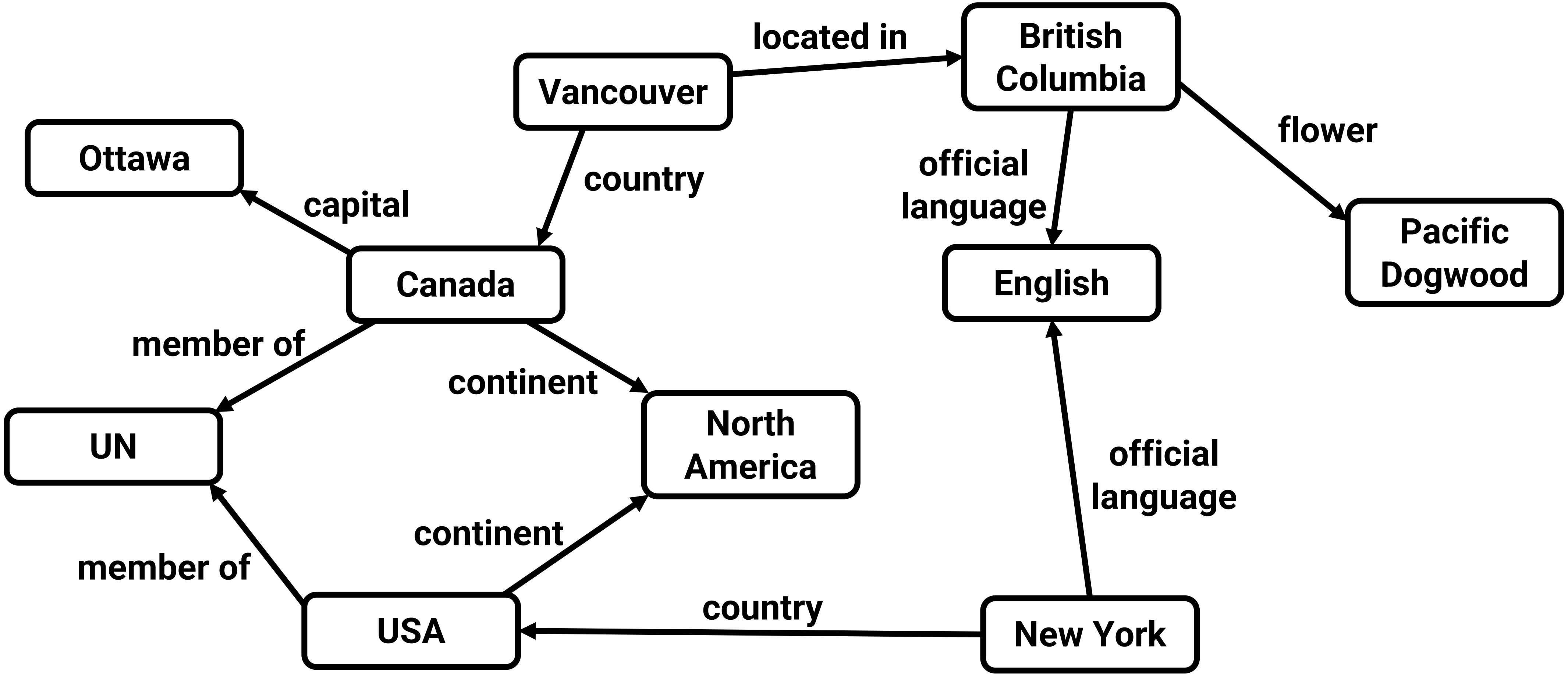
Stability and Generalization Capability of Subgraph Reasoning Models for Inductive Knowledge Graph Completion

Minsung Hwang, Jaejun Lee, and Joyce Jiyoung Whang*
School of Computing, KAIST
* Corresponding Author

The 42nd International Conference on Machine Learning (ICML 2025)

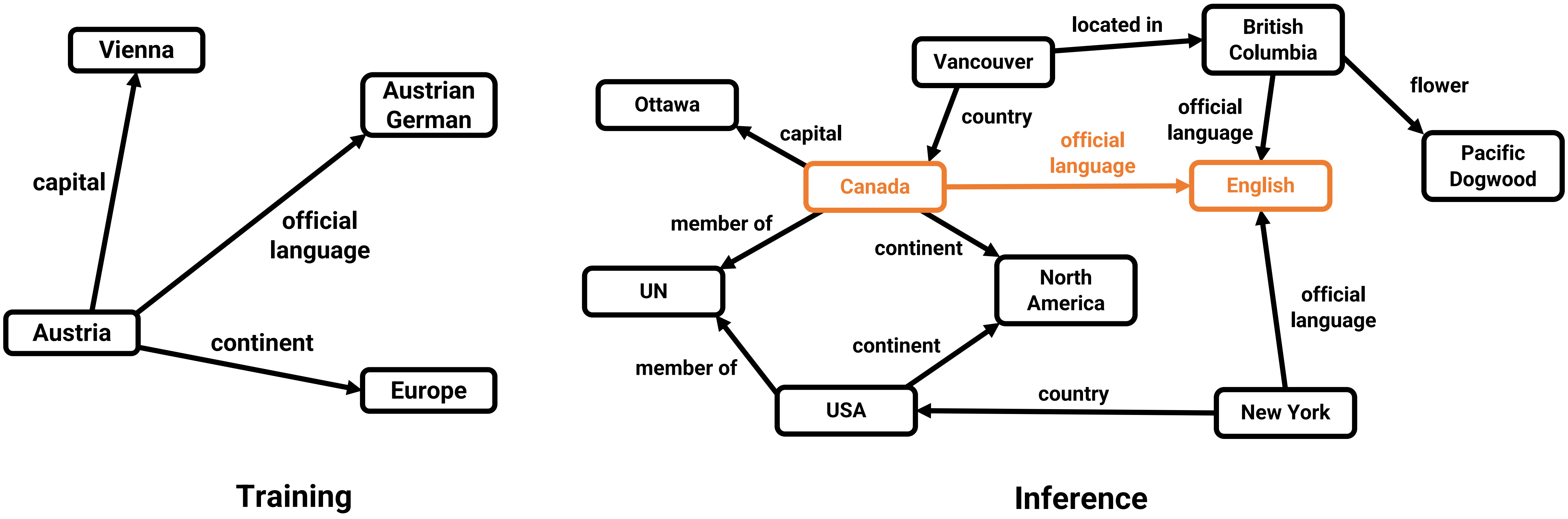
01 Knowledge Graph (KG)

- Represent **real-world knowledge** by modeling **relationships between entities**



