq q_2]\right) + \\
&\sum_{i^{(L)},j^{(L)}} \boldsymbol{P}^{(1)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-1)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-1)}}\right)}[i^{(L)},j^{(L)}]\left(\binom{L}{L}\left(\prod_{k=1}^{L}\eta^{(k)}\right)\|\mathbf{x}^{(0)}_{S_1}(u_{i^{(L)}}) - \mathbf{x}^{(0)}_{S_2}(u_{j^{(L)}})\|_2\right. \\
&\left.\left.\left.+\frac{1}{|\mathcal{R}|^2}\left(\sum_{r=1}^{1}\binom{L-r}{L-1}\prod_{k=r}^{L}\eta^{(k)}\right)\left(\mathbb{1}[r_{i^{(L)}} \neq r_{j^{(L)}}] + \mathbb{1}[q_1 \neq q_2]\right)\right)\right)\dots\right)\right) \\
\leq &\left(\prod_{k=1}^{L}\eta^{(k)}\right)\|\mathbf{x}^{(0)}_{S_1}(v_i) - \mathbf{x}^{(0)}_{S_2}(v_j)\|_2 + \frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{\text{root}} \neq r_{\text{root}}] + \mathbb{1}[q_1 \neq q_2]) + \\
&\frac{\binom{L}{1}}{\binom{L}{0}}\sum_{i',j'} \boldsymbol{P}^{(L)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j']\left(\left(\prod_{k=1}^{L}\eta^{(k)}\right)\|\mathbf{x}^{(0)}_{S_1}(u_{i'}) - \mathbf{x}^{(0)}_{S_2}(u_{j'})\|_2 + \frac{1}{|\mathcal{R}|^2}\left(\sum_{r=1}^{L}\frac{\binom{L-r}{0}}{\binom{L}{1}}\prod_{k=r}^{L}\eta^{(k)}\right)(\mathbb{1}[r_{i'} \neq r_{j'}] + \mathbb{1}[q_1 \neq q_2]) + \right. \\
&\frac{\binom{L}{2}}{\binom{L}{1}}\sum_{i'',j''} \boldsymbol{P}^{(L-1)}_{\widetilde{\mathcal{N}}_{S_1}(v_{i'}),\widetilde{\mathcal{N}}_{S_2}(v_{j'})}[i'',j'']\left(\left(\prod_{k=1}^{L}\eta^{(k)}\right)\|\mathbf{x}^{(0)}_{S_1}(u_{i''}) - \mathbf{x}^{(0)}_{S_2}(u_{j''})\|_2 + \frac{1}{|\mathcal{R}|^2}\left(\sum_{r=1}^{L-1}\frac{\binom{L-r}{1}}{\binom{L}{2}}\prod_{k=r}^{L}\eta^{(k)}\right)(\mathbb{1}[r_{i''} \neq r_{j''}] + \mathbb{1}[q_1 \neq q_2]) + \right. \\
&\dots \\
&\frac{\binom{L}{L-1}}{\binom{L}{L-2}}\sum_{i^{(L-1)},j^{(L-1)}} \boldsymbol{P}^{(2)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-2)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-2)}}\right)}[i^{(L-1)},j^{(L-1)}]\left(\left(\prod_{k=1}^{L}\eta^{(k)}\right)\|\mathbf{x}^{(0)}_{S_1}(u_{i^{(L-1)}}) - \mathbf{x}^{(0)}_{S_2}(u_{j^{(L-1)}})\|_2\right. \\
&+ \frac{1}{|\mathcal{R}|^2}\left(\sum_{r=1}^{2}\frac{\binom{L-r}{L-2}}{\binom{L}{L-1}}\prod_{k=r}^{L}\eta^{(k)}\right)\left(\mathbb{1}[r_{i^{(L-1)}} \neq r_{j^{(L-1)}}] + \mathbb{1}[q_1 \neq q_2]\right) + \\
&\frac{\binom{L}{L}}{\binom{L}{L-1}}\sum_{i^{(L)},j^{(L)}} \boldsymbol{P}^{(1)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-1)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-1)}}\right)}[i^{(L)},j^{(L)}]\left(\left(\prod_{k=1}^{L}\eta^{(k)}\right)\|\mathbf{x}^{(0)}_{S_1}(u_{i^{(L)}}) - \mathbf{x}^{(0)}_{S_2}(u_{j^{(L)}})\|_2\right. \\
&\left.\left.\left.+\frac{1}{|\mathcal{R}|^2}\left(\sum_{r=1}^{1}\frac{\binom{L-r}{L-1}}{\binom{L}{L}}\prod_{k=r}^{L}\eta^{(k)}\right)\left(\mathbb{1}[r_{i^{(L)}} \neq r_{j^{(L)}}] + \mathbb{1}[q_1 \neq q_2]\right)\right)\right)\dots\right)\right) \\
\leq &\left(\prod_{k=0}^{L}\eta^{(k)}\right)\left(\|\mathbf{x}^{(0)}_{S_1}(v_i) - \mathbf{x}^{(0)}_{S_2}(v_j)\|_2 + \frac{1}{|\mathcal{R}|^2}\left(\prod_{k=0}^{L}\frac{1}{\eta^{(k)}}\right)(\mathbb{1}[r_{\text{root}} \neq r_{\text{root}}] + \mathbb{1}[q_1 \neq q_2]) + \right. \\
&\frac{\binom{L}{1}}{\binom{L}{0}}\sum_{i',j'} \boldsymbol{P}^{(L)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j']\left(\|\mathbf{x}^{(0)}_{S_1}(u_{i'}) - \mathbf{x}^{(0)}_{S_2}(u_{j'})\|_2 + \frac{1}{|\mathcal{R}|^2}\left(\sum_{r=1}^{L}\frac{\binom{L-r}{0}}{\binom{L}{1}}\prod_{k=0}^{r-1}\frac{1}{\eta^{(k)}}\right)(\mathbb{1}[r_{i'} \neq r_{j'}] + \mathbb{1}[q_1 \neq q_2]) + \right. \\
&\frac{\binom{L}{2}}{\binom{L}{1}}\sum_{i'',j''} \boldsymbol{P}^{(L-1)}_{\widetilde{\mathcal{N}}_{S_1}(v_{i'}),\widetilde{\mathcal{N}}_{S_2}(v_{j'})}[i'',j'']\left(\|\mathbf{x}^{(0)}_{S_1}(u_{i''}) - \mathbf{x}^{(0)}_{S_2}(u_{j''})\|_2 + \frac{1}{|\mathcal{R}|^2}\left(\sum_{r=1}^{L-1}\frac{\binom{L-r}{1}}{\binom{L}{2}}\prod_{k=0}^{r-1}\frac{1}{\eta^{(k)}}\right)(\mathbb{1}[r_{i''} \neq r_{j''}] + \mathbb{1}[q_1 \neq q_2]) + \right. \\
&\dots
\end{aligned}
$$

$$
\begin{aligned}
&\frac{\binom{L}{L-1}}{\binom{L}{L-2}} \sum_{i^{(L-1)},j^{(L-1)}} \boldsymbol{P}^{(2)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-2)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-2)}}\right)}[i^{(L-1)},j^{(L-1)}] \Big(\|\mathbf{x}^{(0)}_{S_1}(u_{i^{(L-1)}}) - \mathbf{x}^{(0)}_{S_2}(u_{j^{(L-1)}})\|_2 \\
&\quad + \frac{1}{|\mathcal{R}|^2} \left(\sum_{r=1}^{2} \frac{\binom{L-r}{L-2}}{\binom{L}{L-1}} \prod_{k=0}^{r-1} \frac{1}{\eta^{(k)}} \right) \Big(\mathbb{1}[r_{i^{(L-1)}} \neq r_{j^{(L-1)}}] + \mathbb{1}[q_1 \neq q_2] \Big) + \\
&\frac{\binom{L}{L}}{\binom{L}{L-1}} \sum_{i^{(L)},j^{(L)}} \boldsymbol{P}^{(1)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-1)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-1)}}\right)}[i^{(L)},j^{(L)}] \Big(\|\mathbf{x}^{(0)}_{S_1}(u_{i^{(L)}}) - \mathbf{x}^{(0)}_{S_2}(u_{j^{(L)}})\|_2 \\
&\quad + \frac{1}{|\mathcal{R}|^2} \left(\sum_{r=1}^{1} \frac{\binom{L-r}{L-1}}{\binom{L}{L}} \prod_{k=0}^{r-1} \frac{1}{\eta^{(k)}} \right) \Big(\mathbb{1}[r_{i^{(L)}} \neq r_{j^{(L)}}] + \mathbb{1}[q_1 \neq q_2] \Big) \Big) \Big) \dots \Big) \Big) \Big) \\
\leq & \left(\prod_{k=0}^{L} \eta^{(k)} \right) \Bigg(\|\mathbf{x}^{(0)}_{S_1}(v_i) - \mathbf{x}^{(0)}_{S_2}(v_j)\|_2 + \frac{1}{|\mathcal{R}|^2} (\mathbb{1}[r_{\text{root}} \neq r_{\text{root}}] + \mathbb{1}[q_1 \neq q_2]) + \\
&\frac{\binom{L}{1}}{\binom{L}{0}} \sum_{i',j'} \boldsymbol{P}^{(L)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j'] \Bigg(\|\mathbf{x}^{(0)}_{S_1}(u_{i'}) - \mathbf{x}^{(0)}_{S_2}(u_{j'})\|_2 + \frac{1}{|\mathcal{R}|^2} \left(\sum_{r=1}^{L} \frac{\binom{L-r}{0}}{\binom{L}{1}} \prod_{k=0}^{r-1} 1 \right) (\mathbb{1}[r_{i'} \neq r_{j'}] + \mathbb{1}[q_1 \neq q_2]) + \\
&\frac{\binom{L}{2}}{\binom{L}{1}} \sum_{i'',j''} \boldsymbol{P}^{(L-1)}_{\widetilde{\mathcal{N}}_{S_1}(v_{i'}),\widetilde{\mathcal{N}}_{S_2}(v_{j'})}[i'',j''] \Bigg(\|\mathbf{x}^{(0)}_{S_1}(u_{i''}) - \mathbf{x}^{(0)}_{S_2}(u_{j''})\|_2 + \frac{1}{|\mathcal{R}|^2} \left(\sum_{r=1}^{L-1} \frac{\binom{L-r}{1}}{\binom{L}{2}} \prod_{k=0}^{r-1} 1 \right) (\mathbb{1}[r_{i''} \neq r_{j''}] + \mathbb{1}[q_1 \neq q_2]) + \\
&\dots \\
&\frac{\binom{L}{L-1}}{\binom{L}{L-2}} \sum_{i^{(L-1)},j^{(L-1)}} \boldsymbol{P}^{(2)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-2)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-2)}}\right)}[i^{(L-1)},j^{(L-1)}] \Big(\|\mathbf{x}^{(0)}_{S_1}(u_{i^{(L-1)}}) - \mathbf{x}^{(0)}_{S_2}(u_{j^{(L-1)}})\|_2 \\
&\quad + \frac{1}{|\mathcal{R}|^2} \left(\sum_{r=1}^{2} \frac{\binom{L-r}{L-2}}{\binom{L}{L-1}} \prod_{k=0}^{r-1} 1 \right) \Big(\mathbb{1}[r_{i^{(L-1)}} \neq r_{j^{(L-1)}}] + \mathbb{1}[q_1 \neq q_2] \Big) + \\
&\frac{\binom{L}{L}}{\binom{L}{L-1}} \sum_{i^{(L)},j^{(L)}} \boldsymbol{P}^{(1)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-1)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-1)}}\right)}[i^{(L)},j^{(L)}] \Big(\|\mathbf{x}^{(0)}_{S_1}(u_{i^{(L)}}) - \mathbf{x}^{(0)}_{S_2}(u_{j^{(L)}})\|_2 \\
&\quad + \frac{1}{|\mathcal{R}|^2} \left(\sum_{r=1}^{1} \frac{\binom{L-r}{L-1}}{\binom{L}{L}} \prod_{k=0}^{r-1} 1 \right) \Big(\mathbb{1}[r_{i^{(L)}} \neq r_{j^{(L)}}] + \mathbb{1}[q_1 \neq q_2] \Big) \Big) \Big) \dots \Big) \Big) \Big) \\
\leq & \left(\prod_{k=0}^{L} \eta^{(k)} \right) \Bigg(\|\mathbf{x}^{(0)}_{S_1}(v_i) - \mathbf{x}^{(0)}_{S_2}(v_j)\|_2 + \frac{1}{|\mathcal{R}|^2} (\mathbb{1}[r_{\text{root}} \neq r_{\text{root}}] + \mathbb{1}[q_1 \neq q_2]) + \\
&\frac{\binom{L}{1}}{\binom{L}{0}} \sum_{i',j'} \boldsymbol{P}^{(L)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j'] \Big(\|\mathbf{x}^{(0)}_{S_1}(u_{i'}) - \mathbf{x}^{(0)}_{S_2}(u_{j'})\|_2 + \frac{1}{|\mathcal{R}|^2} (\mathbb{1}[r_{i'} \neq r_{j'}] + \mathbb{1}[q_1 \neq q_2]) + \\
&\frac{\binom{L}{2}}{\binom{L}{1}} \sum_{i'',j''} \boldsymbol{P}^{(L-1)}_{\widetilde{\mathcal{N}}_{S_1}(v_{i'}),\widetilde{\mathcal{N}}_{S_2}(v_{j'})}[i'',j''] \Big(\|\mathbf{x}^{(0)}_{S_1}(u_{i''}) - \mathbf{x}^{(0)}_{S_2}(u_{j''})\|_2 + \frac{1}{|\mathcal{R}|^2} (\mathbb{1}[r_{i''} \neq r_{j''}] + \mathbb{1}[q_1 \neq q_2]) + \\
&\dots \\
&\quad \frac{\binom{L}{L}}{\binom{L}{L-1}} \sum_{i^{(L)},j^{(L)}} \boldsymbol{P}^{(1)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-1)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-1)}}\right)}[i^{(L)},j^{(L)}] \Big(\|\mathbf{x}^{(0)}_{S_1}(u_{i^{(L)}}) - \mathbf{x}^{(0)}_{S_2}(u_{j^{(L)}})\|_2 + \frac{1}{|\mathcal{R}|^2} (\mathbb{1}[r_{i^{(L)}} \neq r_{j^{(L)}}] + \mathbb{1}[q_1 \neq q_2]) \Big) \dots \Big) \Big) \Big) \Big)
\end{aligned}
$$

