PAC-Bayesian Generalization Bounds for Knowledge Graph Representation Learning

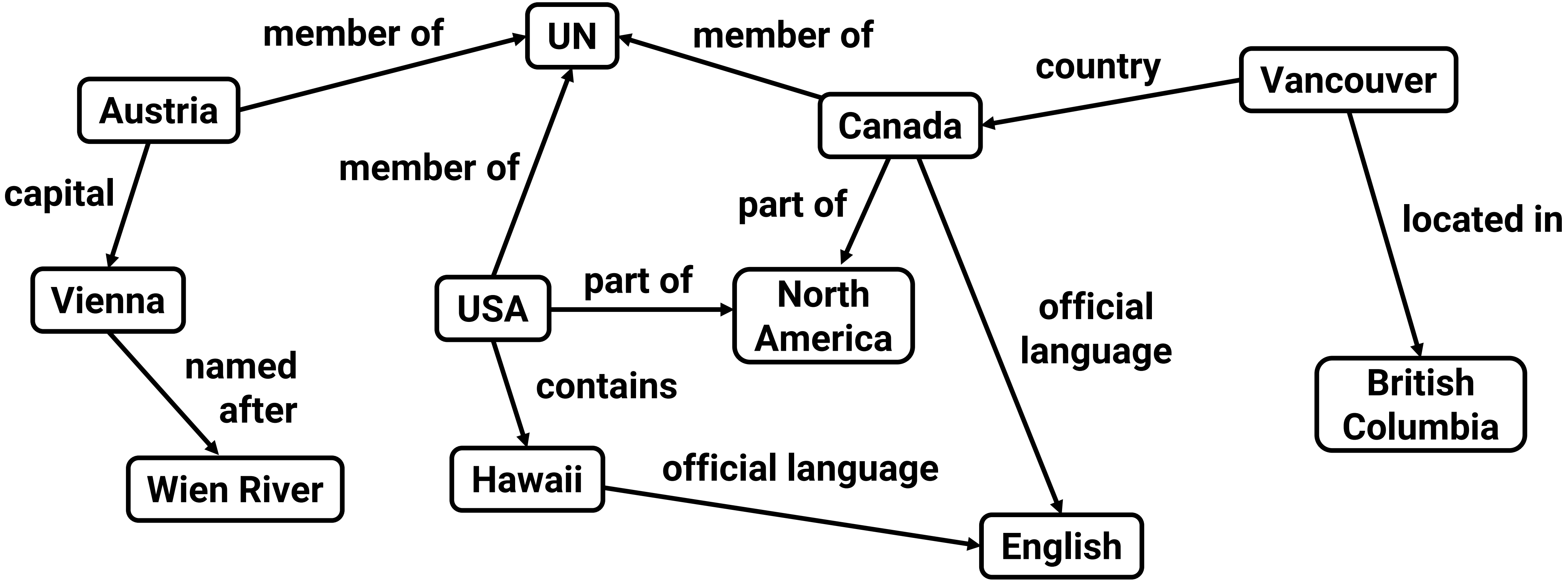
Jaejun Lee, Minsung Hwang, and Joyce Jiyoung Whang*
School of Computing, KAIST
* Corresponding Author

The 41st International Conference on Machine Learning
(ICML 2024)

KAIST BDILab
BIG DATA INTELLIGENCE

01 Knowledge Graphs

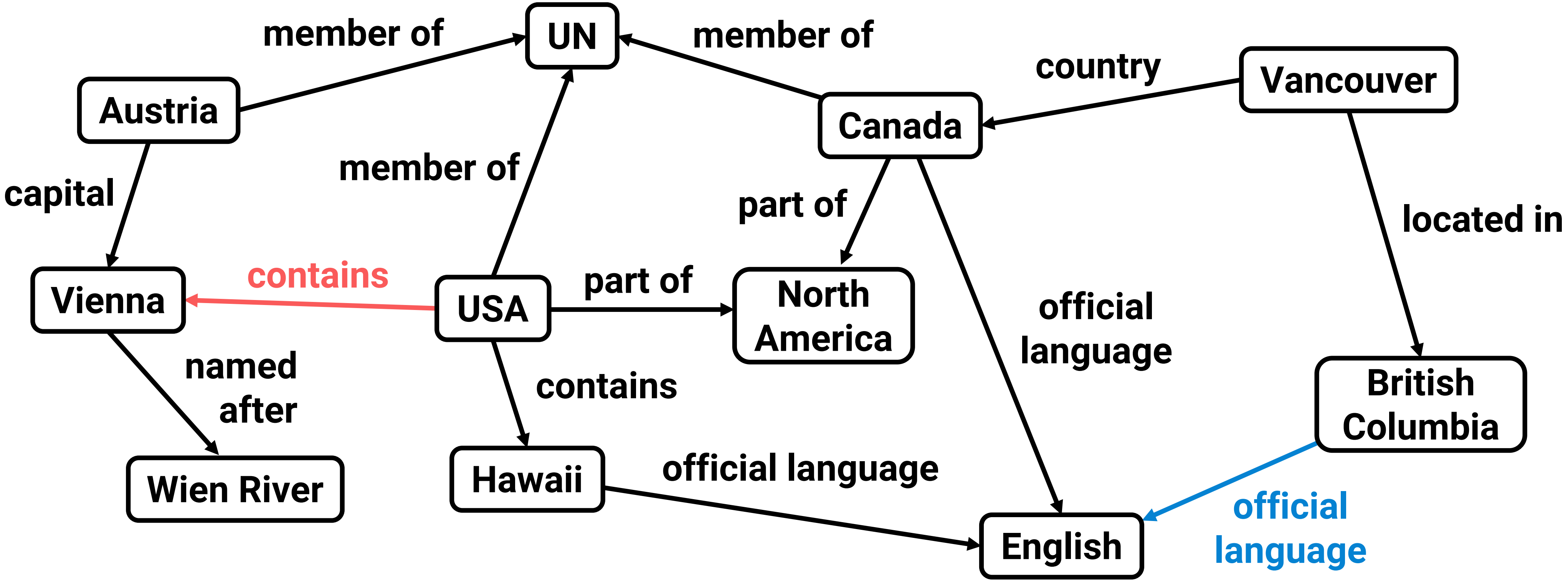
- Represent human knowledge using triplets