01 Inductive Knowledge Graph Completion (KGC)

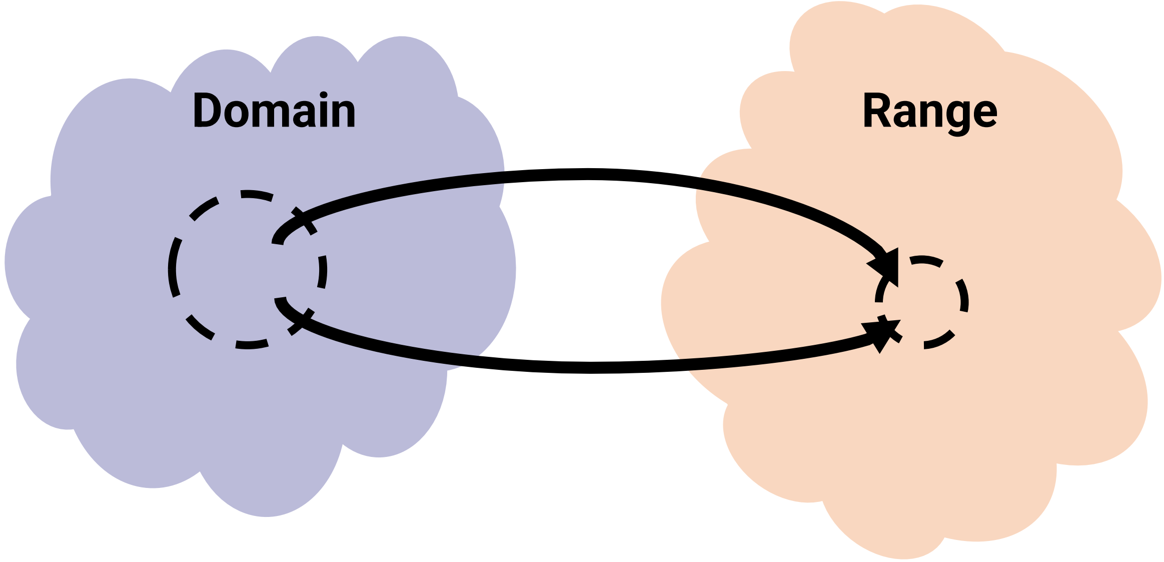
- **Predict missing triplets** with knowledge graphs
- KG that appears during **inference** differs from the one used for **training**



01 Theoretical Properties

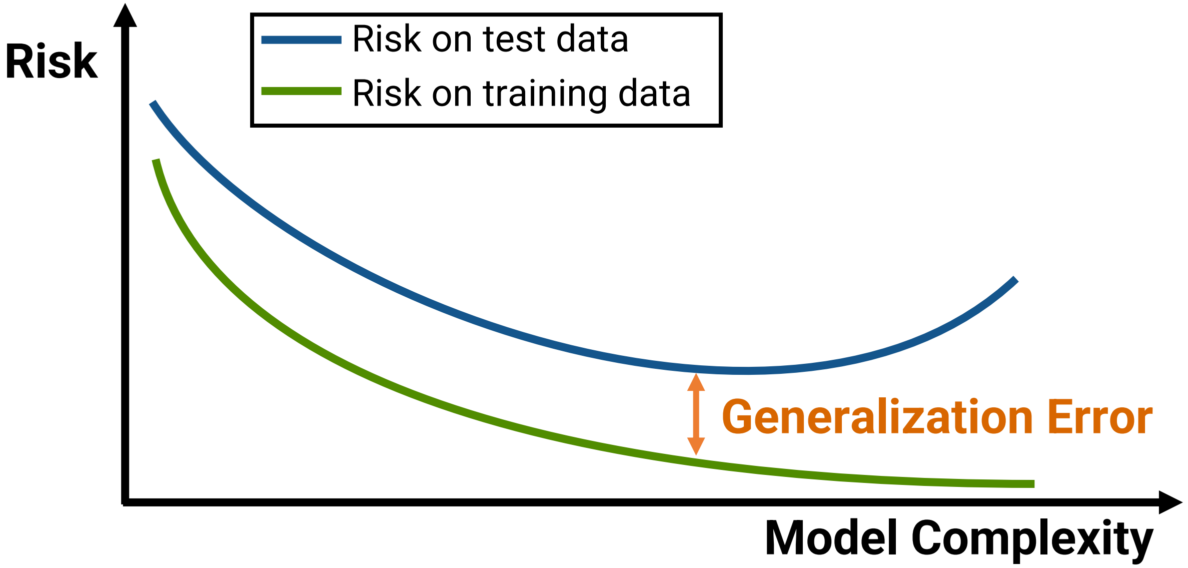
Stability

- **Consistency** of the model's output
- Measured by **Lipschitz constant**



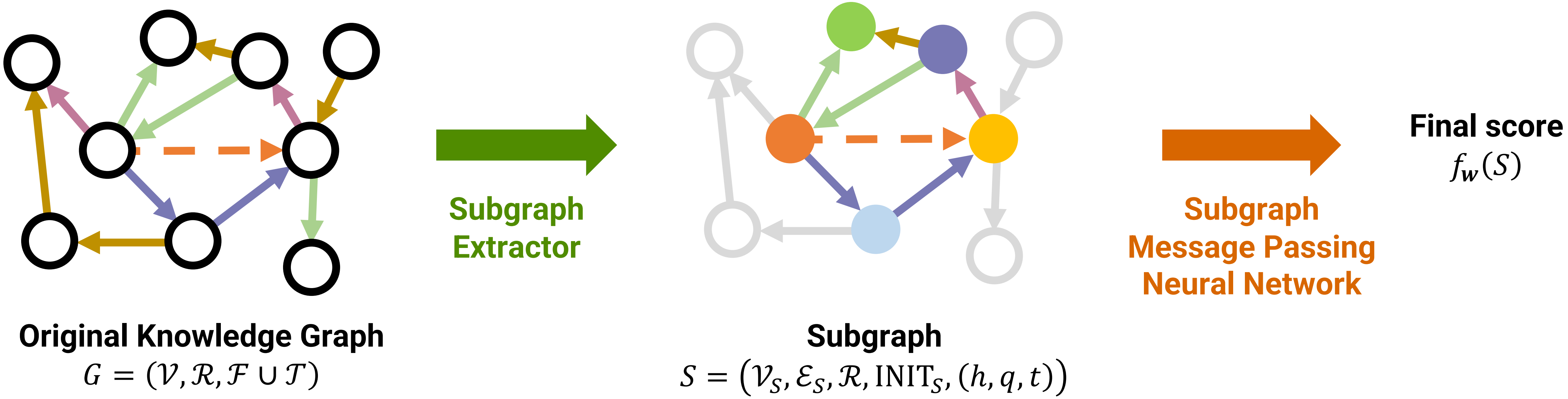
Generalization Capability

- Performance **discrepancy** between **training** and **test** data
- Measured by **generalization bound**