If we set each transportation plan as the optimal transportation plan of the RTD$((r_{\text{root}}, T^{(L)}_{S_1}(v_i)), (r_{\text{root}}, T^{(L)}_{S_2}(v_j)))$ with the weight function $w(l) = \frac{\binom{L}{L-l+1}}{\binom{L}{L-l}}$, we can bound the above equation as follows.

$$
\|\mathbf{x}^{(L)}_{S_1}(v_i) - \mathbf{x}^{(L)}_{S_2}(v_j)\|_2 + \frac{1}{|\mathcal{R}|^2} (\mathbb{1}[r_{\text{root}} \neq r_{\text{root}}] + \mathbb{1}[q_1 \neq q_2]) \leq \left(\prod_{l=0}^{L} \eta^{(l)} \right) \text{RTD}((r_{\text{root}}, T^{(L)}_{S_1}(v_i)), (r_{\text{root}}, T^{(L)}_{S_2}(v_j)))
$$

Finally if we set $\boldsymbol{P}^{(l)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}$ as the optimal transportation plan of the $\text{OT}_{\text{RTD}}(S_1, S_2)$, we can get following equations.

$$
\begin{aligned}
&|f_{\mathbf{w}}(S_1) - f_{\mathbf{w}}(S_2)| \\
\leq & A_{\text{rd}} \|\mathbf{x}^{(L)}_{S_1}(h_1) - \mathbf{x}^{(L)}_{S_2}(h_2)\|_2 + B_{\text{rd}} \|\mathbf{x}^{(L)}_{S_1}(t_1) - \mathbf{x}^{(L)}_{S_2}(t_2)\|_2 + D_{\text{rd}} \mathbb{1}[q_1 \neq q_2] + \\
& C_{\text{rd}} A_{\text{grd}} \sum_{i,j} \boldsymbol{P}_{\mathcal{X}_{S_1},\mathcal{X}_{S_2}}[i,j] \|\mathbf{x}^{(L)}_{S_1}(v_i) - \mathbf{x}^{(L)}_{S_2}(v_j)\|_2 \\
\leq & \eta^{(L+1)} \Bigg(\|\mathbf{x}^{(L)}_{S_1}(h_1) - \mathbf{x}^{(L)}_{S_2}(h_2)\|_2 + \|\mathbf{x}^{(L)}_{S_1}(t_1) - \mathbf{x}^{(L)}_{S_2}(t_2)\|_2 + \frac{2 + \max(|\mathcal{V}_{S_1}|, |\mathcal{V}_{S_2}|)}{|\mathcal{R}|^2} \mathbb{1}[q_1 \neq q_2]
\end{aligned}
$$

$$
\begin{aligned}
&\quad + \sum_{i,j} \boldsymbol{P}_{\mathcal{X}_{S_1},\mathcal{X}_{S_2}}[i,j] \|\mathbf{x}_{S_1}^{(L)}(v_i) - \mathbf{x}_{S_2}^{(L)}(v_j)\|_2 \Bigg) \\
&\leq \eta^{(L+1)} \Bigg(\|\mathbf{x}_{S_1}^{(L)}(h_1) - \mathbf{x}_{S_2}^{(L)}(h_2)\|_2 + \frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{\text{root}} \neq r_{\text{root}}] + \mathbb{1}[q_1 \neq q_2]) + \\
&\quad \|\mathbf{x}_{S_1}^{(L)}(t_1) - \mathbf{x}_{S_2}^{(L)}(t_2)\|_2 + \frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{\text{root}} \neq r_{\text{root}}] + \mathbb{1}[q_1 \neq q_2]) + \\
&\quad \sum_{i,j} \boldsymbol{P}_{\mathcal{X}_{S_1},\mathcal{X}_{S_2}}[i,j] \left(\|\mathbf{x}_{S_1}^{(L)}(v_i) - \mathbf{x}_{S_2}^{(L)}(v_j)\|_2 + \frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{\text{root}} \neq r_{\text{root}}] + \mathbb{1}[q_1 \neq q_2]) \right) \Bigg) \\
&\leq \left(\prod_{l=1}^{L+1} \eta^{(k)} \right) \Big(\text{RTD}((r_{\text{root}}, T_{S_1}^{(L)}(h_1)), (r_{\text{root}}, T_{S_2}^{(L)}(h_2))) + \text{RTD}((r_{\text{root}}, T_{S_1}^{(L)}(t_1)), (r_{\text{root}}, T_{S_2}^{(L)}(t_2))) + \text{OT}_{\text{RTD}}(S_1, S_2) \Big) \\
&\leq \left(\prod_{l=1}^{L+1} \eta^{(k)} \right) \text{RTMD}(S_1, S_2)
\end{aligned}
$$

Next, we consider the case where the *history* function is defined as $\theta(k) = 0$. Using the Lipschitz continuity of each function in an SMPNN defined in Definition A.2, we deduce the following inequalities:

$$
\begin{aligned}
&\|\mathbf{x}_{S_1}^{(l)}(v_i) - \mathbf{x}_{S_2}^{(l)}(v_j)\|_2 \\
\leq & A_{\text{upd}}^{(l)} \|\mathbf{x}_{S_1}^{(0)}(v_i) - \mathbf{x}_{S_2}^{(0)}(v_j)\|_2 + \\
& B_{\text{upd}}^{(l)} A_{\text{agg}}^{(l)} \sum_{i',j'} \boldsymbol{P}^{(l)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j'] \Big(A_{\text{msg}}^{(l)} \|\mathbf{x}_{S_1}^{(l-1)}(u_{i'}) - \mathbf{x}_{S_2}^{(l-1)}(u_{j'})\|_2 + B_{\text{msg}}^{(l)} \|\mathbf{x}_{S_1}^{(l-1)}(v_i) - \mathbf{x}_{S_2}^{(l-1)}(v_j)\|_2 + C_{\text{msg}}^{(l)} \mathbb{1}[r_{i'} \neq r_{j'}] + D_{\text{msg}}^{(l)} \mathbb{1}[q_1 \neq q_2] \Big) \\
\leq & A_{\text{upd}}^{(l)} \|\mathbf{x}_{S_1}^{(0)}(v_i) - \mathbf{x}_{S_2}^{(0)}(v_j)\|_2 + d_{\max} B_{\text{upd}}^{(l)} A_{\text{agg}}^{(l)} B_{\text{msg}}^{(l)} \|\mathbf{x}_{S_1}^{(l-1)}(v_i) - \mathbf{x}_{S_2}^{(l-1)}(v_j)\|_2 + \\
& \sum_{i',j'} \boldsymbol{P}^{(l)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j'] \Big(B_{\text{upd}}^{(l)} A_{\text{agg}}^{(l)} A_{\text{msg}}^{(l)} \|\mathbf{x}_{S_1}^{(l-1)}(u_{i'}) - \mathbf{x}_{S_2}^{(l-1)}(u_{j'})\|_2 + B_{\text{upd}}^{(l)} A_{\text{agg}}^{(l)} C_{\text{msg}}^{(l)} \mathbb{1}[r_{i'} \neq r_{j'}] + B_{\text{upd}}^{(l)} A_{\text{agg}}^{(l)} D_{\text{msg}}^{(l)} \mathbb{1}[q_1 \neq q_2] \Big) \\
\leq & \eta^{(l)} \Bigg(\|\mathbf{x}_{S_1}^{(0)}(v_i) - \mathbf{x}_{S_2}^{(0)}(v_j)\|_2 + \|\mathbf{x}_{S_1}^{(l-1)}(v_i) - \mathbf{x}_{S_2}^{(l-1)}(v_j)\|_2 + \\
& \sum_{i',j'} \boldsymbol{P}^{(l)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j'] \left(\|\mathbf{x}_{S_1}^{(l-1)}(u_{i'}) - \mathbf{x}_{S_2}^{(l-1)}(u_{j'})\|_2 + \frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{i'} \neq r_{j'}] + \mathbb{1}[q_1 \neq q_2]) \right) \Bigg)
\end{aligned}
$$