01 Triplet Classification on Knowledge Graphs

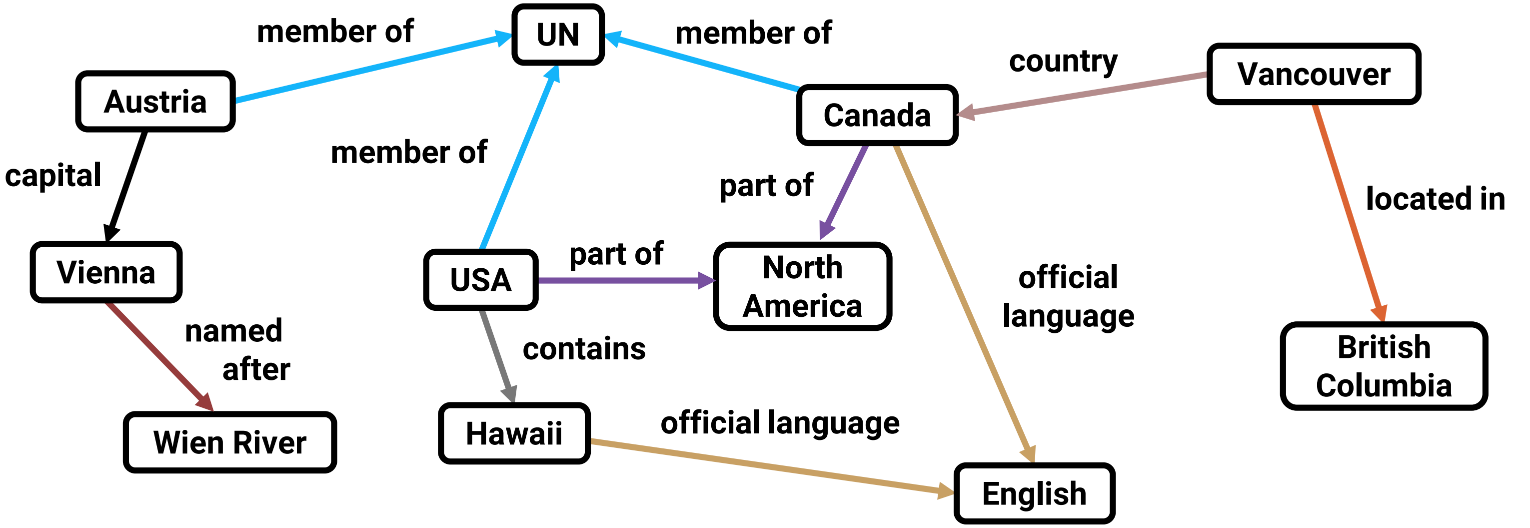
(USA, contains, Vienna) X

(British Columbia, official language, English) O

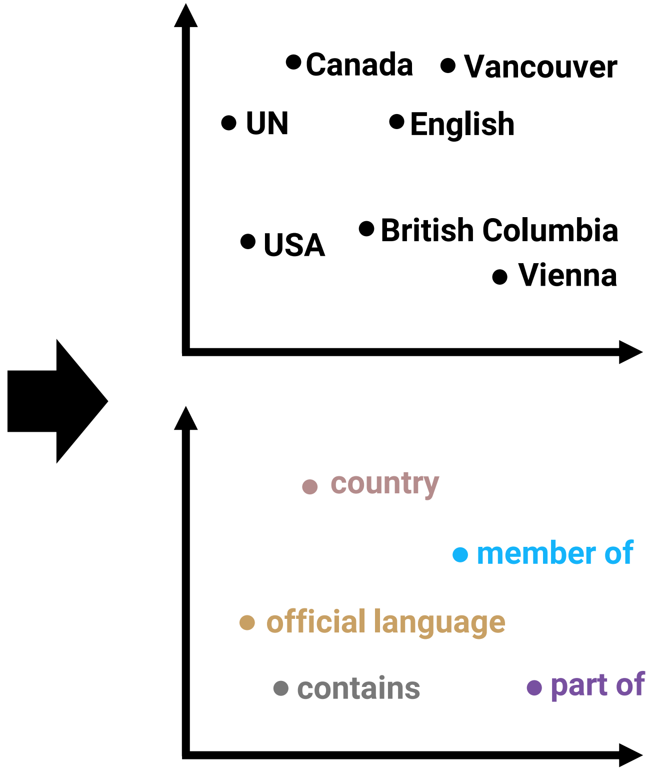


01 Knowledge Graph Representation Learning

- Learn representations of the entities and relations in a knowledge graph



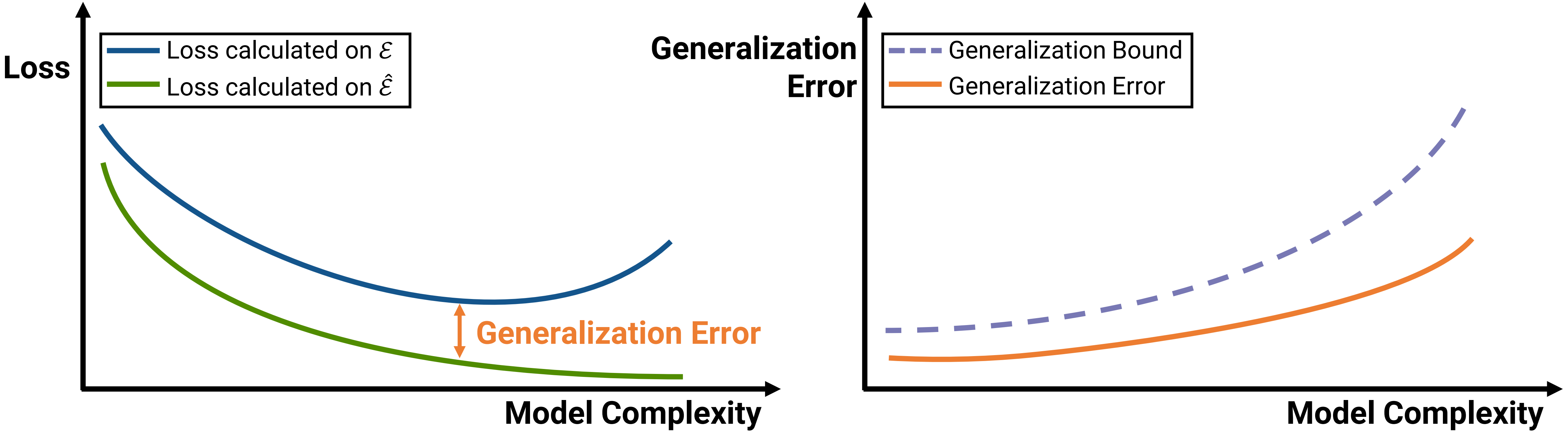
Knowledge Graph



Representations of Entities and Relations

01 Generalization Bound

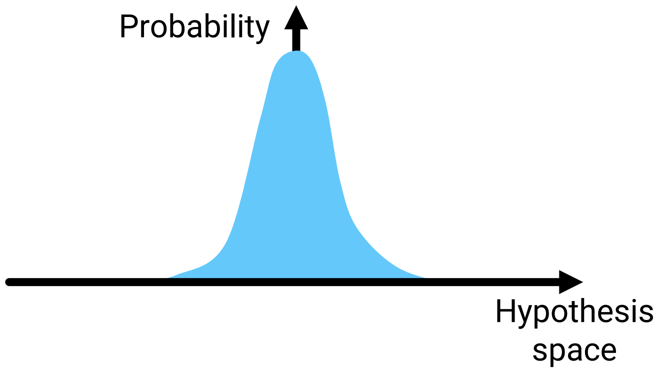
- **Generalization Error**
  - Difference between the losses calculated on the **full set** $\mathcal{E}$ and the **training set** $\hat{\mathcal{E}}$
- **Generalization Bound**
  - Theoretical upper bound of the generalization error



01 PAC-Bayesian Generalization Bounds

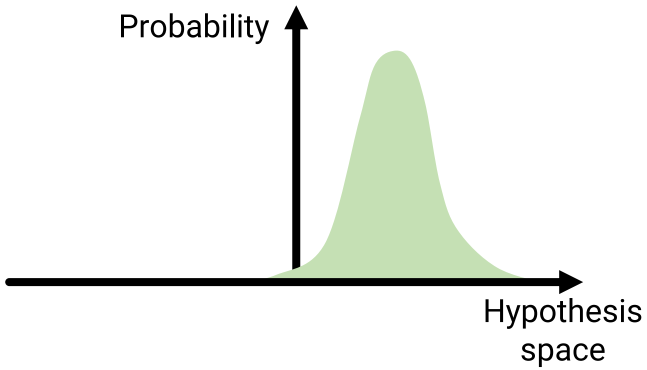
- **Probably Approximately Correct (PAC) theory**
  - Fundamental tools for analyzing the **generalization bounds**
- **PAC-Bayesian approach**
  - Measure generalization bounds based on the difference between the **prior** distribution and the **posterior** distribution

PAC theory
- **VC dimension**
- **Rademacher complexity**
- **PAC-Bayesian approach**
- ⋮