02 General Framework for Subgraph Reasoning Model

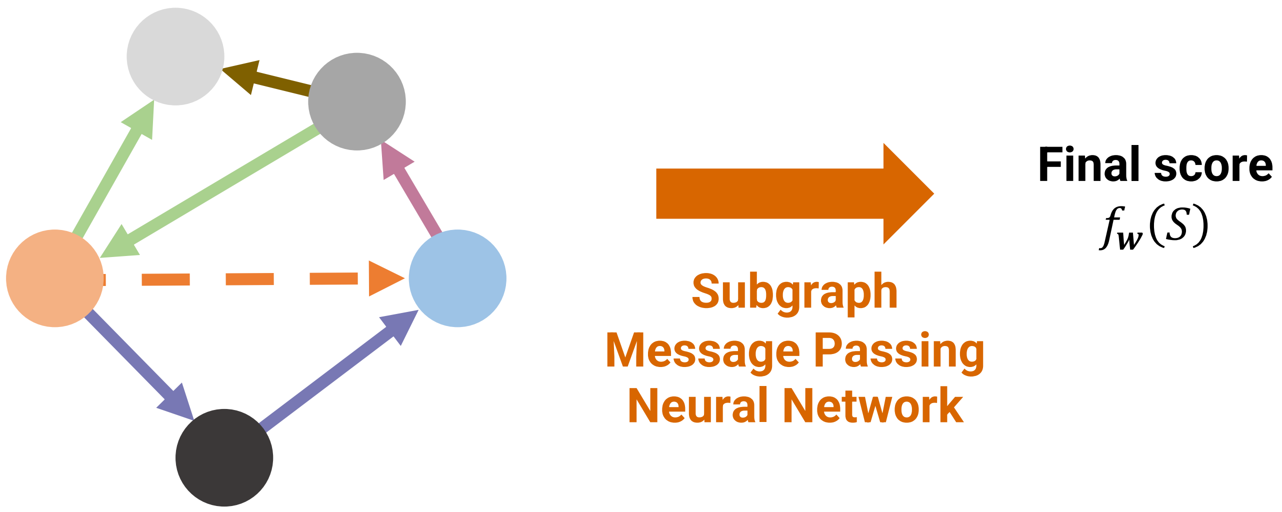
- Determine the validity of a triplet **using the subgraph** extracted around the triplet
  - **Extract a subgraph** associated with a target triplet
  - **Relabel the entities** within the subgraph
  - **Compute a score** of the subgraph through message-passing



02 Subgraph Message Passing Neural Network (SMPNN)

- Compute a score of the subgraph through **message-passing**

$$\boldsymbol{x}_S^{(0)}(v) = \mathrm{INIT}_S$$
$$\mathcal{M}_S^{(l)}(v) = \{\{\mathrm{MSG}^{(l)}\left(\boldsymbol{x}_S^{(l-1)}(u), \boldsymbol{x}_S^{(l-1)}(v), r, q\right) \mid (r,u) \in \mathcal{N}_S(v)\}\}$$
$$\boldsymbol{x}_S^{(l)}(v) = \mathrm{UPD}^{(l)}\left(\boldsymbol{x}_S^{(\theta(l))}(v), \mathrm{AGG}^{(l)}\left(\mathcal{M}_S^{(l)}(v)\right)\right)$$
$$f_w(S) = \mathrm{RD}\left(\boldsymbol{x}_S^{(L)}(h), \boldsymbol{x}_S^{(L)}(t), \mathrm{GRD}\left(\{\{\boldsymbol{x}_S^{(L)}(u) | u \in \mathcal{V}_S\}\}\right), q\right)$$



Subgraph Message Passing Neural Network (SMPNN)

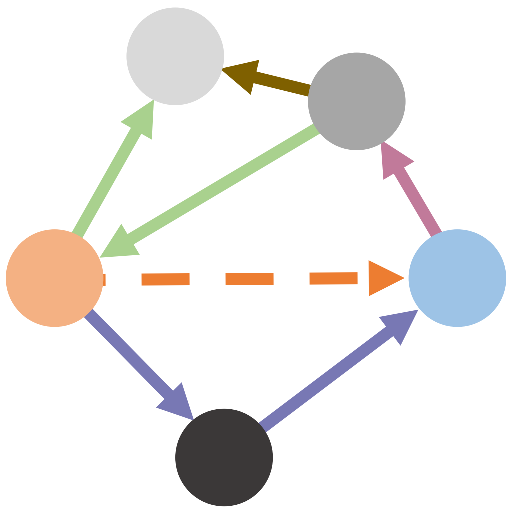
- **Initialize embedding vectors** using embedding vectors from INIT_S

$$\boldsymbol{x}_S^{(0)}(v) = \mathrm{INIT}_S$$

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

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

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



Subgraph Message Passing Neural Network

Final score
$f_{\boldsymbol{w}}(S)$

02 Subgraph Message Passing Neural Network (SMPNN)

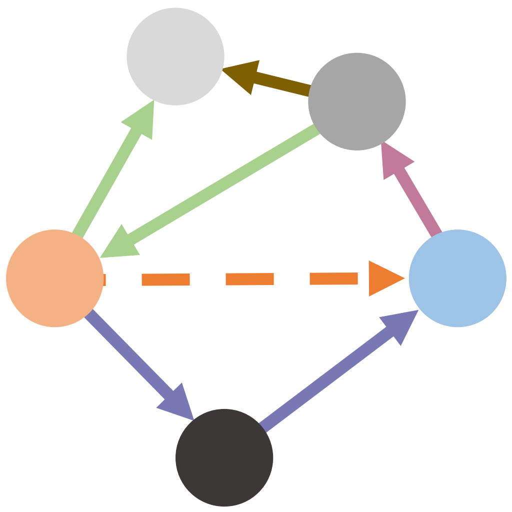
- **Compute messages** of all neighbors for each node

$$\boldsymbol{x}_S^{(0)}(v) = \mathrm{INIT}_S$$

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

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

$$f_w(S) = \mathrm{RD}\left(\boldsymbol{x}_S^{(L)}(h), \boldsymbol{x}_S^{(L)}(t), \mathrm{GRD}\left(\{\{\boldsymbol{x}_S^{(L)}(u) | u \in \mathcal{V}_S\}\}\right), q\right)$$