Using the same approach as for the case where $\theta(k) = k - 1$, we can derive the following equation. The only difference is in calculating the coefficients for the values corresponding to the leaf boxes. In this case, the coefficient of each leaf box is equal to the sum of paths from all boxes in that column to the leaf box.

$$
\begin{aligned}
&\|\mathbf{x}_{S_1}^{(L)}(v_i) - \mathbf{x}_{S_2}^{(L)}(v_j)\|_2 + \frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{\text{root}} \neq r_{\text{root}}] + \mathbb{1}[q_1 \neq q_2]) \\
\leq & \left(\binom{L}{0} \left(\prod_{k=1}^{L} \eta^{(k)} \right) + \sum_{r=1}^{L} \binom{L-r}{0} \prod_{k=r}^{L} \eta^{(k)} \right) \|\mathbf{x}_{S_1}^{(0)}(v_i) - \mathbf{x}_{S_2}^{(0)}(v_j)\|_2 + \frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{\text{root}} \neq r_{\text{root}}] + \mathbb{1}[q_1 \neq q_2]) + \\
& \sum_{i',j'} \boldsymbol{P}^{(L)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j'] \Bigg(\Bigg(\left(\binom{L}{1} \left(\prod_{k=1}^{L} \eta^{(k)} \right) + \sum_{r=1}^{L-1} \binom{L-r}{1} \prod_{k=r}^{L} \eta^{(k)} \right) \|\mathbf{x}_{S_1}^{(0)}(u_{i'}) - \mathbf{x}_{S_2}^{(0)}(u_{j'})\|_2 \\
& + \frac{1}{|\mathcal{R}|^2} \left(\sum_{r=1}^{L} \binom{L-r}{0} \prod_{k=r}^{L} \eta^{(k)} \right) (\mathbb{1}[r_{i'} \neq r_{j'}] + \mathbb{1}[q_1 \neq q_2]) + \\
& \sum_{i'',j''} \boldsymbol{P}^{(L-1)}_{\widetilde{\mathcal{N}}_{S_1}(v_{i'}),\widetilde{\mathcal{N}}_{S_2}(v_{j'})}[i'',j''] \Bigg(\Bigg(\left(\binom{L}{2} \left(\prod_{k=1}^{L} \eta^{(k)} \right) + \sum_{r=1}^{L-2} \binom{L-r}{2} \prod_{k=r}^{L} \eta^{(k)} \right) \|\mathbf{x}_{S_1}^{(0)}(u_{i''}) - \mathbf{x}_{S_2}^{(0)}(u_{j''})\|_2 \\
& + \frac{1}{|\mathcal{R}|^2} \left(\sum_{r=1}^{L-1} \binom{L-r}{1} \prod_{k=r}^{L} \eta^{(k)} \right) (\mathbb{1}[r_{i''} \neq r_{j''}] + \mathbb{1}[q_1 \neq q_2]) + \\
& \cdots \\
& \sum_{i^{(L-1)},j^{(L-1)}} \boldsymbol{P}^{(2)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-2)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-2)}}\right)}[i^{(L-1)},j^{(L-1)}] \Bigg(\Bigg(\left(\binom{L}{L-1} \left(\prod_{k=1}^{L} \eta^{(k)} \right) + \sum_{r=1}^{1} \binom{L-r}{L-1} \prod_{k=r}^{L} \eta^{(k)} \right) \|\mathbf{x}_{S_1}^{(0)}(u_{i^{(L-1)}}) - \mathbf{x}_{S_2}^{(0)}(u_{j^{(L-1)}})\|_2 \\
& + \frac{1}{|\mathcal{R}|^2} \left(\sum_{r=1}^{2} \binom{L-r}{L-2} \prod_{k=r}^{L} \eta^{(k)} \right) \Big(\mathbb{1}[r_{i^{(L-1)}} \neq r_{j^{(L-1)}}] + \mathbb{1}[q_1 \neq q_2] \Big) +
\end{aligned}
$$

$$\sum_{i^{(L)},j^{(L)}} \boldsymbol{P}^{(1)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-1)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-1)}}\right)}[i^{(L)},j^{(L)}] \Bigg(\binom{L}{L} \left(\prod_{k=1}^{L} \eta^{(k)}\right) \|\mathbf{x}^{(0)}_{S_1}(u_{i^{(L)}}) - \mathbf{x}^{(0)}_{S_2}(u_{j^{(L)}})\|_2$$

$$+\frac{1}{|\mathcal{R}|^2} \left(\sum_{r=1}^{1} \binom{L-r}{L-1} \prod_{k=r}^{L} \eta^{(k)}\right) \Big(\mathbb{1}[r_{i^{(L)}} \neq r_{j^{(L)}}] + \mathbb{1}[q_1 \neq q_2]\Big)\Bigg)\Bigg) \ldots \Bigg)\Bigg)$$

$$\leq \left(\prod_{k=1}^{L} \eta^{(k)}\right) \Bigg(\Bigg(\Bigg(\binom{L}{0} + \sum_{r=1}^{L} \binom{L-r}{0} \prod_{k=0}^{r-1} \frac{1}{\eta^{(k)}}\Bigg) \|\mathbf{x}^{(0)}_{S_1}(v_i) - \mathbf{x}^{(0)}_{S_2}(v_j)\|_2 + \frac{1}{|\mathcal{R}|^2} \left(\prod_{k=0}^{L} \frac{1}{\eta^{(k)}}\right) (\mathbb{1}[r_{\text{root}} \neq r_{\text{root}}] + \mathbb{1}[q_1 \neq q_2]) +$$

$$\sum_{i',j'} \boldsymbol{P}^{(L)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j'] \Bigg(\Bigg(\Bigg(\binom{L}{1} + \sum_{r=1}^{L-1} \binom{L-r}{1} \prod_{k=0}^{r-1} \frac{1}{\eta^{(k)}}\Bigg) \|\mathbf{x}^{(0)}_{S_1}(u_{i'}) - \mathbf{x}^{(0)}_{S_2}(u_{j'})\|_2$$

$$+ \frac{1}{|\mathcal{R}|^2} \left(\sum_{r=1}^{L} \binom{L-r}{0} \prod_{k=0}^{r-1} \frac{1}{\eta^{(k)}}\right) (\mathbb{1}[r_{i'} \neq r_{j'}] + \mathbb{1}[q_1 \neq q_2]) +$$

$$\sum_{i'',j''} \boldsymbol{P}^{(L-1)}_{\widetilde{\mathcal{N}}_{S_1}(v_{i'}),\widetilde{\mathcal{N}}_{S_2}(v_{j'})}[i'',j''] \Bigg(\Bigg(\Bigg(\binom{L}{2} + \sum_{r=1}^{L-2} \binom{L-r}{2} \prod_{k=0}^{r-1} \frac{1}{\eta^{(k)}}\Bigg) \|\mathbf{x}^{(0)}_{S_1}(u_{i''}) - \mathbf{x}^{(0)}_{S_2}(u_{j''})\|_2$$

$$+ \frac{1}{|\mathcal{R}|^2} \left(\sum_{r=1}^{L-1} \binom{L-r}{1} \prod_{k=0}^{r-1} \frac{1}{\eta^{(k)}}\right) (\mathbb{1}[r_{i''} \neq r_{j''}] + \mathbb{1}[q_1 \neq q_2]) +$$

$$\ldots$$

$$\sum_{i^{(L-1)},j^{(L-1)}} \boldsymbol{P}^{(2)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-2)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-2)}}\right)}[i^{(L-1)},j^{(L-1)}] \Bigg(\Bigg(\Bigg(\binom{L}{L-1} + \sum_{r=1}^{1} \binom{L-r}{L-1} \prod_{k=0}^{r-1} \frac{1}{\eta^{(k)}}\Bigg) \|\mathbf{x}^{(0)}_{S_1}(u_{i^{(L-1)}}) - \mathbf{x}^{(0)}_{S_2}(u_{j^{(L-1)}})\|_2$$

$$+ \frac{1}{|\mathcal{R}|^2} \left(\sum_{r=1}^{2} \binom{L-r}{L-2} \prod_{k=0}^{r-1} \frac{1}{\eta^{(k)}}\right) \Big(\mathbb{1}[r_{i^{(L-1)}} \neq r_{j^{(L-1)}}] + \mathbb{1}[q_1 \neq q_2]\Big) +$$

$$\sum_{i^{(L)},j^{(L)}} \boldsymbol{P}^{(1)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-1)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-1)}}\right)}[i^{(L)},j^{(L)}] \Bigg(\binom{L}{L} \|\mathbf{x}^{(0)}_{S_1}(u_{i^{(L)}}) - \mathbf{x}^{(0)}_{S_2}(u_{j^{(L)}})\|_2$$

$$+\frac{1}{|\mathcal{R}|^2} \left(\sum_{r=1}^{1} \binom{L-r}{L-1} \prod_{k=0}^{r-1} \frac{1}{\eta^{(k)}}\right) \Big(\mathbb{1}[r_{i^{(L)}} \neq r_{j^{(L)}}] + \mathbb{1}[q_1 \neq q_2]\Big)\Bigg)\Bigg) \ldots \Bigg)\Bigg)\Bigg)$$

$$\leq \left(\prod_{k=1}^{L} \eta^{(k)}\right) \Bigg(\Bigg(\Bigg(\binom{L}{0} + \sum_{r=1}^{L} \binom{L-r}{0}\Bigg) \|\mathbf{x}^{(0)}_{S_1}(v_i) - \mathbf{x}^{(0)}_{S_2}(v_j)\|_2 + \frac{1}{|\mathcal{R}|^2} (\mathbb{1}[r_{\text{root}} \neq r_{\text{root}}] + \mathbb{1}[q_1 \neq q_2]) +$$