Prior distribution
Independent of the training set

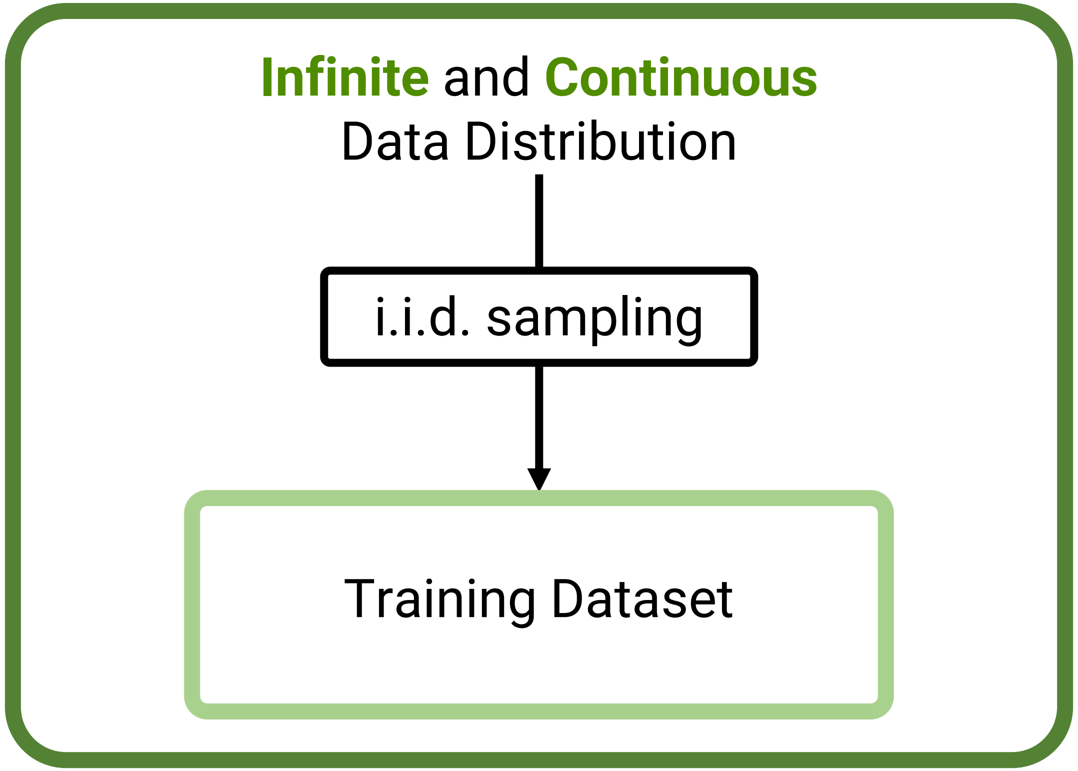
Training set → **Training**



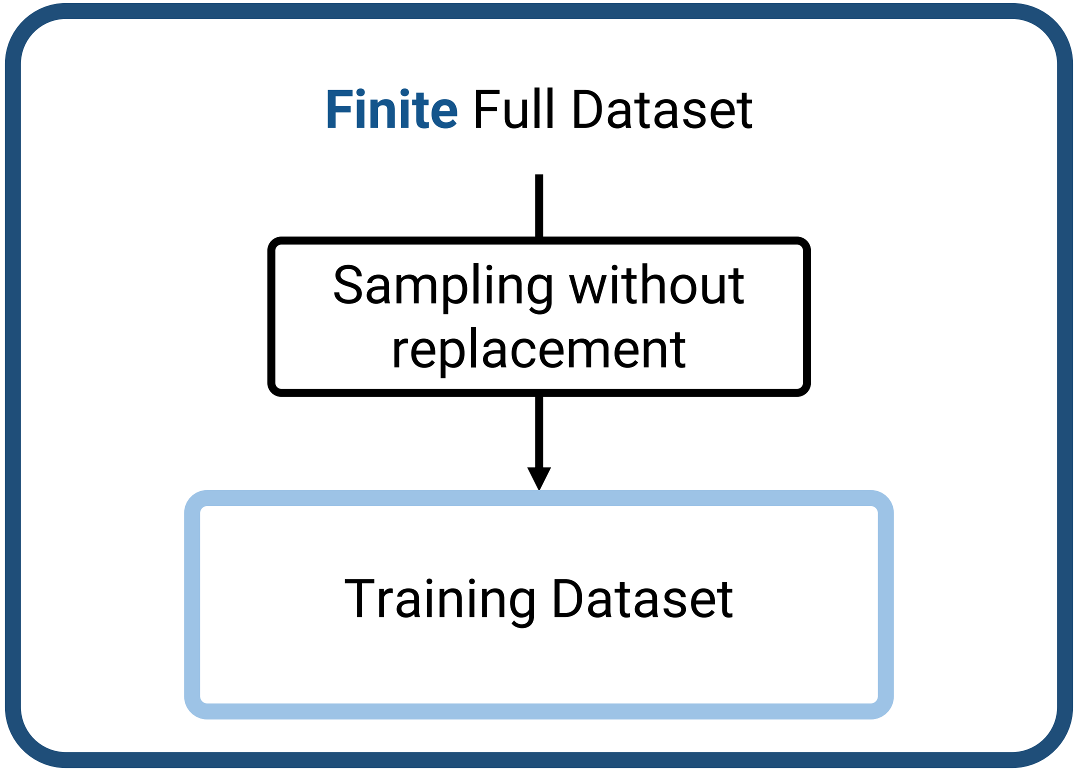
Posterior distribution
Learned by a learning algorithm

01 Transductive PAC-Bayesian Generalization Bounds

Original PAC-Bayesian framework



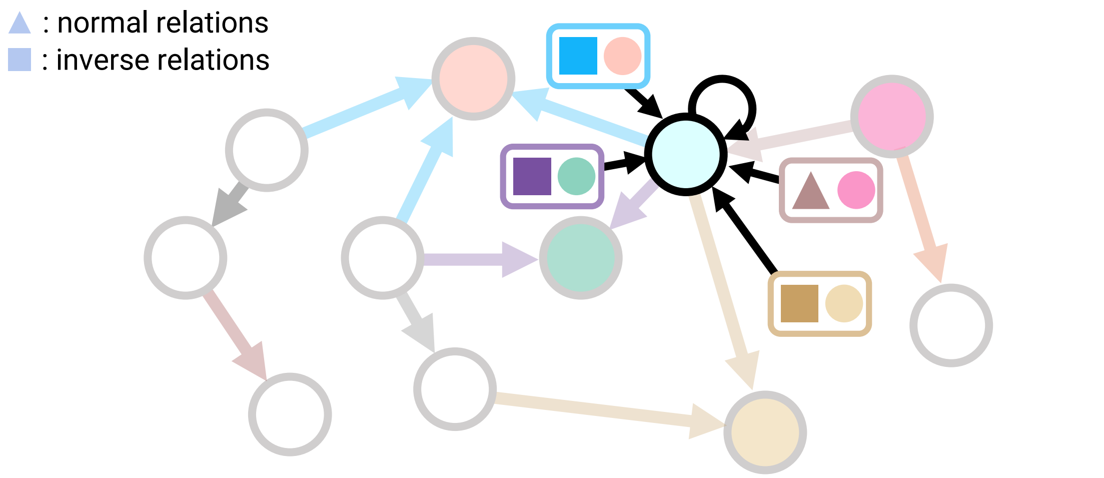
Transductive PAC-Bayesian framework



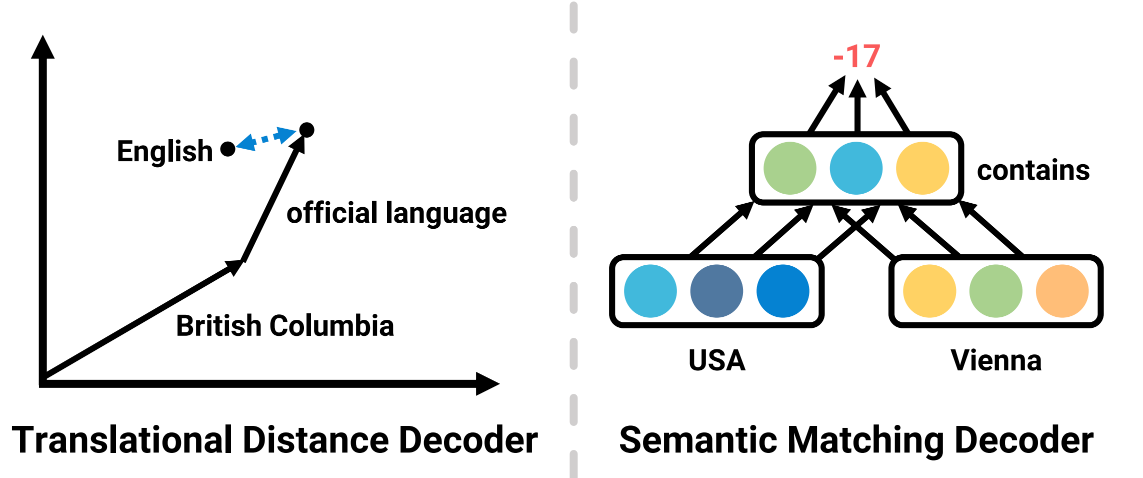
02 Relation-aware Encoder-Decoder Framework

- Consists of the **RAMP encoder** and a **triplet classification decoder**
  - RAMP encoder **learns the representations of entities** by aggregating representations of the neighboring entities and relations
  - Triplet classification decoder uses the representations to **compute the scores of each triplet**
  - Assigns **two different scores** for each triplet, stored in $f_{\mathbf{w}}(h,r,t)[0]$ and $f_{\mathbf{w}}(h,r,t)[1]$

Relation-Aware Message Passing Encoder

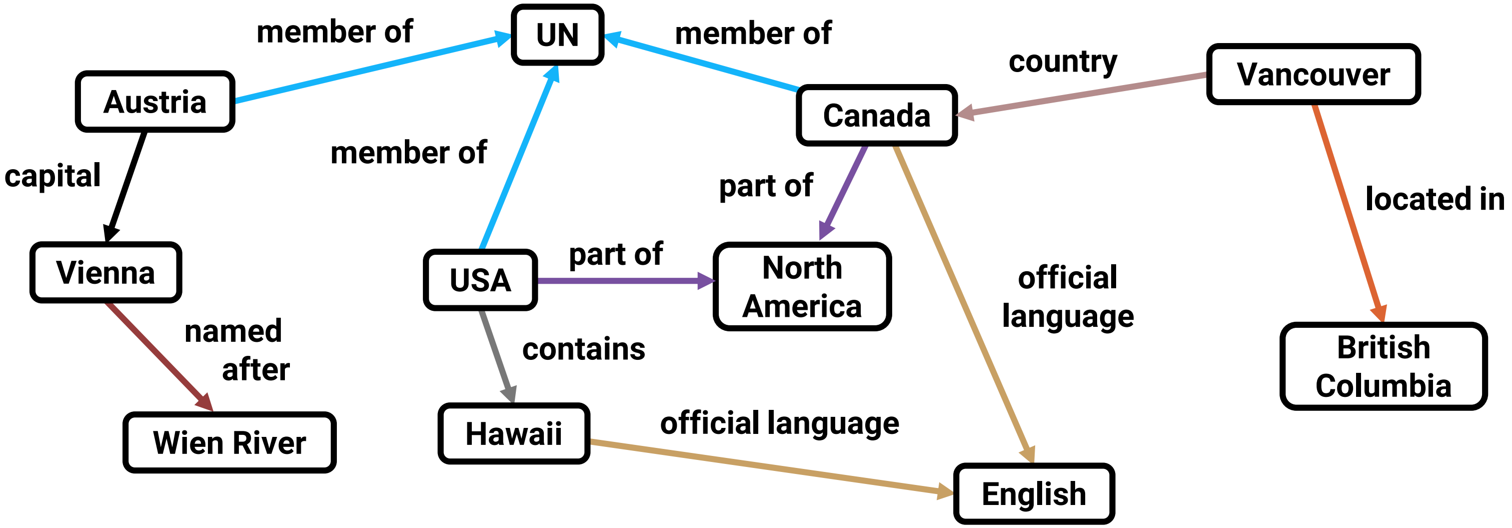


Triplet Classification Decoder



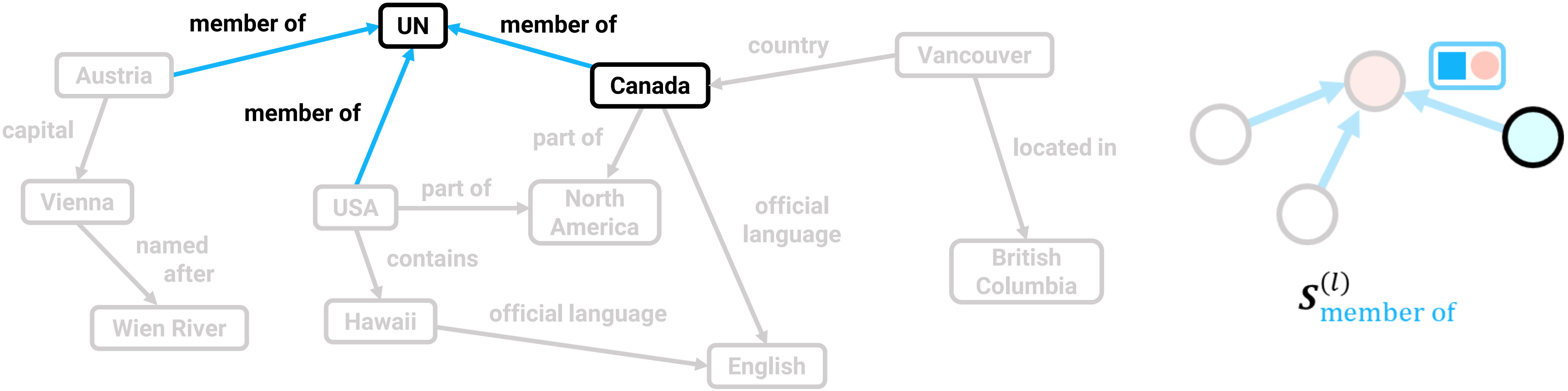
02 RAMP Encoder

- Update an entity's representation based on the **entity** and **relation** representations of its neighbors which are defined per relation using a relation-specific **graph diffusion matrix**