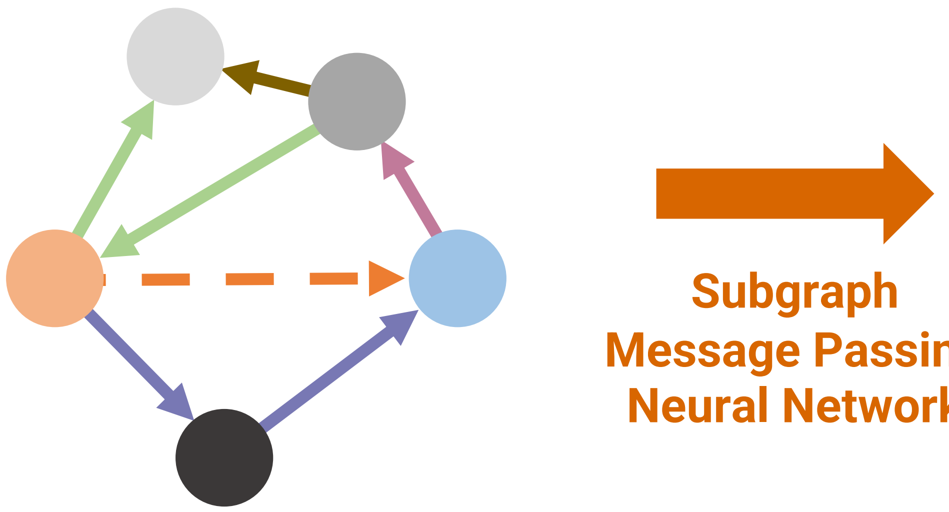
Subgraph Message Passing Neural Network

Final score $f_w(S)$

02 Subgraph Message Passing Neural Network (SMPNN)

- Update embedding vectors by **aggregating messages** of neighbors

$$\boldsymbol{x}_S^{(0)}(v) = \mathrm{INIT}_S$$
$$\mathcal{M}_S^{(l)}(v) = \{\{\mathrm{MSG}^{(l)}\left(\boldsymbol{x}_S^{(l-1)}(u), \boldsymbol{x}_S^{(l-1)}(v), r, q\right) \mid (r,u) \in \mathcal{N}_S(v)\}\}$$
$$\boldsymbol{x}_S^{(l)}(v) = \mathrm{UPD}^{(l)}\left(\boldsymbol{x}_S^{(\theta(l))}(v), \mathrm{AGG}^{(l)}\left(\mathcal{M}_S^{(l)}(v)\right)\right)$$
$$f_w(S) = \mathrm{RD}\left(\boldsymbol{x}_S^{(L)}(h), \boldsymbol{x}_S^{(L)}(t), \mathrm{GRD}\left(\{\{\boldsymbol{x}_S^{(L)}(u) | u \in \mathcal{V}_S\}\}\right), q\right)$$



Subgraph Message Passing Neural Network

Final score
$f_w(S)$

02 Subgraph Message Passing Neural Network (SMPNN)

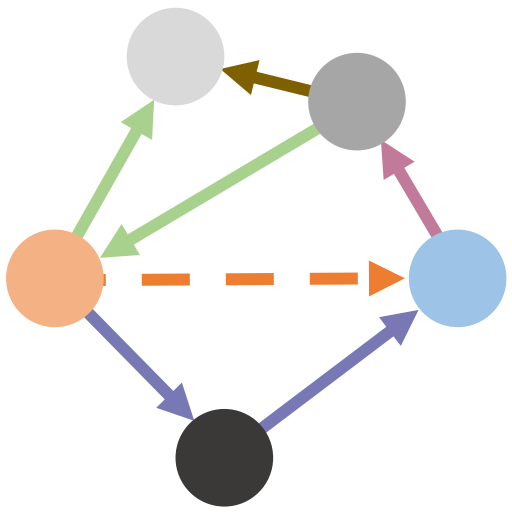
- **Update embedding vectors** by aggregating messages of neighbors

$$\boldsymbol{x}_S^{(0)}(v) = \mathrm{INIT}_S$$

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

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

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



Subgraph Message Passing Neural Network

Final score
$f_{\boldsymbol{w}}(S)$

02 Subgraph Message Passing Neural Network (SMPNN)

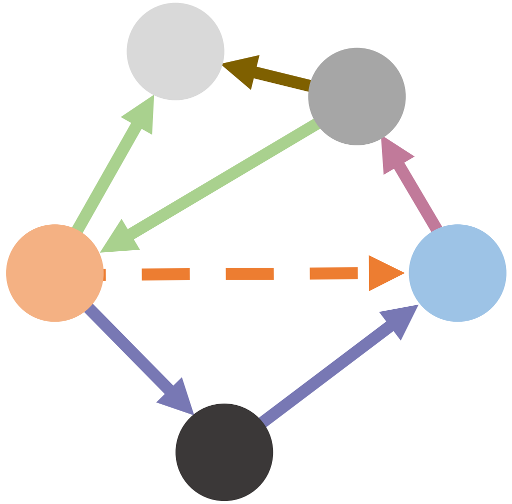
- Compute the final score using **readout and global-readout functions**

$$\boldsymbol{x}_S^{(0)}(v) = \mathrm{INIT}_S$$

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

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

$$f_w(S) = \mathrm{RD}\left(\boldsymbol{x}_S^{(L)}(h), \boldsymbol{x}_S^{(L)}(t), \mathrm{GRD}\left(\{\{\boldsymbol{x}_S^{(L)}(u) | u \in \mathcal{V}_S\}\}\right), q\right)$$