$$\sum_{i',j'} \boldsymbol{P}^{(L)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j'] \Bigg(\Bigg(\Bigg(\binom{L}{1} + \sum_{r=1}^{L-1} \binom{L-r}{1}\Bigg) \|\mathbf{x}^{(0)}_{S_1}(u_{i'}) - \mathbf{x}^{(0)}_{S_2}(u_{j'})\|_2 + \frac{1}{|\mathcal{R}|^2} \left(\sum_{r=1}^{L} \binom{L-r}{0}\right) (\mathbb{1}[r_{i'} \neq r_{j'}] + \mathbb{1}[q_1 \neq q_2]) +$$

$$\sum_{i'',j''} \boldsymbol{P}^{(L-1)}_{\widetilde{\mathcal{N}}_{S_1}(v_{i'}),\widetilde{\mathcal{N}}_{S_2}(v_{j'})}[i'',j''] \Bigg(\Bigg(\Bigg(\binom{L}{2} + \sum_{r=1}^{L-2} \binom{L-r}{2}\Bigg) \|\mathbf{x}^{(0)}_{S_1}(u_{i''}) - \mathbf{x}^{(0)}_{S_2}(u_{j''})\|_2 + \frac{1}{|\mathcal{R}|^2} \left(\sum_{r=1}^{L-1} \binom{L-r}{1}\right) (\mathbb{1}[r_{i''} \neq r_{j''}] + \mathbb{1}[q_1 \neq q_2]) +$$

$$\ldots$$

$$\sum_{i^{(L-1)},j^{(L-1)}} \boldsymbol{P}^{(2)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-2)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-2)}}\right)}[i^{(L-1)},j^{(L-1)}] \Bigg(\Bigg(\Bigg(\binom{L}{L-1} + \sum_{r=1}^{1} \binom{L-r}{L-1}\Bigg) \|\mathbf{x}^{(0)}_{S_1}(u_{i^{(L-1)}}) - \mathbf{x}^{(0)}_{S_2}(u_{j^{(L-1)}})\|_2$$

$$+ \frac{1}{|\mathcal{R}|^2} \left(\sum_{r=1}^{2} \binom{L-r}{L-2}\right) \Big(\mathbb{1}[r_{i^{(L-1)}} \neq r_{j^{(L-1)}}] + \mathbb{1}[q_1 \neq q_2]\Big) +$$

$$\sum_{i^{(L)},j^{(L)}} \boldsymbol{P}^{(1)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-1)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-1)}}\right)}[i^{(L)},j^{(L)}] \Bigg(\binom{L}{L} \|\mathbf{x}^{(0)}_{S_1}(u_{i^{(L)}}) - \mathbf{x}^{(0)}_{S_2}(u_{j^{(L)}})\|_2$$

$$+\frac{1}{|\mathcal{R}|^2} \left(\sum_{r=1}^{1} \binom{L-r}{L-1}\right) \Big(\mathbb{1}[r_{i^{(L)}} \neq r_{j^{(L)}}] + \mathbb{1}[q_1 \neq q_2]\Big)\Bigg)\Bigg) \ldots \Bigg)\Bigg)\Bigg)$$

$$\leq \left(\prod_{k=1}^{L} \eta^{(k)}\right) \Bigg(\Bigg(\Bigg(\binom{L}{0} + \binom{L}{1}\Bigg) \|\mathbf{x}^{(0)}_{S_1}(v_i) - \mathbf{x}^{(0)}_{S_2}(v_j)\|_2 + \frac{1}{|\mathcal{R}|^2} (\mathbb{1}[r_{\text{root}} \neq r_{\text{root}}] + \mathbb{1}[q_1 \neq q_2]) +$$

$$\sum_{i',j'} \boldsymbol{P}^{(L)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j'] \Bigg(\Bigg(\Bigg(\binom{L}{1} + \binom{L}{2}\Bigg) \|\mathbf{x}^{(0)}_{S_1}(u_{i'}) - \mathbf{x}^{(0)}_{S_2}(u_{j'})\|_2 + \frac{1}{|\mathcal{R}|^2} \binom{L}{1} (\mathbb{1}[r_{i'} \neq r_{j'}] + \mathbb{1}[q_1 \neq q_2]) +$$

$$\sum_{i'',j''} \boldsymbol{P}^{(L-1)}_{\widetilde{\mathcal{N}}_{S_1}(v_{i'}),\widetilde{\mathcal{N}}_{S_2}(v_{j'})}[i'',j''] \Bigg(\Bigg(\Bigg(\binom{L}{2} + \binom{L}{3}\Bigg) \|\mathbf{x}^{(0)}_{S_1}(u_{i''}) - \mathbf{x}^{(0)}_{S_2}(u_{j''})\|_2 + \frac{1}{|\mathcal{R}|^2} \binom{L}{2} (\mathbb{1}[r_{i''} \neq r_{j''}] + \mathbb{1}[q_1 \neq q_2]) +$$

$$\ldots$$

$$\sum_{i^{(L-1)},j^{(L-1)}} \boldsymbol{P}^{(2)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-2)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-2)}}\right)}[i^{(L-1)},j^{(L-1)}] \Bigg(\Bigg(\Bigg(\binom{L}{L-1} + \binom{L}{L}\Bigg) \|\mathbf{x}^{(0)}_{S_1}(u_{i^{(L-1)}}) - \mathbf{x}^{(0)}_{S_2}(u_{j^{(L-1)}})\|_2$$

$$+ \frac{1}{|\mathcal{R}|^2} \binom{L}{L-1} \Big(\mathbb{1}[r_{i^{(L-1)}} \neq r_{j^{(L-1)}}] + \mathbb{1}[q_1 \neq q_2]\Big) +$$

$$\sum_{i^{(L)},j^{(L)}} \boldsymbol{P}^{(1)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-1)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-1)}}\right)}[i^{(L)},j^{(L)}]\left(\binom{L}{L}\|\mathbf{x}^{(0)}_{S_1}(u_{i^{(L)}})-\mathbf{x}^{(0)}_{S_2}(u_{j^{(L)}})\|_2\right.$$

$$\left.+\frac{1}{|\mathcal{R}|^2}\binom{L}{L}\left(\mathbb{1}[r_{i^{(L)}}\neq r_{j^{(L)}}]+\mathbb{1}[q_1\neq q_2]\right)\right)\Bigg)\ldots\Bigg)\Bigg)\Bigg)$$

$$\leq\left(\binom{L}{0}+\binom{L}{1}\right)\left(\prod_{k=1}^{L}\eta^{(k)}\right)\Bigg(\|\mathbf{x}^{(0)}_{S_1}(v_i)-\mathbf{x}^{(0)}_{S_2}(v_j)\|_2+\frac{1}{|\mathcal{R}|^2}\frac{\binom{L}{0}}{\binom{L}{0}+\binom{L}{1}}\left(\mathbb{1}[r_{\text{root}}\neq r_{\text{root}}]+\mathbb{1}[q_1\neq q_2]\right)+$$

$$\frac{\binom{L}{1}+\binom{L}{2}}{\binom{L}{0}+\binom{L}{1}}\sum_{i',j'}\boldsymbol{P}^{(L)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j']\Bigg(\|\mathbf{x}^{(0)}_{S_1}(u_{i'})-\mathbf{x}^{(0)}_{S_2}(u_{j'})\|_2+\frac{1}{|\mathcal{R}|^2}\frac{\binom{L}{1}}{\binom{L}{1}+\binom{L}{2}}\left(\mathbb{1}[r_{i'}\neq r_{j'}]+\mathbb{1}[q_1\neq q_2]\right)+$$

$$\frac{\binom{L}{2}+\binom{L}{3}}{\binom{L}{1}+\binom{L}{2}}\sum_{i'',j''}\boldsymbol{P}^{(L-1)}_{\widetilde{\mathcal{N}}_{S_1}(v_{i'}),\widetilde{\mathcal{N}}_{S_2}(v_{j'})}[i'',j'']\Bigg(\|\mathbf{x}^{(0)}_{S_1}(u_{i''})-\mathbf{x}^{(0)}_{S_2}(u_{j''})\|_2+\frac{1}{|\mathcal{R}|^2}\frac{\binom{L}{2}}{\binom{L}{2}+\binom{L}{3}}\left(\mathbb{1}[r_{i''}\neq r_{j''}]+\mathbb{1}[q_1\neq q_2]\right)+$$

$$\ldots$$

$$\frac{\binom{L}{L-1}+\binom{L}{L}}{\binom{L}{L-2}+\binom{L}{L-1}}\sum_{i^{(L-1)},j^{(L-1)}}\boldsymbol{P}^{(2)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-2)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-2)}}\right)}[i^{(L-1)},j^{(L-1)}]\Big(\|\mathbf{x}^{(0)}_{S_1}(u_{i^{(L-1)}})-\mathbf{x}^{(0)}_{S_2}(u_{j^{(L-1)}})\|_2$$

$$+\frac{1}{|\mathcal{R}|^2}\frac{\binom{L}{L-1}}{\binom{L}{L-1}+\binom{L}{L}}\left(\mathbb{1}[r_{i^{(L-1)}}\neq r_{j^{(L-1)}}]+\mathbb{1}[q_1\neq q_2]\right)+$$

$$\frac{\binom{L}{L}}{\binom{L}{L-1}+\binom{L}{L}}\sum_{i^{(L)},j^{(L)}}\boldsymbol{P}^{(1)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-1)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-1)}}\right)}[i^{(L)},j^{(L)}]\Big(\|\mathbf{x}^{(0)}_{S_1}(u_{i^{(L)}})-\mathbf{x}^{(0)}_{S_2}(u_{j^{(L)}})\|_2$$

$$+\frac{1}{|\mathcal{R}|^2}\left(\mathbb{1}[r_{i^{(L)}}\neq r_{j^{(L)}}]+\mathbb{1}[q_1\neq q_2]\right)\Big)\Big)\ldots\Big)\Big)\Big)$$

$$\leq\left(\binom{L}{0}+\binom{L}{1}\right)\left(\prod_{k=1}^{L}\eta^{(k)}\right)\Big(\|\mathbf{x}^{(0)}_{S_1}(v_i)-\mathbf{x}^{(0)}_{S_2}(v_j)\|_2+\frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{\text{root}}\neq r_{\text{root}}]+\mathbb{1}[q_1\neq q_2])+$$

$$\frac{\binom{L}{1}+\binom{L}{2}}{\binom{L}{0}+\binom{L}{1}}\sum_{i',j'}\boldsymbol{P}^{(L)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}[i',j']\Big(\|\mathbf{x}^{(0)}_{S_1}(u_{i'})-\mathbf{x}^{(0)}_{S_2}(u_{j'})\|_2+\frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{i'}\neq r_{j'}]+\mathbb{1}[q_1\neq q_2])+$$