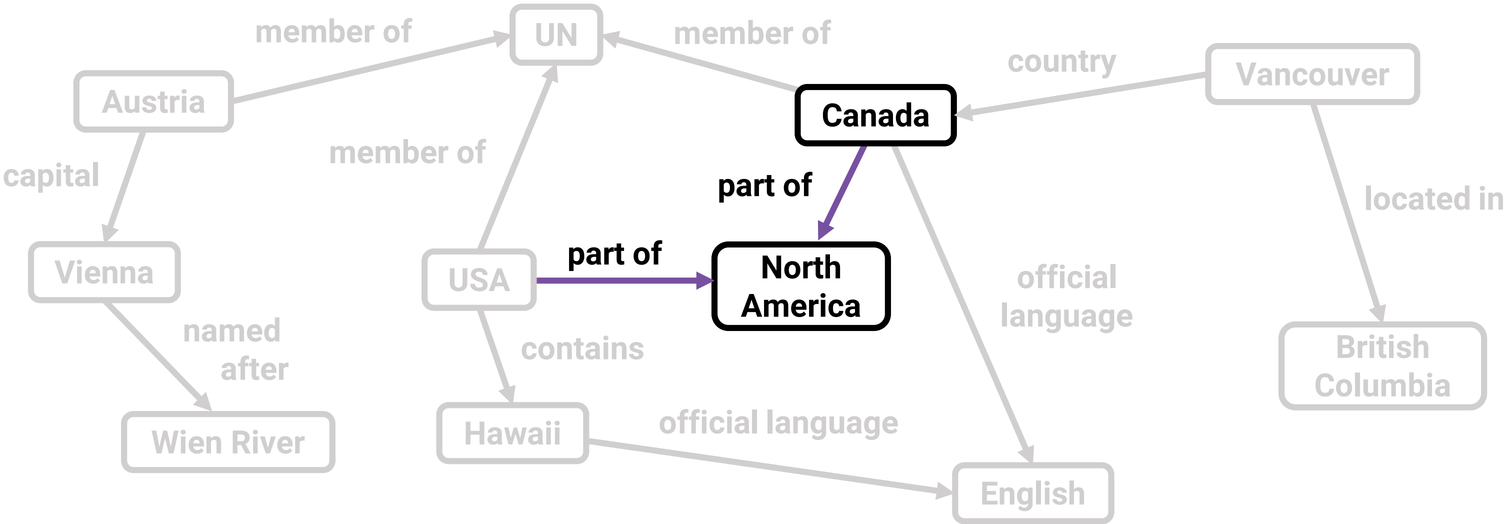
RAMP Encoder

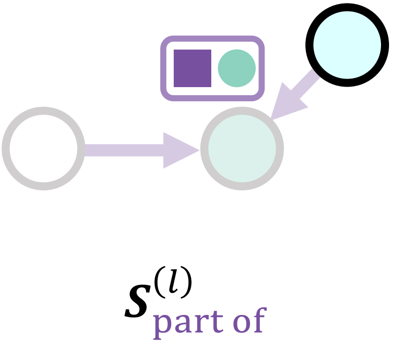
- Update an entity's representation based on the **entity** and **relation** representations of its neighbors which are defined per relation using a relation-specific **graph diffusion matrix**



02 RAMP Encoder

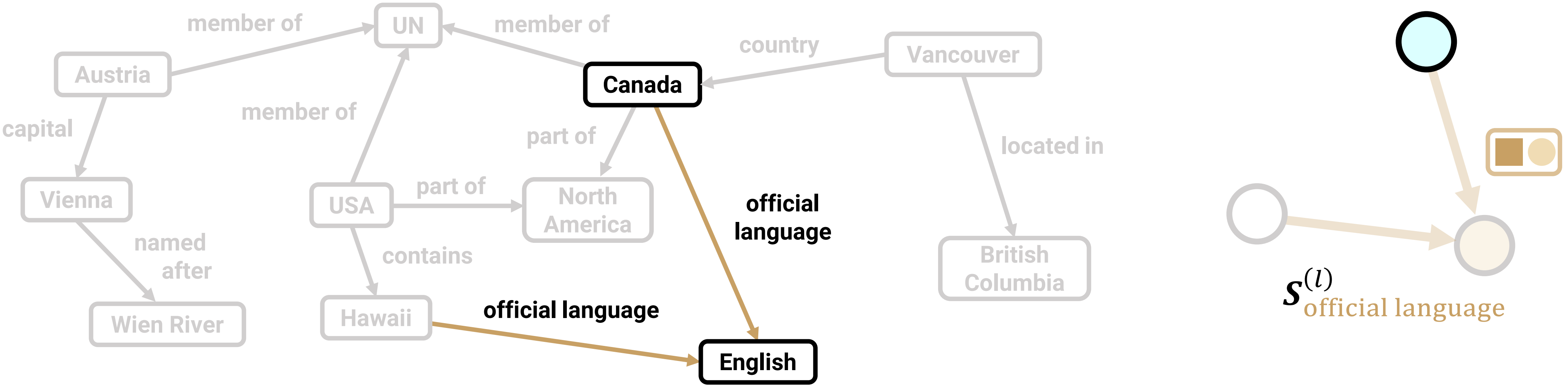
- Update an entity's representation based on the **entity** and **relation** representations of its neighbors which are defined per relation using a relation-specific **graph diffusion matrix**





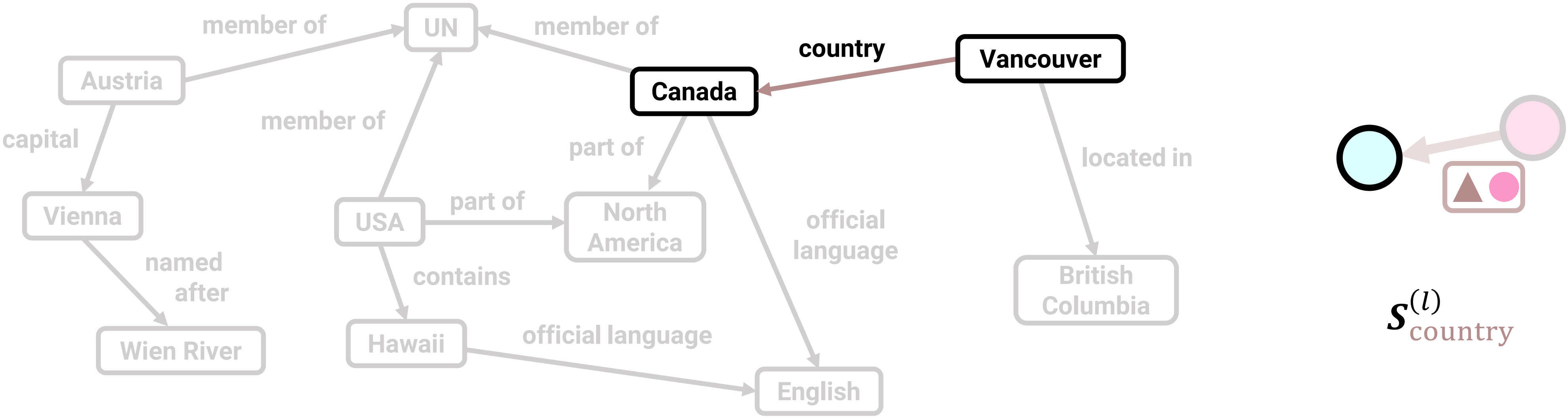
02 RAMP Encoder

- Update an entity's representation based on the **entity** and **relation** representations of its neighbors which are defined per relation using a relation-specific **graph diffusion matrix**



02 RAMP Encoder

- Update an entity's representation based on the **entity** and **relation** representations of its neighbors which are defined per relation using a relation-specific **graph diffusion matrix**