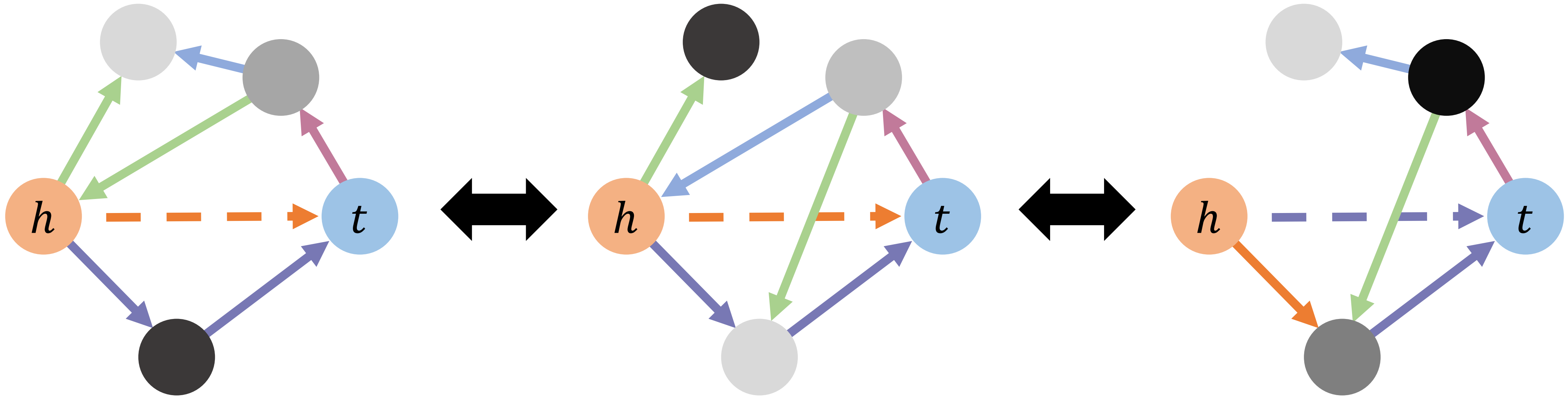
Subgraph Message Passing Neural Network

Final score

$f_w(S)$

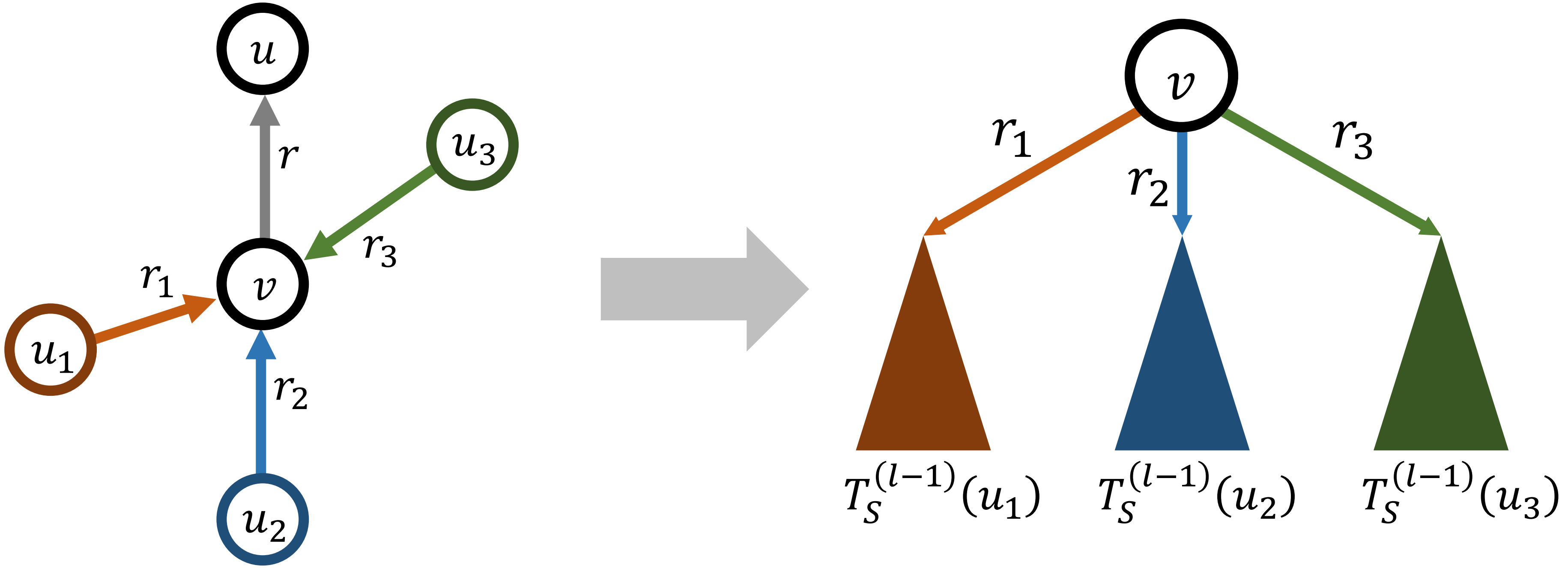
03 Relational Tree Mover's Distance (RTMD)

- Metric to **quantify differences** between subgraphs
  - RTMD reflects the **message-passing mechanism** of SMPNNs.



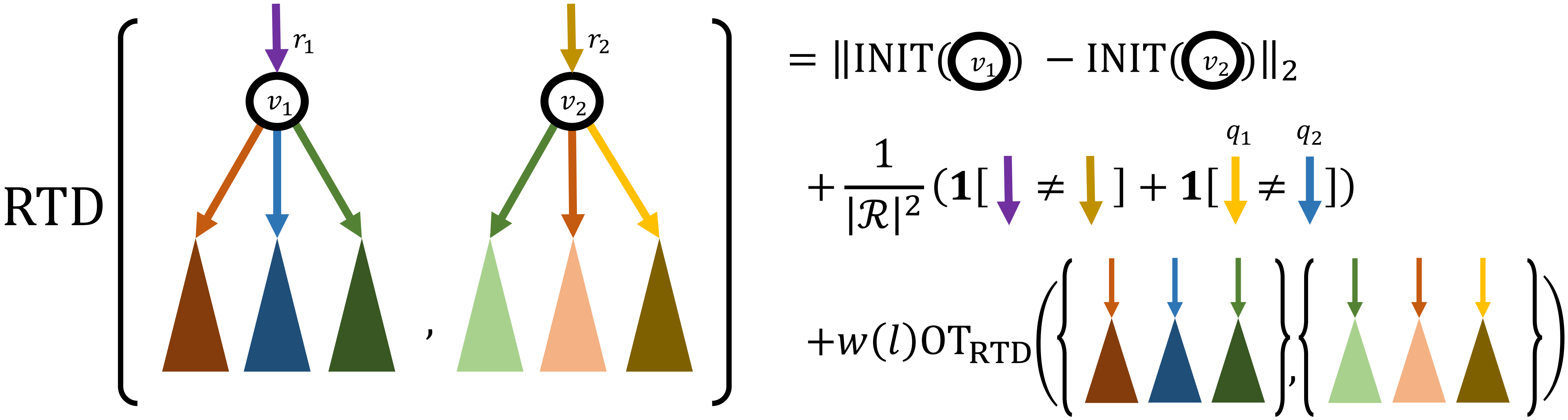
03 Relational Computation Tree

- Modeling how **SMPNNs process the subgraph structures**
  - Constructed by **recursively adding** neighboring **relations and entities** to leaf nodes