$$\frac{\binom{L}{2}+\binom{L}{3}}{\binom{L}{1}+\binom{L}{2}}\sum_{i'',j''}\boldsymbol{P}^{(L-1)}_{\widetilde{\mathcal{N}}_{S_1}(v_{i'}),\widetilde{\mathcal{N}}_{S_2}(v_{j'})}[i'',j'']\Big(\|\mathbf{x}^{(0)}_{S_1}(u_{i''})-\mathbf{x}^{(0)}_{S_2}(u_{j''})\|_2+\frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{i''}\neq r_{j''}]+\mathbb{1}[q_1\neq q_2])+$$

$$\ldots$$

$$\frac{\binom{L}{L-1}+\binom{L}{L}}{\binom{L}{L-2}+\binom{L}{L-1}}\sum_{i^{(L-1)},j^{(L-1)}}\boldsymbol{P}^{(2)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-2)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-2)}}\right)}[i^{(L-1)},j^{(L-1)}]\Big(\|\mathbf{x}^{(0)}_{S_1}(u_{i^{(L-1)}})-\mathbf{x}^{(0)}_{S_2}(u_{j^{(L-1)}})\|_2$$

$$+\frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{i^{(L-1)}}\neq r_{j^{(L-1)}}]+\mathbb{1}[q_1\neq q_2])+$$

$$\frac{\binom{L}{L}}{\binom{L}{L-1}+\binom{L}{L}}\sum_{i^{(L)},j^{(L)}}\boldsymbol{P}^{(1)}_{\widetilde{\mathcal{N}}_{S_1}\left(v_{i^{(L-1)}}\right),\widetilde{\mathcal{N}}_{S_2}\left(v_{j^{(L-1)}}\right)}[i^{(L)},j^{(L)}]\Big(\|\mathbf{x}^{(0)}_{S_1}(u_{i^{(L)}})-\mathbf{x}^{(0)}_{S_2}(u_{j^{(L)}})\|_2$$

$$+\frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{i^{(L)}}\neq r_{j^{(L)}}]+\mathbb{1}[q_1\neq q_2])\Big)\Big)\ldots\Big)\Big)\Big)$$

If we set each transportation plan as the optimal transportation plan of the $\text{RTD}((r_{\text{root}},T^{(L)}_{S_1}(v_i)),(r_{\text{root}},T^{(L)}_{S_2}(v_j)))$ with the weight function in Equation 3,

$$w(l)=\begin{cases}\frac{\binom{L}{L-l+1}+\binom{L}{L-l+2}}{\binom{L}{L-l}+\binom{L}{L-l+1}} & l\geq 2\\ \frac{\binom{L}{L-l+1}}{\binom{L}{L-l}+\binom{L}{L-l+1}} & l=1\end{cases}\tag{3}$$

we can get the bound below

$$\|\mathbf{x}^{(L)}_{S_1}(v_i)-\mathbf{x}^{(L)}_{S_2}(v_j)\|_2+\frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{\text{root}}\neq r_{\text{root}}]+\mathbb{1}[q_1\neq q_2])\leq(L+1)\left(\prod_{l=0}^{L}\eta^{(k)}\right)\text{RTD}((r_{\text{root}},T^{(L)}_{S_1}(v_i)),(r_{\text{root}},T^{(L)}_{S_2}(v_j)))$$

Finally if we set $\boldsymbol{P}^{(l)}_{\widetilde{\mathcal{N}}_{S_1}(v_i),\widetilde{\mathcal{N}}_{S_2}(v_j)}$ as the optimal transportation plan of the $\text{OT}_{\text{RTD}}(S_1,S_2)$, we can get following equations.

$$|f_{\mathbf{w}}(S_1)-f_{\mathbf{w}}(S_2)|$$

$$\leq A_{\text{rd}}\|\mathbf{x}^{(L)}_{S_1}(h_1)-\mathbf{x}^{(L)}_{S_2}(h_2)\|_2+B_{\text{rd}}\|\mathbf{x}^{(L)}_{S_1}(t_1)-\mathbf{x}^{(L)}_{S_2}(t_2)\|_2+D_{\text{rd}}\mathbb{1}[q_1\neq q_2]+$$

$$
\begin{aligned}
&C_{\text{rd}}A_{\text{grd}}\sum_{i,j}\boldsymbol{P}_{\mathcal{X}_{S_1},\mathcal{X}_{S_2}}[i,j]\|\mathbf{x}^{(L)}_{S_1}(v_i)-\mathbf{x}^{(L)}_{S_2}(v_j)\|_2\\
\leq&\eta^{(L+1)}\Bigg(\|\mathbf{x}^{(L)}_{S_1}(h_1)-\mathbf{x}^{(L)}_{S_2}(h_2)\|_2+\|\mathbf{x}^{(L)}_{S_1}(t_1)-\mathbf{x}^{(L)}_{S_2}(t_2)\|_2+\frac{2+\max(|\mathcal{V}_{S_1}|,|\mathcal{V}_{S_2}|)}{|\mathcal{R}|^2}\mathbb{1}[q_1\neq q_2]\\
&+\sum_{i,j}\boldsymbol{P}_{\mathcal{X}_{S_1},\mathcal{X}_{S_2}}[i,j]\|\mathbf{x}^{(L)}_{S_1}(v_i)-\mathbf{x}^{(L)}_{S_2}(v_j)\|_2\Bigg)\\
\leq&\eta^{(L+1)}\Bigg(\|\mathbf{x}^{(L)}_{S_1}(h_1)-\mathbf{x}^{(L)}_{S_2}(h_2)\|_2+\frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{\text{root}}\neq r_{\text{root}}]+\mathbb{1}[q_1\neq q_2])+\\
&\|\mathbf{x}^{(L)}_{S_1}(t_1)-\mathbf{x}^{(L)}_{S_2}(t_2)\|_2+\frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{\text{root}}\neq r_{\text{root}}]+\mathbb{1}[q_1\neq q_2])+\\
&\sum_{i,j}\boldsymbol{P}_{\mathcal{X}_{S_1},\mathcal{X}_{S_2}}[i,j]\left(\|\mathbf{x}^{(L)}_{S_1}(v_i)-\mathbf{x}^{(L)}_{S_2}(v_j)\|_2+\frac{1}{|\mathcal{R}|^2}(\mathbb{1}[r_{\text{root}}\neq r_{\text{root}}]+\mathbb{1}[q_1\neq q_2])\right)\Bigg)\\
\leq&(L+1)\left(\prod_{l=1}^{L+1}\eta^{(k)}\right)\left(\text{RTD}((r_{\text{root}},T^{(L)}_{S_1}(h_1)),(r_{\text{root}},T^{(L)}_{S_2}(h_2)))+\text{RTD}((r_{\text{root}},T^{(L)}_{S_1}(t_1)),(r_{\text{root}},T^{(L)}_{S_2}(t_2)))+\text{OT}_{\text{RTD}}(S_1,S_2)\right)\\
\leq&(L+1)\left(\prod_{l=1}^{L+1}\eta^{(k)}\right)\text{RTMD}(S_1,S_2)
\end{aligned}
$$

□

D. Proof for Generalization Bound

We prove Theorem 5.3 and Theorem 5.4.

D.1. Proof for Theorem 5.3

The risks associated with any distribution $\mathcal{Q}$ over the parameters of a stochastic SMPNN can be defined as follows.

$$\widehat{\mathcal{L}}_G(Q,\gamma)=\mathbb{E}_{f_{\mathbf{w}}\sim\mathcal{Q}}\widehat{\mathcal{L}}_G(f_{\mathbf{w}},\gamma)$$

$$\mathcal{L}_G(Q,\gamma)=\mathbb{E}_{f_{\mathbf{w}}\sim\mathcal{Q}}\mathcal{L}_G(f_{\mathbf{w}},\gamma)$$

Lemma D.1 is a modified version of the PAC-Bayesian generalization bound proposed in (Ma et al., 2021) for subgraph reasoning models.

Lemma D.1 (Generalization Bound for Stochastic Subgraph Reasoning Models)**.** *Given G_{tr},G_{inf}, and a subgraph reasoning model with a subgraph extractor g and an* SMPNN *$f_{\mathbf{w}}$, for any prior distribution $\mathcal{P}$ on the parameter space of $f_{\mathbf{w}}$, posterior distribution $\mathcal{Q}$ on the parameter space of $f_{\mathbf{w}}$, and $\lambda>0$, the following holds with probability at least $1-\delta$:*

$$\mathcal{L}_{G_{\text{inf}}}(\mathcal{Q},\frac{\gamma}{2})\leq\widehat{\mathcal{L}}_{G_{\text{tr}}}(\mathcal{Q},\gamma)+\frac{1}{\lambda}\left(KL(\mathcal{Q}||\mathcal{P})+\ln\frac{1}{\delta}+\frac{\lambda^2}{4|\mathcal{T}_{\text{tr}}|}+D\left(\mathcal{P},\lambda,\gamma\right)\right)$$

where $D(\mathcal{P},\lambda,\gamma)=\ln\left(\mathbb{E}_{\mathbf{w}\sim\mathcal{P}}\left[\exp\left(\lambda\left(\mathcal{L}_{G_{\text{tr}}}(f_{\mathbf{w}},\gamma)-\mathcal{L}_{G_{\text{inf}}}(f_{\mathbf{w}},\gamma)\right)\right)\right]\right)$

From Lemma D.1, we derive the PAC-Bayesian generalization bound for deterministic subgraph reasoning models in Theorem 5.3.