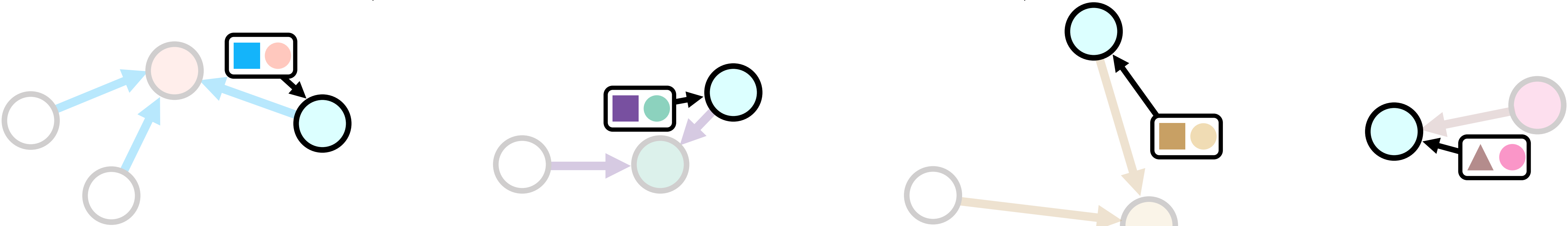
$s_{\text{country}}^{(l)}$

02 RAMP Encoder

- Update an entity's representation based on the **entity** and **relation** representations of its neighbors which are defined per relation using a relation-specific **graph diffusion matrix**

$$\boldsymbol{M}_r^{(l)}[v,:] = [\boldsymbol{H}^{(l-1)}[v,:] \quad \boldsymbol{R}^{(l-1)}[r,:]] \qquad v \in \mathcal{V}, r \in \mathcal{R}$$

$$\boldsymbol{H}^{(l)} = \phi\left(\boldsymbol{H}^{(l-1)}\boldsymbol{W}_0^{(l)} + \rho\left(\sum_{r\in\mathcal{R}} \boldsymbol{S}_r^{(l)} \psi\left(\boldsymbol{M}_r^{(l)}\right)\begin{bmatrix}\boldsymbol{W}_r^{(l)} \\ \boldsymbol{U}_r^{(l)}\end{bmatrix}\right)\right), \qquad \boldsymbol{R}^{(l)} = \boldsymbol{R}^{(l-1)}\boldsymbol{U}_0^{(l)}$$



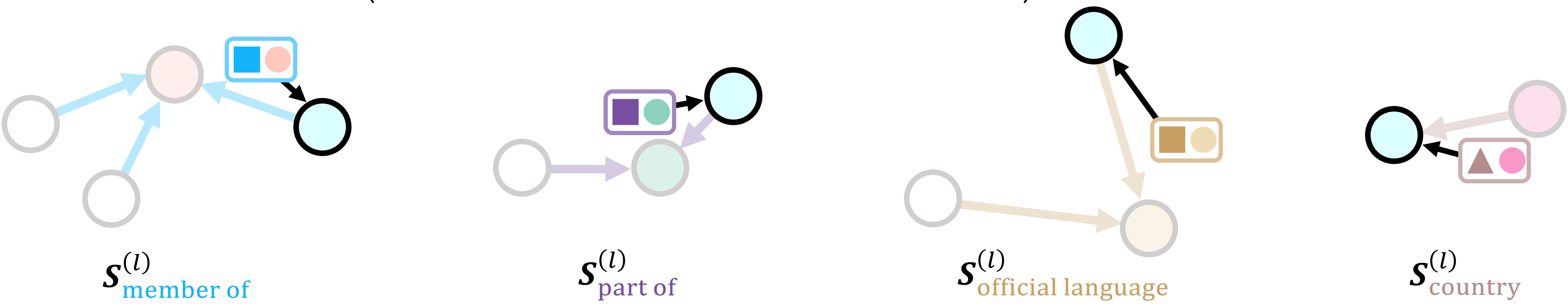
$\boldsymbol{S}^{(l)}_{\text{member of}}$ $\boldsymbol{S}^{(l)}_{\text{part of}}$ $\boldsymbol{S}^{(l)}_{\text{official language}}$ $\boldsymbol{S}^{(l)}_{\text{country}}$

RAMP Encoder

- Project the neighbor entities and relations' representations using **relation-specific projection matrices**
  - Different projection matrices for entities and relations

$$\boldsymbol{M}_r^{(l)}[v,:] = [\boldsymbol{H}^{(l-1)}[v,:] \quad \boldsymbol{R}^{(l-1)}[r,:]] \qquad v \in \mathcal{V}, r \in \mathcal{R}$$

$$\boldsymbol{H}^{(l)} = \phi\left(\boldsymbol{H}^{(l-1)}\boldsymbol{W}_0^{(l)} + \rho\left(\sum_{r\in\mathcal{R}} \boldsymbol{S}_r^{(l)}\psi\left(\boldsymbol{M}_r^{(l)}\right)\begin{bmatrix}\boldsymbol{W}_r^{(l)} \\ \boldsymbol{U}_r^{(l)}\end{bmatrix}\right)\right), \qquad \boldsymbol{R}^{(l)} = \boldsymbol{R}^{(l-1)}\boldsymbol{U}_0^{(l)}$$

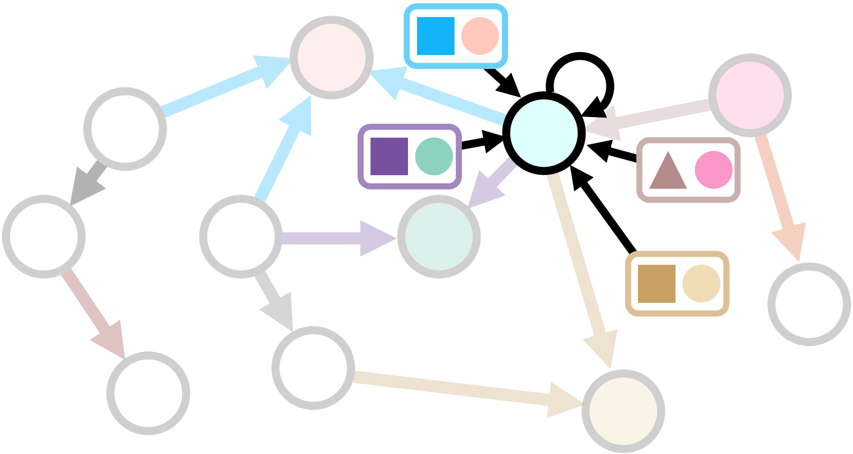


02 RAMP Encoder

- Aggregate the neighbor representations to update the entity's representation
  - The diffusion matrix also represents the type of aggregator (e.g., sum, mean)

$$\boldsymbol{M}_r^{(l)}[v,:] = [\boldsymbol{H}^{(l-1)}[v,:] \quad \boldsymbol{R}^{(l-1)}[r,:]] \qquad v \in \mathcal{V}, r \in \mathcal{R}$$

$$\boldsymbol{H}^{(l)} = \phi\left(\boldsymbol{H}^{(l-1)}\boldsymbol{W}_0^{(l)} + \rho\left(\sum_{r\in\mathcal{R}} \boldsymbol{S}_r^{(l)} \psi\left(\boldsymbol{M}_r^{(l)}\right)\begin{bmatrix}\boldsymbol{W}_r^{(l)} \\ \boldsymbol{U}_r^{(l)}\end{bmatrix}\right)\right), \qquad \boldsymbol{R}^{(l)} = \boldsymbol{R}^{(l-1)}\boldsymbol{U}_0^{(l)}$$

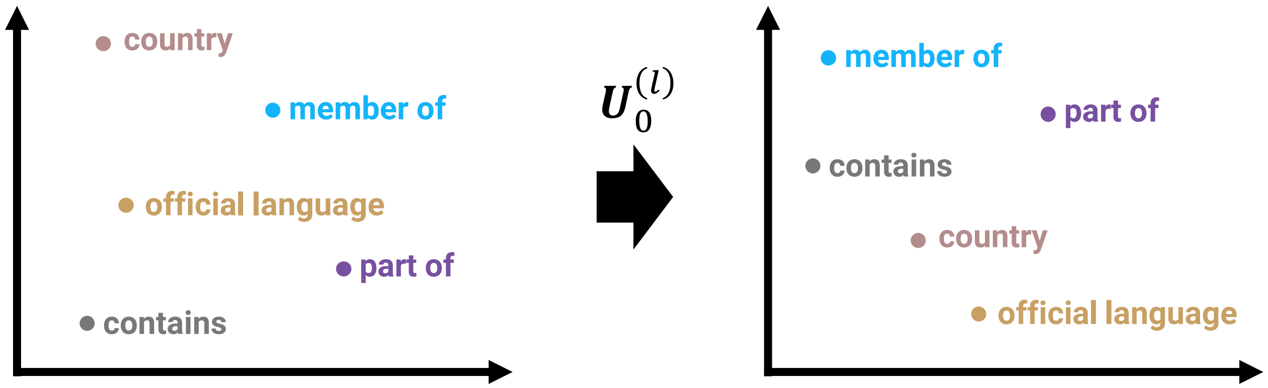


02 RAMP Encoder

- Update a relation's representation with a **projection matrix**
  - Same projection matrix for all relations

$$\boldsymbol{M}_r^{(l)}[v,:] = \left[\boldsymbol{H}^{(l-1)}[v,:] \quad \boldsymbol{R}^{(l-1)}[r,:]\right] \qquad v \in \mathcal{V}, r \in \mathcal{R}$$

$$\boldsymbol{H}^{(l)} = \phi\left(\boldsymbol{H}^{(l-1)}\boldsymbol{W}_0^{(l)} + \rho\left(\sum_{r\in\mathcal{R}} \boldsymbol{S}_r^{(l)} \psi\left(\boldsymbol{M}_r^{(l)}\right)\begin{bmatrix}\boldsymbol{W}_r^{(l)} \\ \boldsymbol{U}_r^{(l)}\end{bmatrix}\right)\right), \qquad \boldsymbol{R}^{(l)} = \boldsymbol{R}^{(l-1)}\boldsymbol{U}_0^{(l)}$$