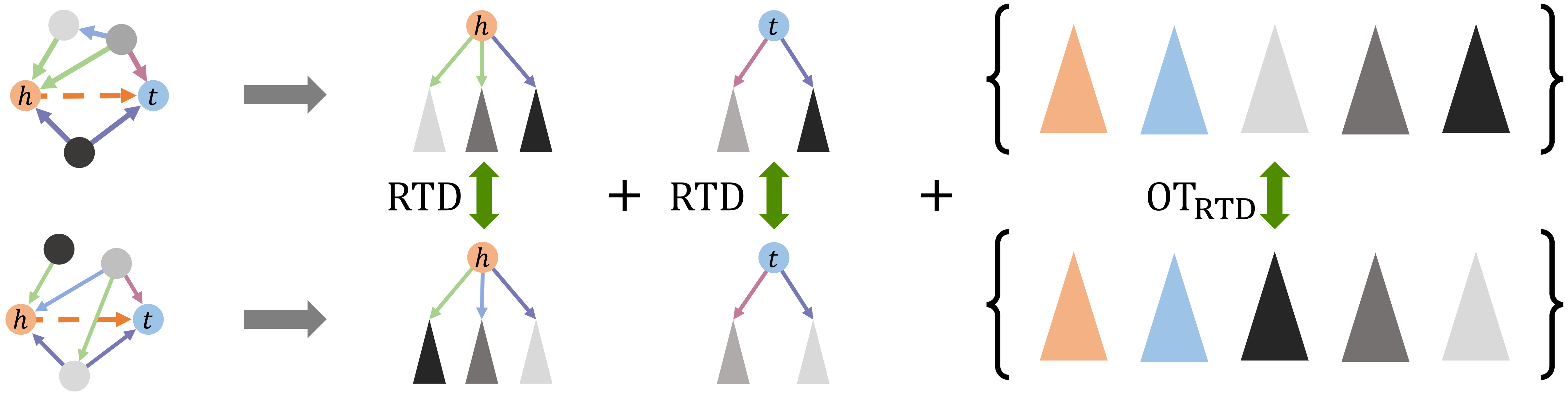
03 Relational Tree Distance (RTD)

- **Difference** between the **relational computation trees**
  - (1) The difference between the **initial embedding vectors** of their **root entities**
  - (2) The difference between the **sets of their subtree**
  - (3) Whether their **query relations** differ



03 Relational Tree Mover's Distance (RTMD)

- **RTMD** between two subgraphs
  - (1) The RTD between **the head entities** of the query triplet
  - (2) The RTD between **the tail entities** of the query triplet
  - (3) The difference between **the sets of relational computation trees**



03 Stability of SMPNNs

- Define **stability** C_f as the reciprocal of the **Lipchitz constant** η w.r.t **RTMD**
  - Bounded by the **Lipschitz constants of each function** of the SMPNNs.

Theorem 4.5 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_w with L layers, if the message, aggregation, update, global-readout, and readout function of f_w are Lipschitz continuous, then f_w is Lipschitz continuous with the **Lipschitz constant** η_f and the following holds:

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

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

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

where $1 \leq l \leq L$, and $\boldsymbol{A, B, C, D}$ are **the Lipschitz constants of the corresponding function**.

04 Risk of Subgraph Reasoning Model

- **γ-margin risk**
  - Increases when a score for a positive triplet is less than or equal to γ
  - Increases when a score for a negative triplet is greater than or equal to $-\gamma$

Empirical γ-margin risk

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

Expected γ-margin risk

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

- **Expected Risk Discrepancy**
  - Each risk is **measured on different KGs** in the **inductive setting**

Expected Risk Discrepancy

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

04 Generalization Bound of Subgraph Reasoning Models

- Using the **PAC-Bayesian approach**, compute the **generalization bound** of subgraph reasoning models
  - **Key terms**: **KL divergence** / **Expected risk discrepancy**
- **Expected Risk Discrepancy**
  - Each risk is **measured on different KGs** in the **inductive setting**

Theorem 5.3 Given G_{tr}, G_{inf}, and a subgraph reasoning model with a subgraph extractor g and an SMPNN $f_{\boldsymbol{w}}$, for any prior distribution $\mathcal{P}$ and posterior distribution $\mathcal{Q}$ on the parameter space of $f_{\boldsymbol{w}}$ constructed by adding random noise $\ddot{\boldsymbol{w}}$ to $\boldsymbol{w}$ such that $\mathbb{P}\left(\max\left(\max_{e\in\mathcal{T}_{\text{tr}}}\left|f_{\widetilde{\boldsymbol{w}}}\big(g(G_{\text{tr}},e)\big)-f_{\boldsymbol{w}}\big(g(G_{\text{tr}},e)\big)\right|,\max_{e\in\mathcal{T}_{\text{inf}}}\left|f_{\widetilde{\boldsymbol{w}}}\big(g(G_{\text{inf}},e)\big)-f_{\boldsymbol{w}}\big(g(G_{\text{inf}},e)\big)\right|\right)\right)$, and $\gamma,\lambda>0$, the following holds with probability at least $1-\delta$

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

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

04 Upper Bound of Expected Risk Discrepancy

- To focus on the discrepancy between two KGs, derive **the upper bound of the expected risk discrepancy**
  - $\mathrm{OT}_{\mathrm{RTMD}}\big(\psi(\mathcal{T}_{\mathrm{inf}}, \mathcal{T}_{\mathrm{tr}})\big)$: **Optimal transport distance** between the sets of subgraphs
  - C_f: **Stability**(=inverse of Lipschitz constant) of the subgraph reasoning model
    - Stability C_f is **inversely proportional** to the upper bound of the expected risk discrepancy

Theorem 5.5 Given G_{tr}, G_{inf}, and a subgraph reasoning model with a subgraph extractor g and an SMPNN f_w with stability C_f, for any prior distribution $\mathcal{P}$ and posterior distribution $\mathcal{Q}$ on the parameter space of f_w, and $\lambda > 0$, the following holds:

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

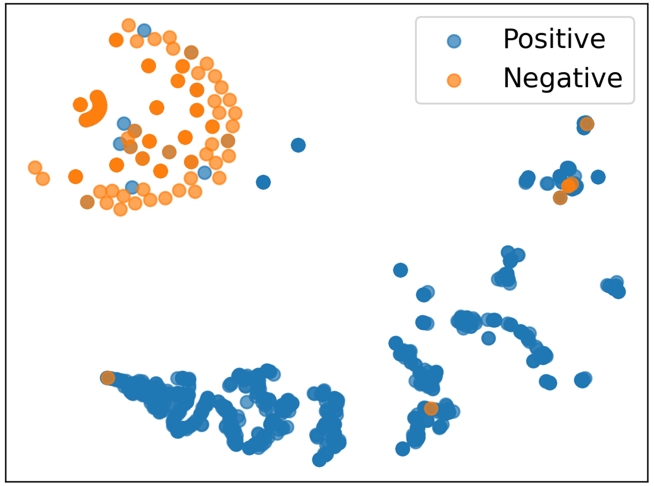
05 Experiments

- **Empirically validate** our theoretical findings
  - Demonstrate that **RTMD is a valid metric** for quantifying differences between subgraphs
  - Demonstrate that **SMPNNs are Lipschitz continuous w.r.t. RTMD**
  - Show that a **more stable model** tends to exhibit **better generalization capability**

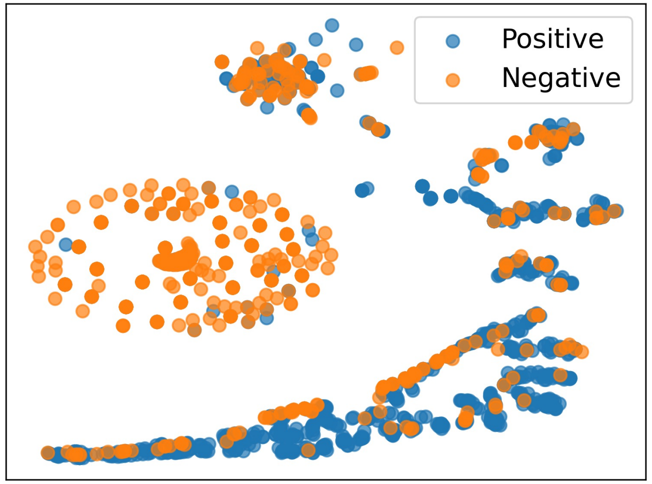
- **Datasets**
  - Benchmark datasets for **inductive KGC** provided in GraIL (ICML 2020)
  - **v3 of WN18RR / v1 of FB15K-237 / v2 of NELL-995**
  - Extract 2-hop subgraphs for each dataset

05 Label Classification using RTMD

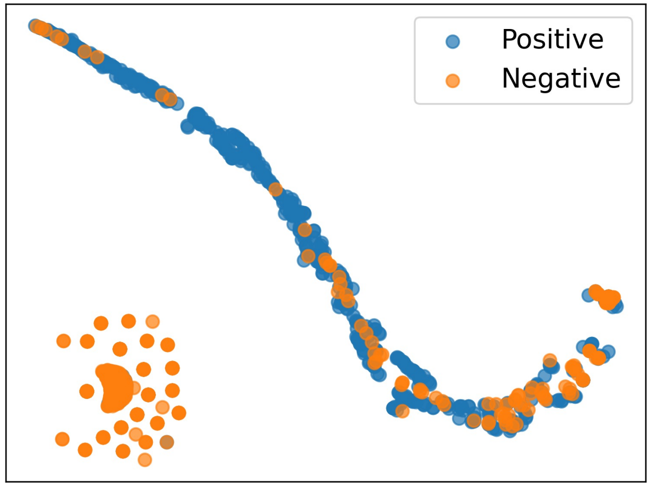
- Demonstrate that **RTMD is a valid metric** for quantifying differences between subgraphs
  - **tSNE visualization**: Distance between points is proportional to the RTMD.