Theorem 5.3 (PAC-Bayesian Generalization Bound of Deterministic Subgraph Reasoning Models)**.** *Given $G_{\text{tr}}=(\mathcal{V}_{\text{tr}},\mathcal{R},\mathcal{F}_{\text{tr}}\cup\mathcal{T}_{\text{tr}})$, $G_{\text{inf}}=(\mathcal{V}_{\text{inf}},\mathcal{R},\mathcal{F}_{\text{inf}}\cup\mathcal{T}_{\text{inf}})$, and a subgraph reasoning model with a subgraph extractor g and an* SMPNN *$f_{\mathbf{w}}$, for any prior distribution $\mathcal{P}$ and posterior distribution $\mathcal{Q}$ on the parameter space of $f_{\mathbf{w}}$ constructed by adding random noise $\ddot{\mathbf{w}}$ to $\mathbf{w}$ such that $\mathbb{P}(\max(\max_{e\in\mathcal{T}_{\text{tr}}}|f_{\tilde{\mathbf{w}}}(g(G_{\text{tr}},e))-f_{\mathbf{w}}(g(G_{\text{tr}},e))|,\max_{e\in\mathcal{T}_{\text{inf}}}|f_{\tilde{\mathbf{w}}}(g(G_{\text{inf}},e))-f_{\mathbf{w}}(g(G_{\text{inf}},e))|)<\frac{\gamma}{4})>\frac{1}{2}$,* *and* $\gamma > 0$*,* $\lambda > 0$*, the following holds with probability at least* $1 - \delta$*:*

$$\begin{aligned}&\mathcal{L}_{G_{\text{inf}}}(f_{\mathbf{w}}, 0) \leq \widehat{\mathcal{L}}_{G_{\text{tr}}}(f_{\mathbf{w}}, \gamma) + \\ &\frac{1}{\lambda}\left(2KL(\mathcal{Q}\|\mathcal{P}) + \ln\frac{4}{\delta} + \frac{\lambda^2}{4|\mathcal{T}_{\text{tr}}|} + D\left(\mathcal{P}, \lambda, \frac{\gamma}{2}\right)\right)\end{aligned}$$

where $D\big(\mathcal{P},\lambda,\frac{\gamma}{2}\big)$ *is the expected risk discrepancy between* G_{tr} *and* G_{inf} *and* $KL(\mathcal{Q}\|\mathcal{P})$ *is a KL divergence of* $\mathcal{Q}$ *from* $\mathcal{P}$*.*

Proof. The posterior distribution $\mathcal{Q}$ is the probability distribution of $\widetilde{\mathbf{w}}$ in the parameter space $\mathcal{H}$. We define the following set in the parameter space $\mathcal{H}$.

$$\mathcal{C} = \{\widetilde{\mathbf{w}} \in \mathcal{H}|\max(\max_{e\in\mathcal{T}_{\text{tr}}}|f_{\widetilde{\mathbf{w}}}(g(G_{\text{tr}}, e)) - f_{\mathbf{w}}(g(G_{\text{tr}}, e))|, \max_{e\in\mathcal{T}_{\text{inf}}}|f_{\widetilde{\mathbf{w}}}(g(G_{\text{inf}}, e)) - f_{\mathbf{w}}(g(G_{\text{inf}}, e))|) < \frac{\gamma}{4}\} \subset \mathcal{H}$$

Then, $p = \mathbb{P}_{\widetilde{\mathbf{w}}\sim\mathcal{Q}}(\widetilde{\mathbf{w}} \in \mathcal{C}) > \frac{1}{2}$. Using $\mathcal{Q}$, we create the following distributions.

$$\grave{\mathcal{Q}} = \begin{cases}\frac{1}{p}\mathcal{Q}(\widetilde{\mathbf{w}}) & \widetilde{\mathbf{w}} \in \mathcal{C} \\ 0 & \widetilde{\mathbf{w}} \in \mathcal{H}\backslash\mathcal{C}\end{cases}, \acute{\mathcal{Q}} = \begin{cases}0 & \widetilde{\mathbf{w}} \in \mathcal{C} \\ \frac{1}{1-p}\mathcal{Q}(\widetilde{\mathbf{w}}) & \widetilde{\mathbf{w}} \in \mathcal{H}\backslash\mathcal{C}\end{cases}$$

For any $(h, r, t) \in \mathcal{T}_{\text{inf}}$ and $\widetilde{\mathbf{w}} \sim \grave{\mathcal{Q}}$

$$\begin{aligned}&|y_{\text{hrt}}f_{\widetilde{\mathbf{w}}}(g(G_{\text{inf}}, (h, r, t))) - y_{\text{hrt}}f_{\mathbf{w}}(g(G_{\text{inf}}, (h, r, t)))| \\ =&|y_{\text{hrt}}\left(f_{\widetilde{\mathbf{w}}}(g(G_{\text{inf}}, (h, r, t))) - f_{\mathbf{w}}(g(G_{\text{inf}}, (h, r, t)))\right)| \\ =&|f_{\widetilde{\mathbf{w}}}(g(G_{\text{inf}}, (h, r, t))) - f_{\mathbf{w}}(g(G_{\text{inf}}, (h, r, t)))| \leq \frac{\gamma}{4}\end{aligned}$$

Then,

$$f_{\mathbf{w}}(g(G_{\text{inf}}, (h, r, t))) \leq 0 \rightarrow f_{\widetilde{\mathbf{w}}}(g(G_{\text{inf}}, (h, r, t))) \leq \frac{\gamma}{4}$$

which indicates that

$$\mathbb{1}[f_{\mathbf{w}}(g(G_{\text{inf}}, (h, r, t))) \leq 0] \leq \mathbb{1}[f_{\widetilde{\mathbf{w}}}(g(G_{\text{inf}}, (h, r, t))) \leq \frac{\gamma}{4}]$$

Therefore, $\mathcal{L}_{G_{\text{inf}}}(f_{\mathbf{w}}, 0) \leq \mathcal{L}_{G_{\text{inf}}}(f_{\widetilde{\mathbf{w}}}, \frac{\gamma}{4})$ for any $\widetilde{\mathbf{w}} \sim \grave{\mathcal{Q}}$, which means that that

$$\mathcal{L}_{G_{\text{inf}}}(f_{\mathbf{w}}, 0) \leq \mathbb{E}_{\widetilde{\mathbf{w}}\sim\grave{\mathcal{Q}}}\mathcal{L}_{G_{\text{inf}}}(f_{\widetilde{\mathbf{w}}}, \frac{\gamma}{4})$$

Also, for any $(h, r, t) \in \mathcal{T}_{\text{tr}}$ and $\widetilde{\mathbf{w}} \sim \grave{\mathcal{Q}}$

$$\begin{aligned}&|y_{\text{hrt}}f_{\widetilde{\mathbf{w}}}(g(G_{\text{tr}}, (h, r, t))) - y_{\text{hrt}}f_{\mathbf{w}}(g(G_{\text{tr}}, (h, r, t)))| \\ =&|y_{\text{hrt}}\left(f_{\widetilde{\mathbf{w}}}(g(G_{\text{tr}}, (h, r, t))) - f_{\mathbf{w}}(g(G_{\text{tr}}, (h, r, t)))\right)| \\ =&|f_{\widetilde{\mathbf{w}}}(g(G_{\text{tr}}, (h, r, t))) - f_{\mathbf{w}}(g(G_{\text{tr}}, (h, r, t)))| \leq \frac{\gamma}{4}\end{aligned}$$

Then,

$$f_{\widetilde{\mathbf{w}}}(g(G_{\text{tr}}, (h, r, t))) \leq \frac{\gamma}{2} \rightarrow f_{\mathbf{w}}(g(G_{\text{tr}}, (h, r, t))) \leq \gamma$$

which indicates that

$$\mathbb{1}[f_{\widetilde{\mathbf{w}}}(g(G_{\text{tr}}, (h, r, t))) \leq \frac{\gamma}{2}] \leq \mathbb{1}[f_{\mathbf{w}}(g(G_{\text{tr}}, (h, r, t))) \leq \gamma]$$

Therefore, $\widehat{\mathcal{L}}_{G_{\text{tr}}}(f_{\widetilde{\mathbf{w}}}, \frac{\gamma}{2}) \leq \widehat{\mathcal{L}}_{G_{\text{tr}}}(f_{\mathbf{w}}, \gamma)$ for any $\widetilde{\mathbf{w}} \sim \grave{\mathcal{Q}}$, meaning that

$$\mathbb{E}_{\widetilde{\mathbf{w}}\sim\grave{\mathcal{Q}}}\widehat{\mathcal{L}}_{G_{\text{tr}}}(f_{\widetilde{\mathbf{w}}}, \frac{\gamma}{2}) \leq \widehat{\mathcal{L}}_{G_{\text{tr}}}(f_{\mathbf{w}}, \gamma)$$

Then, with probability at least $1-\delta$,

$$\begin{aligned}\mathcal{L}_{G_{\text{inf}}}(f_{\mathbf{w}},0) \leq& \mathbb{E}_{\widetilde{\mathbf{w}}\sim\grave{\mathcal{Q}}}\mathcal{L}_{G_{\text{inf}}}(f_{\widetilde{\mathbf{w}}},\frac{\gamma}{4})\\ \leq& \mathbb{E}_{\widetilde{\mathbf{w}}\sim\grave{\mathcal{Q}}}\widehat{\mathcal{L}}_{G_{\text{tr}}}(f_{\widetilde{\mathbf{w}}},\frac{\gamma}{2})+\frac{1}{\lambda}\left(KL(\grave{\mathcal{Q}}||\mathcal{P})+\ln\frac{1}{\delta}+\frac{\lambda^2}{4|\mathcal{T}_{\text{tr}}|}+D\left(\mathcal{P},\lambda,\frac{\gamma}{2}\right)\right)\\ \leq& \widehat{\mathcal{L}}_{G_{\text{tr}}}(f_{\mathbf{w}},\gamma)+\frac{1}{\lambda}\left(KL(\grave{\mathcal{Q}}||\mathcal{P})+\ln\frac{1}{\delta}+\frac{\lambda^2}{4|\mathcal{T}_{\text{tr}}|}+D\left(\mathcal{P},\lambda,\frac{\gamma}{2}\right)\right)\end{aligned}$$

by applying Lemma D.1. Also, the following holds:

$$\begin{aligned}KL(\mathcal{Q}||\mathcal{P}) =& \int_{\widetilde{\mathbf{w}}\in\mathcal{C}}\mathcal{Q}\ln\frac{\mathcal{Q}}{\mathcal{P}}d\widetilde{\mathbf{w}}+\int_{\widetilde{\mathbf{w}}\in\mathcal{H}\backslash\mathcal{C}}\mathcal{Q}\ln\frac{\mathcal{Q}}{\mathcal{P}}d\widetilde{\mathbf{w}}\\ =& p\int_{\widetilde{\mathbf{w}}\in\mathcal{C}}\frac{\mathcal{Q}}{p}\ln\frac{\mathcal{Q}}{p\mathcal{P}}d\widetilde{\mathbf{w}}+(1-p)\int_{\widetilde{\mathbf{w}}\in\mathcal{H}\backslash\mathcal{C}}\frac{\mathcal{Q}}{1-p}\ln\frac{\mathcal{Q}}{(1-p)\mathcal{P}}d\widetilde{\mathbf{w}}\\ &+\int_{\widetilde{\mathbf{w}}\in\mathcal{C}}\mathcal{Q}\ln p d\widetilde{\mathbf{w}}+\int_{\widetilde{\mathbf{w}}\in\mathcal{H}\backslash\mathcal{C}}\mathcal{Q}\ln(1-p)d\widetilde{\mathbf{w}}\\ =& pKL(\grave{\mathcal{Q}}||\mathcal{P})+(1-p)KL(\acute{\mathcal{Q}}||\mathcal{P})+p\ln p+(1-p)\ln(1-p)\end{aligned}$$