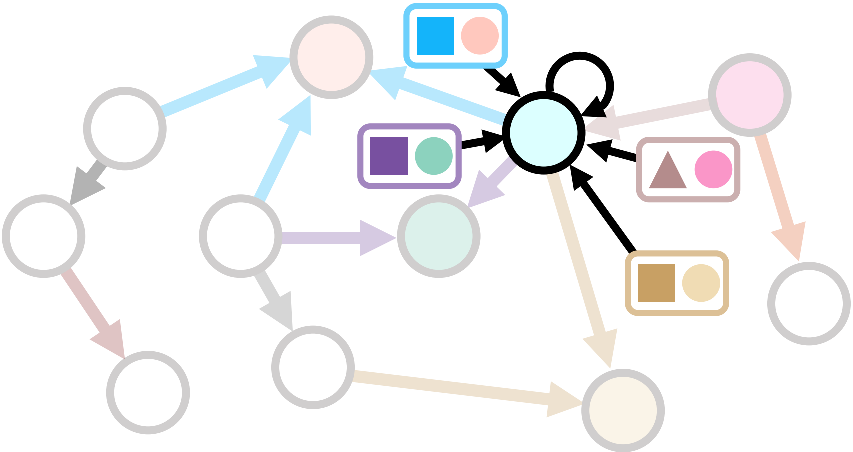


02 Special Cases of RAMP Encoder

- RAMP encoder represents the **aggregation process in a general form** that can subsume many existing KGRL encoders

$$\boldsymbol{M}_r^{(l)}[v,:] = [\boldsymbol{H}^{(l-1)}[v,:] \quad \boldsymbol{R}^{(l-1)}[r,:]] \qquad v \in \mathcal{V}, r \in \mathcal{R}$$

$$\boldsymbol{H}^{(l)} = \phi\left(\boldsymbol{H}^{(l-1)}\boldsymbol{W}_0^{(l)} + \rho\left(\sum_{r\in\mathcal{R}} \boldsymbol{S}_r^{(l)}\psi\left(\boldsymbol{M}_r^{(l)}\right)\begin{bmatrix}\boldsymbol{W}_r^{(l)} \\ \boldsymbol{U}_r^{(l)}\end{bmatrix}\right)\right), \qquad \boldsymbol{R}^{(l)} = \boldsymbol{R}^{(l-1)}\boldsymbol{U}_0^{(l)}$$

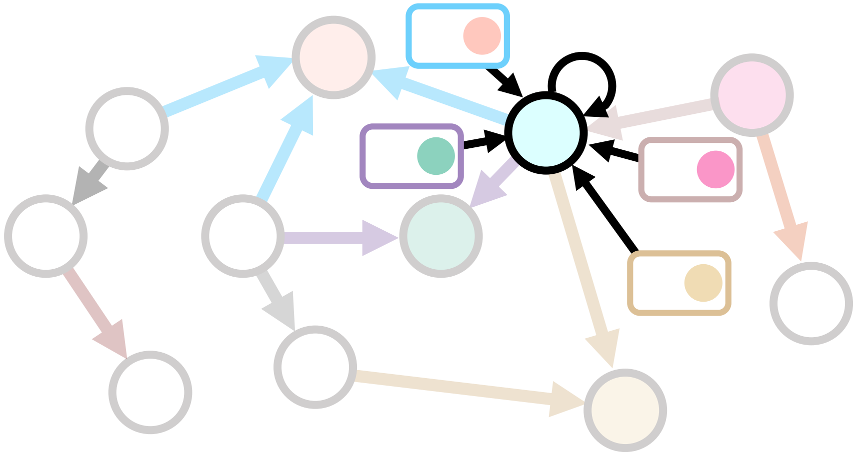


02 Special Cases of RAMP Encoder

- **R-GCN** (ESWC 2018)
  - An adjacency matrix $\boldsymbol{A}_r$ normalized by a **problem-specific constant** $c_{v,r}$ is used as the relation-specific graph diffusion matrix

$$\boldsymbol{M}_r^{(l)}[v,:] = [\boldsymbol{H}^{(l-1)}[v,:] \qquad] \qquad v \in \mathcal{V}, r \in \mathcal{R}$$

$$\boldsymbol{H}^{(l)} = \mathrm{ReLU}\left(\boldsymbol{H}^{(l-1)}\boldsymbol{W}_0^{(l)} + \left(\sum_{r\in\mathcal{R}} \boldsymbol{S}_r^{(l)} \left(\boldsymbol{M}_r^{(l)}\right)\left[\boldsymbol{W}_r^{(l)}\right]\right)\right), \qquad \boldsymbol{S}_r^{(l)}[v,:] = \frac{1}{c_{v,r}}\boldsymbol{A}_r[v,:]$$

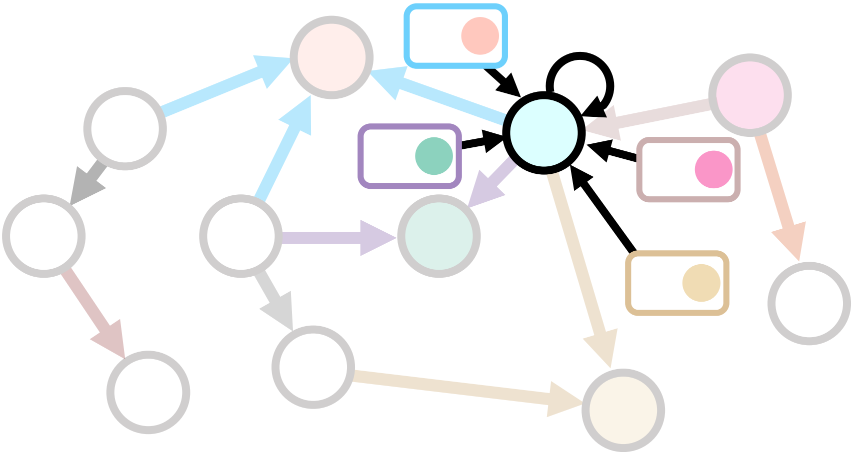


02 Special Cases of RAMP Encoder

- **WGCN** (AAAI 2019)
  - An adjacency matrix $\boldsymbol{A}_r$ is used as the relation-specific graph diffusion matrix
  - Relation-specific projection matrices **share some parameters**

$$\boldsymbol{M}_r^{(l)}[v,:] = [\boldsymbol{H}^{(l-1)}[v,:] \qquad\qquad] \quad v \in \mathcal{V}, r \in \mathcal{R}$$

$$\boldsymbol{H}^{(l)} = \mathrm{Tanh}\left(\boldsymbol{H}^{(l-1)} \boldsymbol{W}_0^{(l)} + \left(\sum_{r \in \mathcal{R}} \boldsymbol{S}_r^{(l)} \left(\boldsymbol{M}_r^{(l)} \right) \left[\alpha_r^{(l)} \boldsymbol{W}_0^{(l)} \right] \right) \right), \qquad \boldsymbol{S}_r^{(l)} = \boldsymbol{A}_r$$

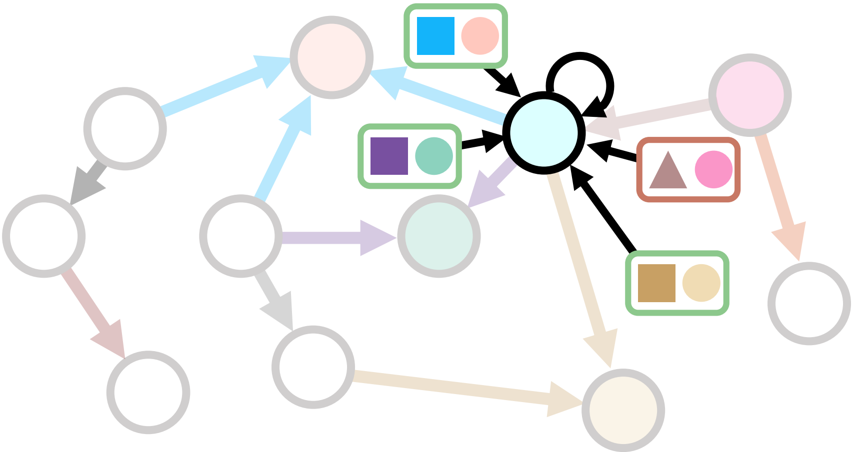


Special Cases of RAMP Encoder

- **CompGCN** (ICLR 2020)
  - An adjacency matrix $\boldsymbol{A}_r$ is used as the relation-specific graph diffusion matrix
  - Relations in the same category **share the relation-specific projection matrix**

$$\boldsymbol{M}_r^{(l)}[v,:] = [\boldsymbol{H}^{(l-1)}[v,:] \quad \boldsymbol{R}^{(l-1)}[r,:]] \qquad v \in \mathcal{V}, r \in \mathcal{R}$$

$$\boldsymbol{H}^{(l)} = \mathrm{Tanh}\left(\boldsymbol{H}^{(l-1)}\boldsymbol{W}_0^{(l)} + \left(\sum_{r\in\mathcal{R}} \boldsymbol{S}_r^{(l)} \left(\boldsymbol{M}_r^{(l)}\right)\begin{bmatrix} \boldsymbol{W}_{\lambda(r)}^{(l)} \\ -\boldsymbol{W}_{\lambda(r)}^{(l)} \end{bmatrix}\right)\right), \boldsymbol{R}^{(l)} = \boldsymbol{R}^{(l-1)}\boldsymbol{U}_0^{(l)}, \boldsymbol{S}_r^{(l)} = \boldsymbol{A}_r$$