WNv3
Classification accuracy: 0.8205



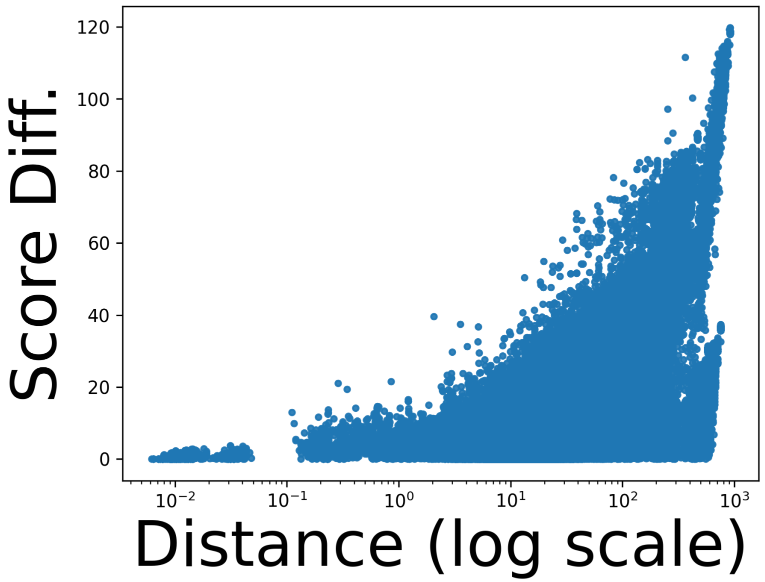
FBv1
Classification accuracy: 0.7739



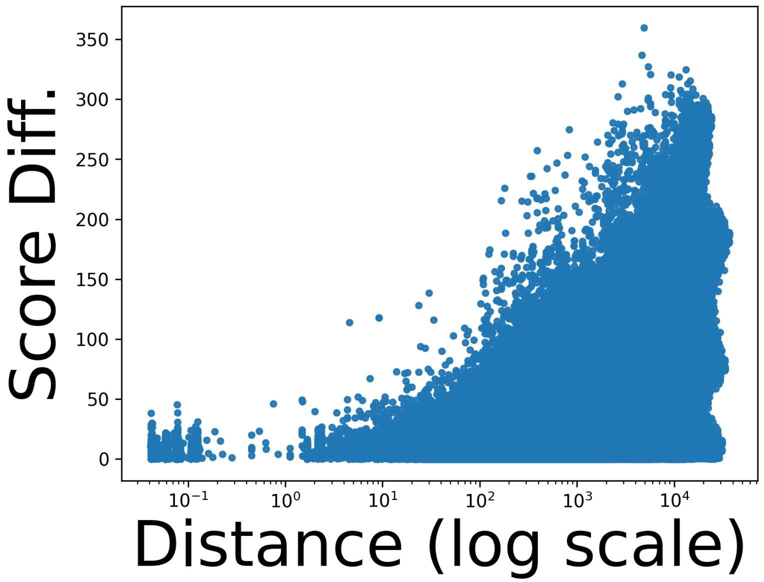
NLv2
Classification accuracy: 0.8654

05 Comparing RTMD with Scores

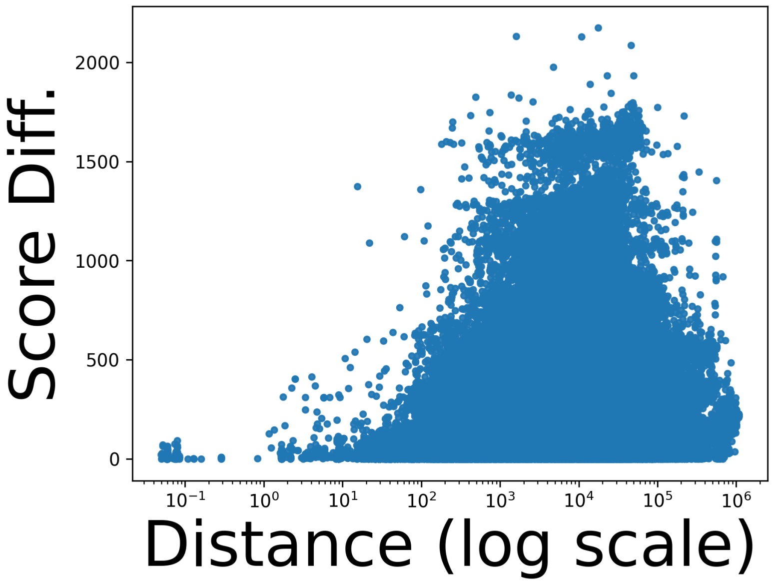
- Demonstrate that **SMPNNs are Lipschitz continuous w.r.t. RTMD**
  - Compare the **score difference** and the **RTMD** between subgraphs
  - Scores are computed by GraIL (ICML 2020)



WNv3



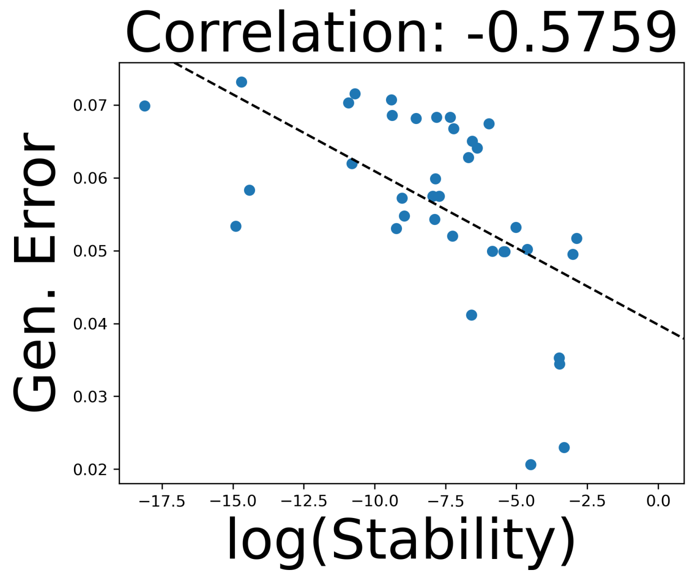
FBv1



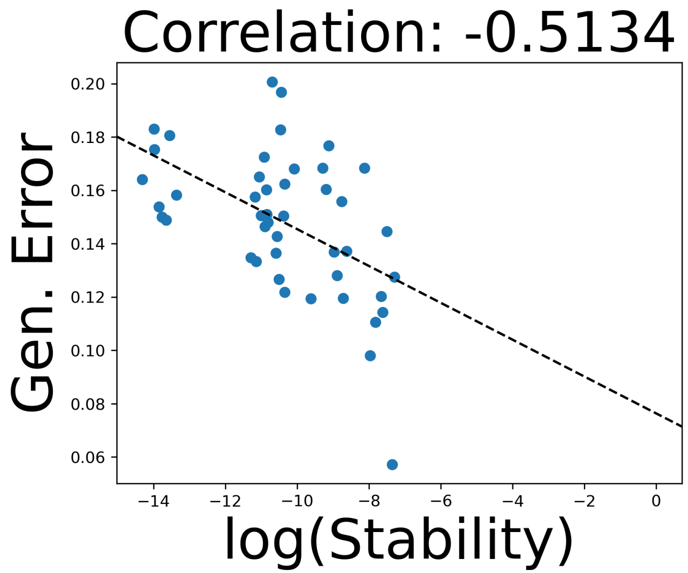
NLv2

05 Comparing Stability with Generalization Errors

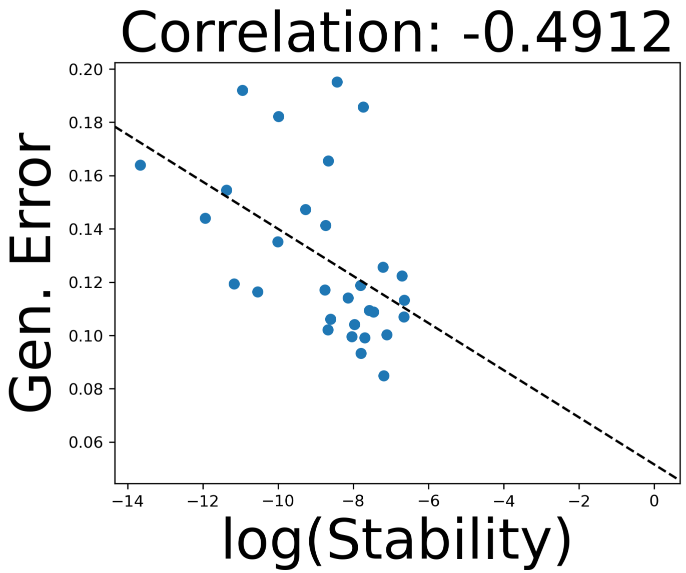
- Show that a **more stable model** tends to exhibit **better generalization capability**
  - Design **48 different subgraph reasoning models** by permuting the functions of SMPNNs
  - Compute the **empirical Lipschitz constant** of each model



WNv3
p-value: 0.00019



FBv1
p-value: 0.00031



NLv2
p-value: 0.00584

06 Conclusion

- Design a **general framework** for subgraph reasoning models
  - **Derive their stability** w.r.t the perturbations of the subgraph structures
- Introduce the **RTMD**, designed for subgraph reasoning models
  - **Use RTMD to compute the stability** of subgraph reasoning models
- **Theoretically analyze** the subgraph reasoning models for inductive KGC
  - Discuss the **impact of the stability** on their **generalization capability**
- **Empirically validate** that our theoretical findings hold on real-world KGs
  - Compare the **stability** and **generalization error** of the subgraph reasoning models

Thank You!

BDILab

You can find us at:

{hminsung, jjlee98, jjwhang}@kaist.ac.kr

https://bdi-lab.kaist.ac.kr