Since $\frac{1}{2}<p<1$, $-\ln 2<p\ln p+(1-p)\ln(1-p)<0$ holds. Considering that KL divergence is non-negative,

$$\begin{aligned}KL(\grave{\mathcal{Q}}||\mathcal{P}) =& \frac{1}{p}KL(\mathcal{Q}||\mathcal{P})-(1-p)KL(\acute{\mathcal{Q}}||\mathcal{P})-p\ln p-(1-p)\ln(1-p)\\ \leq& \frac{1}{p}\left(KL(\mathcal{Q}||\mathcal{P})+\ln 2\right)\leq 2KL(\mathcal{Q}||\mathcal{P})+2\ln 2\end{aligned}$$

Consequently,

$$\begin{aligned}\mathcal{L}_{G_{\text{inf}}}(f_{\mathbf{w}},0) \leq& \widehat{\mathcal{L}}_{G_{\text{tr}}}(f_{\mathbf{w}},\gamma)+\frac{1}{\lambda}\left(KL(\grave{\mathcal{Q}}||\mathcal{P})+\ln\frac{1}{\delta}+\frac{\lambda^2}{4|\mathcal{T}_{\text{tr}}|}+D\left(\mathcal{P},\lambda,\frac{\gamma}{2}\right)\right)\\ \leq& \widehat{\mathcal{L}}_{G_{\text{tr}}}(f_{\mathbf{w}},\gamma)+\frac{1}{\lambda}\left(2KL(\mathcal{Q}||\mathcal{P})+\ln\frac{4}{\delta}+\frac{\lambda^2}{4|\mathcal{T}_{\text{tr}}|}+D\left(\mathcal{P},\lambda,\frac{\gamma}{2}\right)\right)\end{aligned}$$

□

D.2. Proof for Theorem 5.4

The degree of difference between the subgraphs extracted from the training KG and the inference KG can be represented through the optimal transport between the multisets of subgraphs extracted from the sets of triplets $\mathcal{T}_{\text{tr}}$ and $\mathcal{T}_{\text{inf}}$. Since the number of the triplets differs, an empty subgraph S_{blank}, containing only virtual head and tail entities, is defined to facilitate the computation of optimal transport. These virtual head and tail entities are identical to the virtual entities forming the blank tree and always have initial labels distinct from all other entities. The multisets of subgraphs, extended with the empty subgraph set, are defined as follows:

Definition D.2 (Empty Subgraph Augmentation)**.** The empty subgraph augmentation ψ for two triplet sets $\mathcal{T}_{\text{tr}}$ and $\mathcal{T}_{\text{inf}}$ is defined as follows:

$$\psi(\mathcal{T}_{\text{inf}},\mathcal{T}_{\text{tr}})=\begin{cases}\{\{g(G_{\text{inf}},e_i)|e_i\in\mathcal{T}_{\text{inf}}\}\}\cup\bigcup_{k=1}^{n_2-n_1}\{\{S_{\text{blank}}\}\},\{\{g(G_{\text{tr}},e_j)|e_j\in\mathcal{T}_{\text{tr}}\}\} & n_1<n_2\\ \{\{g(G_{\text{inf}},e_i)|e_i\in\mathcal{T}_{\text{inf}}\}\},\{\{g(G_{\text{tr}},e_j)|e_j\in\mathcal{T}_{\text{tr}}\}\}\cup\bigcup_{k=1}^{n_1-n_2}\{\{S_{\text{blank}}\}\} & n_1\geq n_2\end{cases}$$

where $|\mathcal{T}_{\text{inf}}|=n_1, |\mathcal{T}_{\text{tr}}|=n_2$.

In this case, the score calculated by the SMPNNs $f_{\mathbf{w}}$ for an empty subgraph is always 0.

Now, we prove Theorem 5.4.

Theorem 5.4 (Bound of the $D(\mathcal{P}, \lambda, \gamma)$)**.** *Given $G_{\text{tr}} = (\mathcal{V}_{\text{tr}}, \mathcal{R}, \mathcal{F}_{\text{tr}} \cup \mathcal{T}_{\text{tr}})$, $G_{\text{inf}} = (\mathcal{V}_{\text{inf}}, \mathcal{R}, \mathcal{F}_{\text{inf}} \cup \mathcal{T}_{\text{inf}})$, and an* SMPNN $f_{\mathbf{w}}$ *with stability C_f, for any prior distribution $\mathcal{P}$ and posterior distribution $\mathcal{Q}$ on the parameter space of $f_{\mathbf{w}}$, and $\lambda > 0$, the following holds:*

$$D(\mathcal{P}, \lambda, \gamma) \leq \lambda \bigg(\max(0, \frac{|\mathcal{T}_{\text{tr}}|}{|\mathcal{T}_{\text{inf}}|} - 1) + \frac{2 \, \text{OT}_{\text{RTMD}}(\psi(\mathcal{T}_{\text{inf}}, \mathcal{T}_{\text{tr}}))}{\gamma C_f \max(|\mathcal{T}_{\text{inf}}|, |\mathcal{T}_{\text{tr}}|)} \bigg)$$

where ψ is the empty subgraph augmentation defined in Definition D.2.

Proof. Let $\boldsymbol{P}_{\psi(\mathcal{T}_{\text{inf}}, \mathcal{T}_{\text{tr}})}$ be a transportation plan between the augmented multisets of subgraphs $\psi(\mathcal{T}_{\text{inf}}, \mathcal{T}_{\text{tr}})$. Then, the following equations hold.

$$\begin{aligned}
&\mathcal{L}_{\mathcal{G}_{\text{inf}}}(f_{\mathbf{w}}, \frac{\gamma}{2}) - \mathcal{L}_{\mathcal{G}_{\text{tr}}}(f_{\mathbf{w}}, \gamma) \\
=& \mathbb{E}_y[\widehat{\mathcal{L}}_{\mathcal{G}_{\text{inf}}}(f_{\mathbf{w}}, \frac{\gamma}{2})] - \mathbb{E}_y[\widehat{\mathcal{L}}_{\mathcal{G}_{\text{tr}}}(f_{\mathbf{w}}, \gamma)] \\
=& \mathbb{E}_y \Big[\widehat{\mathcal{L}}_{\mathcal{G}_{\text{inf}}}(f_{\mathbf{w}}, \frac{\gamma}{2}) - \widehat{\mathcal{L}}_{\mathcal{G}_{\text{tr}}}(f_{\mathbf{w}}, \gamma) \Big] \\
=& \mathbb{E}_y \Big[\frac{1}{|\mathcal{T}_{\text{inf}}|} \sum_{e_i \in \mathcal{T}_{\text{inf}}} \mathbb{1}[y_i f_{\mathbf{w}}(g(G_{\text{inf}}, e_i)) \leq \frac{\gamma}{2}] - \frac{1}{|\mathcal{T}_{\text{tr}}|} \sum_{e_i \in \mathcal{T}_{\text{tr}}} \mathbb{1}[y_i f_{\mathbf{w}}(g(G_{\text{tr}}, e_i)) \leq \gamma] \Big] \\
=& \mathbb{E}_y \Big[\sum_{(S_i, S_j) \in \psi(\mathcal{T}_{\text{inf}}, \mathcal{T}_{\text{tr}})} \boldsymbol{P}_{\psi(\mathcal{T}_{\text{inf}}, \mathcal{T}_{\text{tr}})}[i, j] \left(\frac{\mathbb{1}[y_i f_{\mathbf{w}}(S_i) \leq \frac{\gamma}{2}]}{|\mathcal{T}_{\text{inf}}|} - \frac{\mathbb{1}[y_j f_{\mathbf{w}}(S_j) \leq \gamma]}{|\mathcal{T}_{\text{tr}}|} \right) \Big]
\end{aligned}$$

First, we consider the case that $|\mathcal{T}_{\text{inf}}| < |\mathcal{T}_{\text{tr}}|$. For any pair of subgraphs $(S_1, S_2) \in \psi(\mathcal{T}_{\text{inf}}, \mathcal{T}_{\text{tr}})$, the following inequalities hold.

$$\frac{\mathbb{1}[y_i f_{\mathbf{w}}(S_i) \leq \frac{\gamma}{2}]}{|\mathcal{T}_{\text{inf}}|} - \frac{\mathbb{1}[y_j f_{\mathbf{w}}(S_j) \leq \gamma]}{|\mathcal{T}_{\text{tr}}|} \leq \frac{1/|\mathcal{T}_{\text{tr}}|}{\gamma/2} |f_{\mathbf{w}}(S_i) - f_{\mathbf{w}}(S_j)| + \frac{1}{|\mathcal{T}_{\text{inf}}|} - \frac{1}{|\mathcal{T}_{\text{tr}}|}$$

Next, we consider the case that $|\mathcal{T}_{\text{inf}}| \geq |\mathcal{T}_{\text{tr}}|$. For any pair of subgraphs $(S_1, S_2) \in \psi(\mathcal{T}_{\text{inf}}, \mathcal{T}_{\text{tr}})$, the following inequalities hold.

$$\frac{\mathbb{1}[y_i f_{\mathbf{w}}(S_i) \leq \frac{\gamma}{2}]}{|\mathcal{T}_{\text{inf}}|} - \frac{\mathbb{1}[y_j f_{\mathbf{w}}(S_j) \leq \gamma]}{|\mathcal{T}_{\text{tr}}|} \leq \frac{1/|\mathcal{T}_{\text{inf}}|}{\gamma/2} |f_{\mathbf{w}}(S_i) - f_{\mathbf{w}}(S_j)|$$