02 Special Cases of RAMP Encoder: Summary

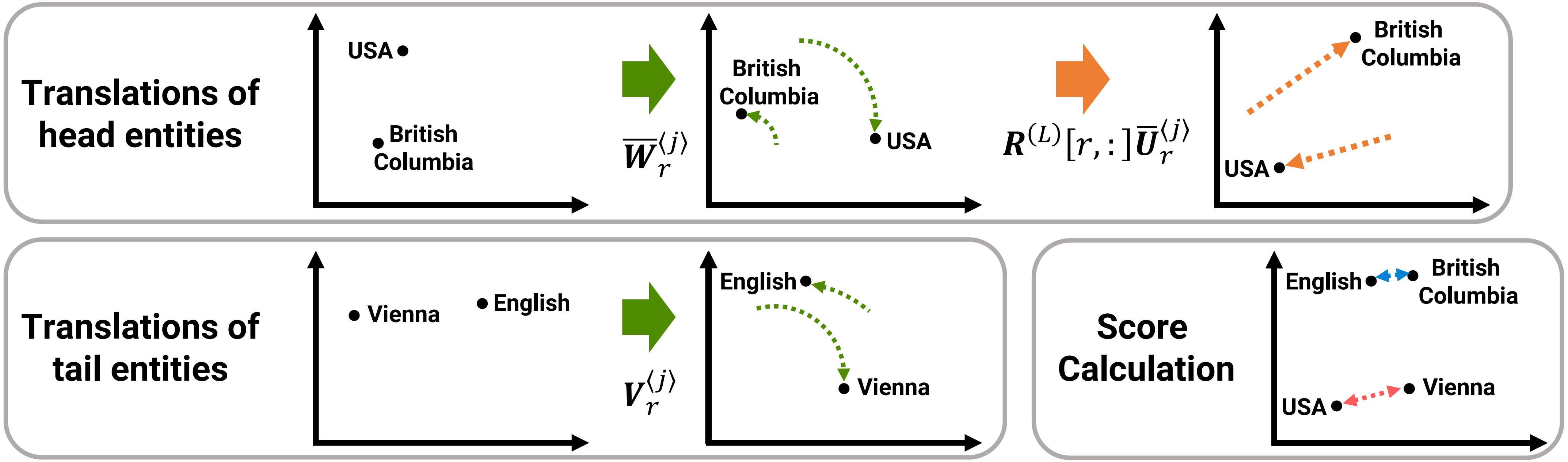
- RAMP encoder can represent R-GCN (ESWC 2018), WGCN (AAAI 2019), and CompGCN (ICLR 2020) by appropriately setting the **functions** and **matrices**

		ϕ	ρ, ψ	$\boldsymbol{W}_r^{(l)}$	$\boldsymbol{U}_r^{(l)}$	$\boldsymbol{S}_r^{(l)}[v,:]$
R-GCN		ReLU	identity	$\boldsymbol{W}_r^{(l)}$	$\mathbf{0}$	$\frac{1}{c_{v,r}}\boldsymbol{A}_r[v,:]$
WGCN		Tanh	identity	$\alpha_r^{(l)}\boldsymbol{W}_0^{(l)}$	$\mathbf{0}$	$\boldsymbol{A}_r[v,:]$
CompGCN	**Subtraction**	Tanh	identity	$\boldsymbol{W}_{\lambda(r)}^{(l)}$	$-\boldsymbol{W}_{\lambda(r)}^{(l)}$	$\boldsymbol{A}_r[v,:]$
	Multiplication	Tanh	identity	$\mathrm{diag}\left(\boldsymbol{R}^{(l-1)}[r,:]\right)\boldsymbol{W}_{\lambda(r)}^{(l)}$	$\mathbf{0}$	$\boldsymbol{A}_r[v,:]$
	Circular-correlation	Tanh	identity	$\boldsymbol{C}_r^{(l-1)}\boldsymbol{W}_{\lambda(r)}^{(l)}$	$\mathbf{0}$	$\boldsymbol{A}_r[v,:]$

02 Translational Distance Decoder

- The score of (h, r, t) is computed by the distance between h and t after **relation-specific projections** and a **relation-specific translation**

$$f_{\mathbf{w}}(h,r,t)[j] = -\left\| \boldsymbol{H}^{(L)}[h,:]\overline{\boldsymbol{W}}_r^{\langle j \rangle} + \boldsymbol{R}^{(L)}[r,:]\overline{\boldsymbol{U}}_r^{\langle j \rangle} - \boldsymbol{H}^{(L)}[t,:]\boldsymbol{V}_r^{\langle j \rangle} \right\|_2$$

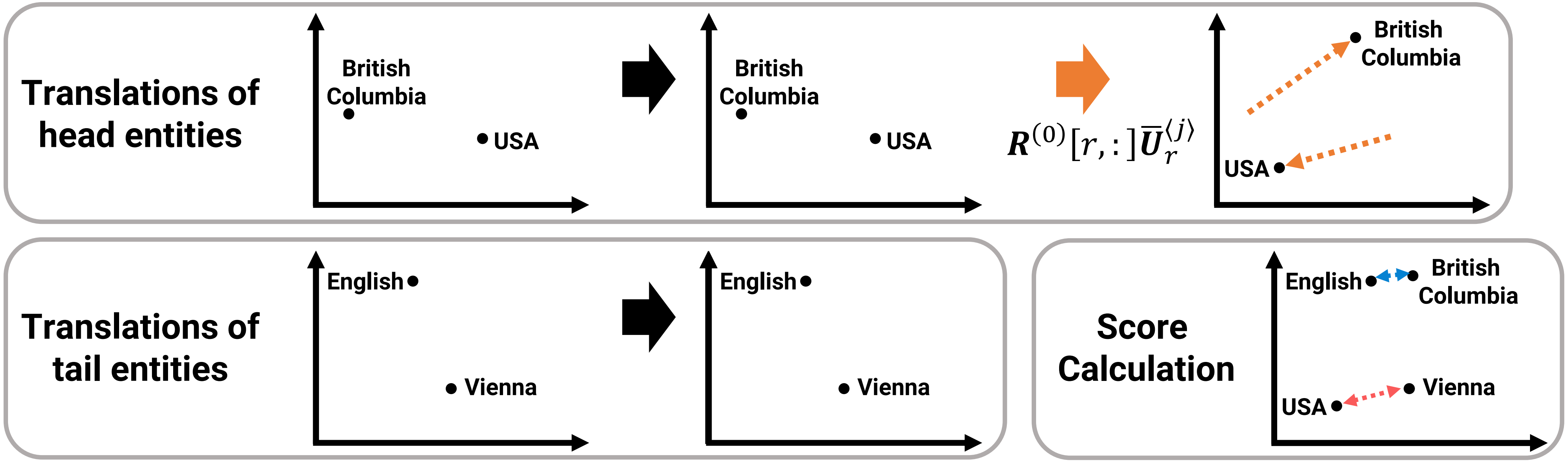


Calculating the scores of **(British Columbia, official language, English)** and **(USA, contains, Vienna)**

02 Translational Distance Decoder: TransE

- The score of (h, r, t) is computed by the distance between h and t after a **relation-specific translation**

$$f_{\mathbf{w}}(h,r,t)[j] = -\left\| \boldsymbol{H}^{(0)}[h,:]\boldsymbol{T}_{\text{ent}}^{\langle j \rangle} + \boldsymbol{R}^{(0)}[r,:]\boldsymbol{T}_{\text{rel}}^{\langle j \rangle} - \boldsymbol{H}^{(0)}[t,:]\boldsymbol{T}_{\text{ent}}^{\langle j \rangle} \right\|_2$$

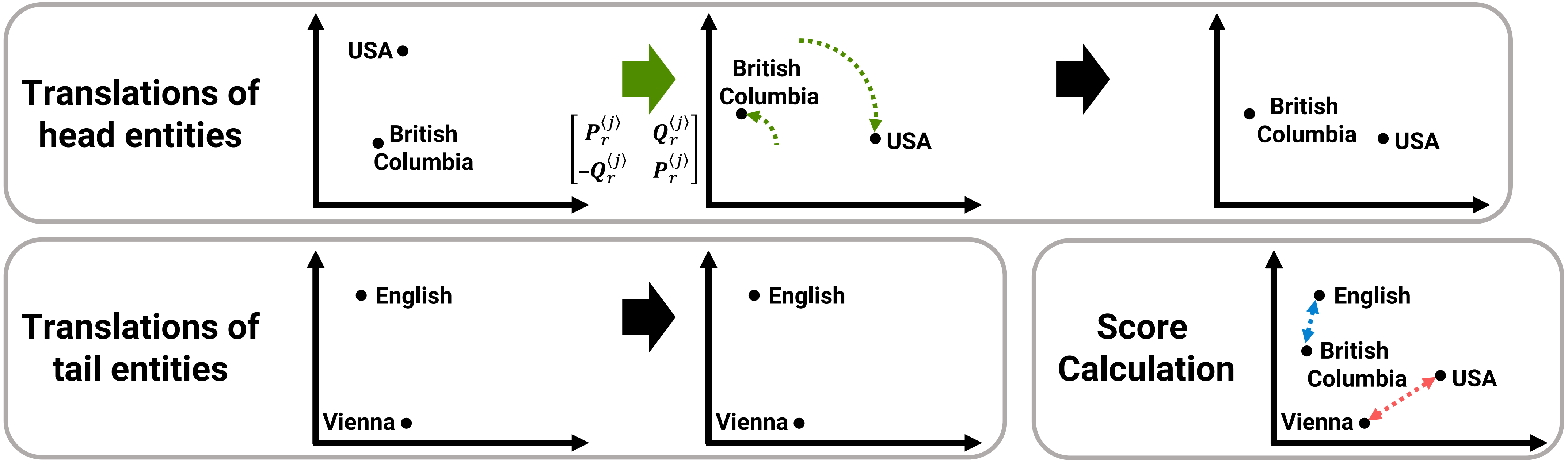


Calculating the scores of **(British Columbia, official language, English)** and **(USA, contains, Vienna)**