By combining the above two inequalities and applying the Lipschitz continuity of an SMPNNin Definition 4.4,

$$\begin{aligned}
&\mathcal{L}_{\mathcal{G}_{\text{inf}}}(f_{\mathbf{w}}, \frac{\gamma}{2}) - \mathcal{L}_{\mathcal{G}_{\text{tr}}}(f_{\mathbf{w}}, \gamma) \\
=& \mathbb{E}_y \Big[\sum_{(S_i, S_j) \in \psi(\mathcal{T}_{\text{inf}}, \mathcal{T}_{\text{tr}})} \boldsymbol{P}_{\psi(\mathcal{T}_{\text{inf}}, \mathcal{T}_{\text{tr}})}[i, j] \left(\frac{\mathbb{1}[y_i f_{\mathbf{w}}(S_i) \leq \frac{\gamma}{2}]}{|\mathcal{T}_{\text{inf}}|} - \frac{\mathbb{1}[y_j f_{\mathbf{w}}(S_j) \leq \gamma]}{|\mathcal{T}_{\text{tr}}|} \right) \Big] \\
\leq& \mathbb{E}_y \Big[\sum_{(S_i, S_j) \in \psi(\mathcal{T}_{\text{inf}}, \mathcal{T}_{\text{tr}})} \boldsymbol{P}_{\psi(\mathcal{T}_{\text{inf}}, \mathcal{T}_{\text{tr}})}[i, j] \left(\frac{2}{\gamma \max(|\mathcal{T}_{\text{inf}}|, |\mathcal{T}_{\text{tr}}|)} |f_{\mathbf{w}}(S_i) - f_{\mathbf{w}}(S_j)| + \max(0, \frac{1}{|\mathcal{T}_{\text{inf}}|} - \frac{1}{|\mathcal{T}_{\text{tr}}|}) \right) \Big] \\
\leq& \mathbb{E}_y \Big[\sum_{(S_i, S_j) \in \psi(\mathcal{T}_{\text{inf}}, \mathcal{T}_{\text{tr}})} \boldsymbol{P}_{\psi(\mathcal{T}_{\text{inf}}, \mathcal{T}_{\text{tr}})}[i, j] \left(\frac{2}{\gamma C_f \max(|\mathcal{T}_{\text{inf}}|, |\mathcal{T}_{\text{tr}}|)} \text{RTMD}(S_i, S_j) + \max(0, \frac{1}{|\mathcal{T}_{\text{inf}}|} - \frac{1}{|\mathcal{T}_{\text{tr}}|}) \right) \Big] \\
=& \mathbb{E}_y \Big[\max(0, \frac{|\mathcal{T}_{\text{tr}}|}{|\mathcal{T}_{\text{inf}}|} - 1) + \frac{2}{\gamma C_f \max(|\mathcal{T}_{\text{inf}}|, |\mathcal{T}_{\text{tr}}|)} \sum_{(S_i, S_j) \in \psi(\mathcal{T}_{\text{inf}}, \mathcal{T}_{\text{tr}})} \boldsymbol{P}_{\psi(\mathcal{T}_{\text{inf}}, \mathcal{T}_{\text{tr}})}[i, j] \text{RTMD}(S_i, S_j) \Big]
\end{aligned}$$

holds. If we set $\boldsymbol{P}_{\psi(\mathcal{T}_{\text{inf}}, \mathcal{T}_{\text{tr}})}$ as the optimal transportation plan of the $\text{OT}_{\text{RTMD}}(\psi(\mathcal{T}_{\text{inf}}, \mathcal{T}_{\text{tr}}))$, we can derive the following inequalities.

$$\mathcal{L}_{\mathcal{G}_{\text{inf}}}(f_{\mathbf{w}}, \frac{\gamma}{2}) - \mathcal{L}_{\mathcal{G}_{\text{tr}}}(f_{\mathbf{w}}, \gamma)$$

$$
\begin{aligned}
&\leq \mathbb{E}_y\Big[\max(0, \frac{|\mathcal{T}_{\text{tr}}|}{|\mathcal{T}_{\text{inf}}|} - 1) + \frac{2}{\gamma C_f \max(|\mathcal{T}_{\text{inf}}|, |\mathcal{T}_{\text{tr}}|)} \sum_{(S_i,S_j)\in\psi(\mathcal{T}_{\text{inf}},\mathcal{T}_{\text{tr}})} \boldsymbol{P}_{\psi(\mathcal{T}_{\text{inf}},\mathcal{T}_{\text{tr}})}[i,j]\text{RTMD}(S_i,S_j)\Big] \\
&\leq \mathbb{E}_y\Big[\max(0, \frac{|\mathcal{T}_{\text{tr}}|}{|\mathcal{T}_{\text{inf}}|} - 1) + \frac{2}{\gamma C_f \max(|\mathcal{T}_{\text{inf}}|, |\mathcal{T}_{\text{tr}}|)} \text{OT}_{\text{RTMD}}(\psi(\mathcal{T}_{\text{inf}},\mathcal{T}_{\text{tr}}))\Big] \\
&= \max(0, \frac{|\mathcal{T}_{\text{tr}}|}{|\mathcal{T}_{\text{inf}}|} - 1) + \frac{2}{\gamma C_f \max(|\mathcal{T}_{\text{inf}}|, |\mathcal{T}_{\text{tr}}|)} \text{OT}_{\text{RTMD}}(\psi(\mathcal{T}_{\text{inf}},\mathcal{T}_{\text{tr}}))
\end{aligned}
$$

Finally, we get

$$
\begin{aligned}
D(\mathcal{P},\lambda,\gamma) =& \ln\left(\mathbb{E}_{\mathbf{w}\sim\mathcal{P}}\left[\exp\left(\lambda\left(\mathcal{L}_{G_{\text{tr}}}(f_{\mathbf{w}},\frac{\gamma}{2}) - \mathcal{L}_{G_{\text{inf}}}(f_{\mathbf{w}},\gamma)\right)\right)\right]\right) \\
\leq& \ln\left(\mathbb{E}_{\mathbf{w}\sim\mathcal{P}}\left[\exp\left(\lambda\left(\max(0, \frac{|\mathcal{T}_{\text{tr}}|}{|\mathcal{T}_{\text{inf}}|} - 1) + \frac{2}{\gamma C_f \max(|\mathcal{T}_{\text{inf}}|, |\mathcal{T}_{\text{tr}}|)} \text{OT}_{\text{RTMD}}(\psi(\mathcal{T}_{\text{inf}},\mathcal{T}_{\text{tr}}))\right)\right)\right]\right) \\
=& \ln\left(\exp\left(\lambda\left(\max(0, \frac{|\mathcal{T}_{\text{tr}}|}{|\mathcal{T}_{\text{inf}}|} - 1) + \frac{2}{\gamma C_f \max(|\mathcal{T}_{\text{inf}}|, |\mathcal{T}_{\text{tr}}|)} \text{OT}_{\text{RTMD}}(\psi(\mathcal{T}_{\text{inf}},\mathcal{T}_{\text{tr}}))\right)\right)\right) \\
=& \lambda\left(\max(0, \frac{|\mathcal{T}_{\text{tr}}|}{|\mathcal{T}_{\text{inf}}|} - 1) + \frac{2}{\gamma C_f \max(|\mathcal{T}_{\text{inf}}|, |\mathcal{T}_{\text{tr}}|)} \text{OT}_{\text{RTMD}}(\psi(\mathcal{T}_{\text{inf}},\mathcal{T}_{\text{tr}}))\right)
\end{aligned}
$$

□

E. Experimental Details

We conduct experiments on the inductive split of three real-world KGs, WN18RR, FB15K-237, and NELL-995 from (Teru et al., 2020). Specifically, we use the v3 of WN18RR (WNv3), v1 of FB15K-237 (FBv1), and v2 of NELL-995 (NLv2). For WNv3 and NLv2, we randomly sample 20% of the prediction triplets from both the training KG and the inference KG due to the large size of the dataset. Additionally, we split the triplet set of the training KG into two parts in a 3:1 ratio, using them as $\mathcal{F}_{\text{tr}}$ and $\mathcal{T}_{\text{tr}}$, respectively. For inference KG, we use the triplets in train.txt as $\mathcal{F}_{\text{inf}}$ and the triplets in test.txt as $\mathcal{T}_{\text{inf}}$. Since all triplets in the original datasets are positive, we generate negative triplets by randomly perturbing either the head or tail entity of each positive triplet in $\mathcal{T}$, following (Socher et al., 2013). The resulting positive and negative triplets are treated as the final prediction triplet sets of the training and inference KGs, $\mathcal{T}_{\text{tr}}$ and $\mathcal{T}_{\text{inf}}$, respectively.

Following GraIL (Teru et al., 2020), we extract 2-hop enclosing subgraphs for all positive and negative triplets. To limit the size of extracted subgraphs, we limit the maximum number of neighbors for each hop to 50. The initial embedding vectors are generated by the double radius vertex labeling of GraIL.

When we compute the optimal transport distance with respect to the relational tree distance in Definition 4.2, we use the Sinkhorn algorithm (Cuturi, 2013), a fast approximation of optimal transport distance, due to the limit of computing time. We use GeomLoss (Feydy et al., 2019) for implementing the Sinkhorn algorithm on GPU. During the computation of the optimal transport distance with respect to the RTMD in Definition 4.3, we compute the exact solution using the POT library (Flamary et al., 2021).

We use Python 3.8 and Pytorch 1.13.0 with cudatoolkit 11.7 to implement SMPNNs. In our experiments, we tune the learning rate from {5e-4, 1e-3, 5e-3, 1e-2} by measuring the empirical risk on the training graph every epoch. The combination of the learning rate and the epoch with the lowest empirical risk is chosen. We run all models for 1000 epochs. To compute the generalization error, we use a margin of 0.5, i.e., $\gamma = 0.5$.

In the experiment in Section 6.1, we use 10% of all subgraphs from both training and inference KGs to train the support vector machine classifier and evaluate classification accuracies on the remaining subgraphs. We evaluate 5 times with different seeds: 1,2,3,4,5.

In the experiment analyzing the relationship between the scores computed by SMPNNs and RTMD in Section 6.2, we use the architecture of GraIL, as described in Section A. For WNv3, we use $d = 32$. For FBv1 and NLv2, we use $d = 64$. Also, for NLv2, we fix the norm of the weight matrices to 20 (Lee et al., 2024).

In the experiment analyzing the relationship between the SMPNN's stability and generalization error in Section 6.3, we use variations of three well-known SMPNNs: GraIL, RED-GNN, and NBFNet. First, we fix the *message*, *update*, and *readout* functions to those of GraIL while varying the *aggregation* and *global-readout* functions, selecting either a sum or mean aggregator. Additionally, to consider models that do not use the *global-readout* function, we design a variation where the output of the *global-readout* function is set to a zero vector. Second, we adopt the *message* and *update* functions from RED-GNN while applying the same variations as above. Last, we explore multiple instantiations of NBFNet by varying the *message* and *aggregation* functions. Specifically, we implement *message* functions based on TransE, DistMult, and RotatE, and design models using a sum aggregation, a max pooling, or a min pooling as *aggregation* function. We also use $\theta(l) = 0$ as the *history* function with a sum aggregation. Note that we retain NBFNet's *update*, *readout*, and *global-readout* functions.

For all models, we use $L = 2$ and $L = 3.$ resulting in 48 instantiations of SMPNNs. These variations include different *message*, *aggregation*, *update*, *global-readout*, *readout*, and *history* functions, and the number of layers. We then compare the stability and generalization error of these models. For WNv3 and FBv1, we use $d = 32$. For NLv2, we use $d = 64$. Also, for FBv1 and NLv2, we fix the norm of the weight matrices to 20. We filter out the models that are not well-trained, by setting a threshold for their empirical risk on the training graph. The threshold is 0.25 for WNv3, 0.2 for FBv1, and 0.05 for NLv2.