02 Translational Distance Decoder: RotatE

- The score of (h, r, t) is computed by the distance between h and t after a **relation-specific rotation of** h

$$f_{\mathbf{w}}(h,r,t)[j] = -\left\| \boldsymbol{H}^{(0)}[h,:]\boldsymbol{T}_{\text{ent}}^{\langle j \rangle} \begin{bmatrix} \boldsymbol{P}_r^{\langle j \rangle} & \boldsymbol{Q}_r^{\langle j \rangle} \\ -\boldsymbol{Q}_r^{\langle j \rangle} & \boldsymbol{P}_r^{\langle j \rangle} \end{bmatrix} - \boldsymbol{H}^{(0)}[t,:]\boldsymbol{T}_{\text{ent}}^{\langle j \rangle} \right\|_2$$



Calculating the scores of **(British Columbia, official language, English)** and **(USA, contains, Vienna)**

Translational Distance Decoder: Summary

- The score of (h, r, t) is computed by the distance between h and t after **relation-specific projections** and a **relation-specific translation**

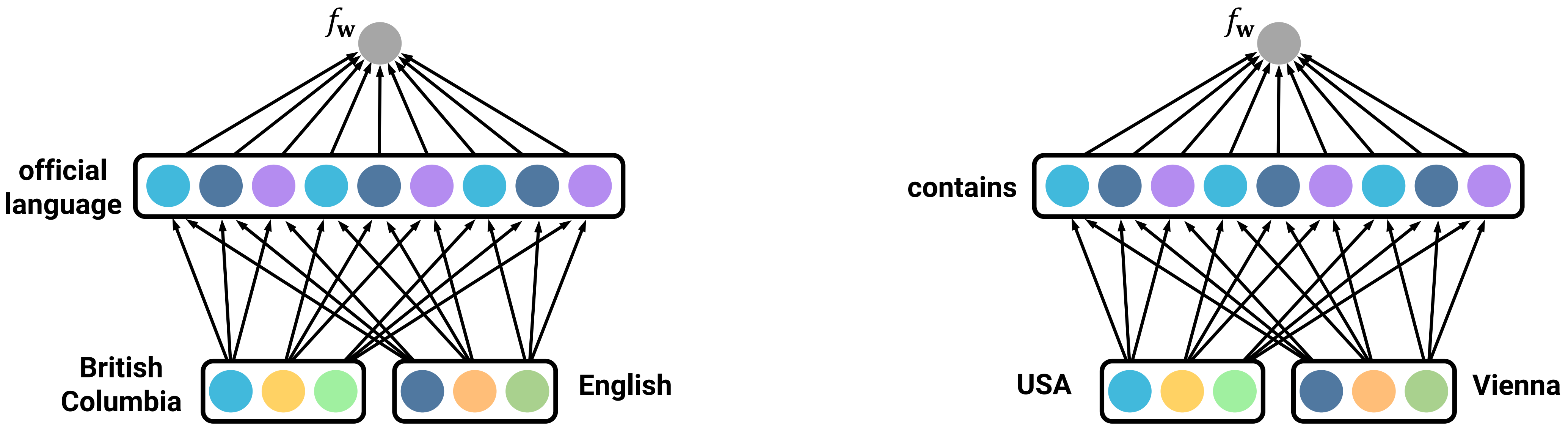
$$f_{\mathbf{w}}(h,r,t)[j] = -\left\| \boldsymbol{H}^{(0)}[h,:]\overline{\boldsymbol{W}}_r^{\langle j\rangle} + \boldsymbol{R}^{(0)}[r,:]\overline{\boldsymbol{U}}_r^{\langle j\rangle} - \boldsymbol{H}^{(0)}[t,:]\boldsymbol{V}_r^{\langle j\rangle} \right\|_2$$

	$\overline{\boldsymbol{W}}_r^{\langle j\rangle}$	$\overline{\boldsymbol{U}}_r^{\langle j\rangle}$	$\boldsymbol{V}_r^{\langle j\rangle}$
TransE (NeurIPS 2013)	$\boldsymbol{T}_{\text{ent}}^{\langle j\rangle}$	$\boldsymbol{T}_{\text{rel}}^{\langle j\rangle}$	$\boldsymbol{T}_{\text{ent}}^{\langle j\rangle}$
TransH (AAAI 2014)	$\boldsymbol{T}_{\text{ent}}^{\langle j\rangle}(\boldsymbol{I} - \mathbf{f}_r^{\langle j\rangle\top}\mathbf{f}_r^{\langle j\rangle})$	$\boldsymbol{T}_{\text{rel}}^{\langle j\rangle}$	$\boldsymbol{T}_{\text{ent}}^{\langle j\rangle}(\boldsymbol{I} - \mathbf{f}_r^{\langle j\rangle\top}\mathbf{f}_r^{\langle j\rangle})$
TransR (AAAI 2015)	$\boldsymbol{T}_{\text{ent}}^{\langle j\rangle}\boldsymbol{F}_r^{\langle j\rangle}$	$\boldsymbol{T}_{\text{rel}}^{\langle j\rangle}$	$\boldsymbol{T}_{\text{ent}}^{\langle j\rangle}\boldsymbol{F}_r^{\langle j\rangle}$
RotatE (ICLR 2019)	$\boldsymbol{T}_{\text{ent}}^{\langle j\rangle}\begin{bmatrix}\boldsymbol{P}_r^{\langle j\rangle} & \boldsymbol{Q}_r^{\langle j\rangle} \\ -\boldsymbol{Q}_r^{\langle j\rangle} & \boldsymbol{P}_r^{\langle j\rangle}\end{bmatrix}$	$\boldsymbol{0}$	$\boldsymbol{T}_{\text{ent}}^{\langle j\rangle}$
PairRE (ACL 2021)	$\boldsymbol{T}_{\text{ent}}^{\langle j\rangle}\boldsymbol{\mathfrak{F}}_r^{\langle j\rangle}$	$\boldsymbol{0}$	$\boldsymbol{T}_{\text{ent}}^{\langle j\rangle}\dot{\boldsymbol{\mathfrak{F}}}_r^{\langle j\rangle}$

02 Semantic Matching Decoder

- The score of (h, r, t) is computed by the **similarity** between the individual components of the triplet

$$f_{\mathbf{w}}(h,r,t)[j] = \boldsymbol{H}^{(L)}[h,:]\overline{\boldsymbol{U}}_r^{\langle j \rangle}\left(\boldsymbol{H}^{(L)}[t,:]\right)^{\top}$$

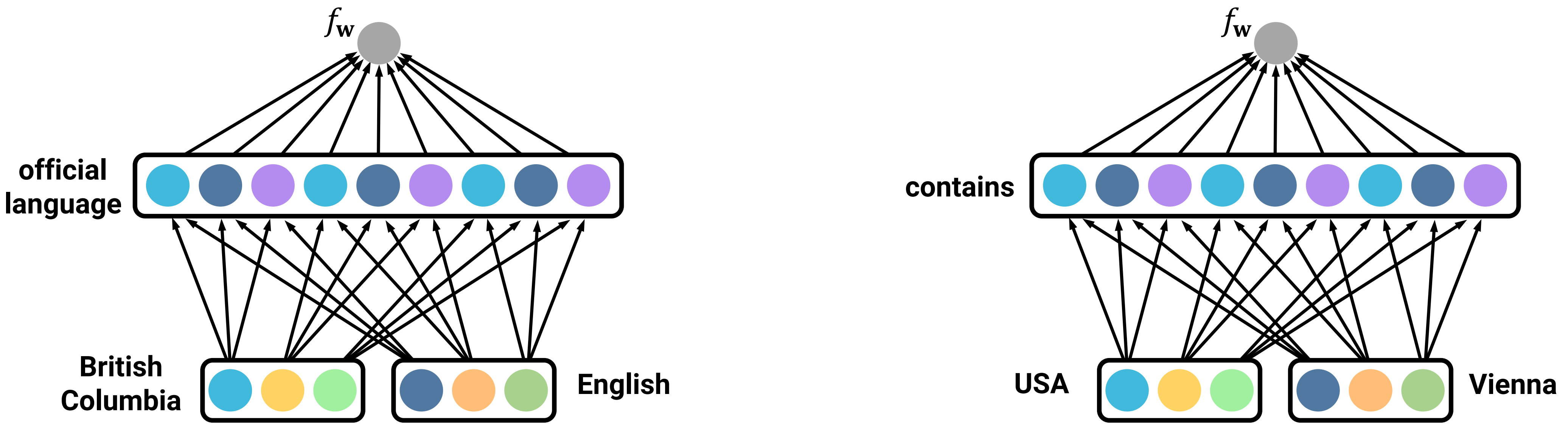


Calculating scores of **(British Columbia, official language, English)** and **(USA, contains, Vienna)**

02 Semantic Matching Decoder: RESCAL

- The score of (h, r, t) is computed by the **pairwise multiplication** between the individual components of the triplet

$$f_{\mathbf{w}}(h,r,t)[j] = \boldsymbol{H}^{(0)}[h,:]\boldsymbol{T}_{\text{ent}}^{<j>}\boldsymbol{B}_r^{<j>}{\boldsymbol{T}_{\text{ent}}^{<j>}}^{\top}\left(\boldsymbol{H}^{(0)}[t,:]\right)^{\top}$$

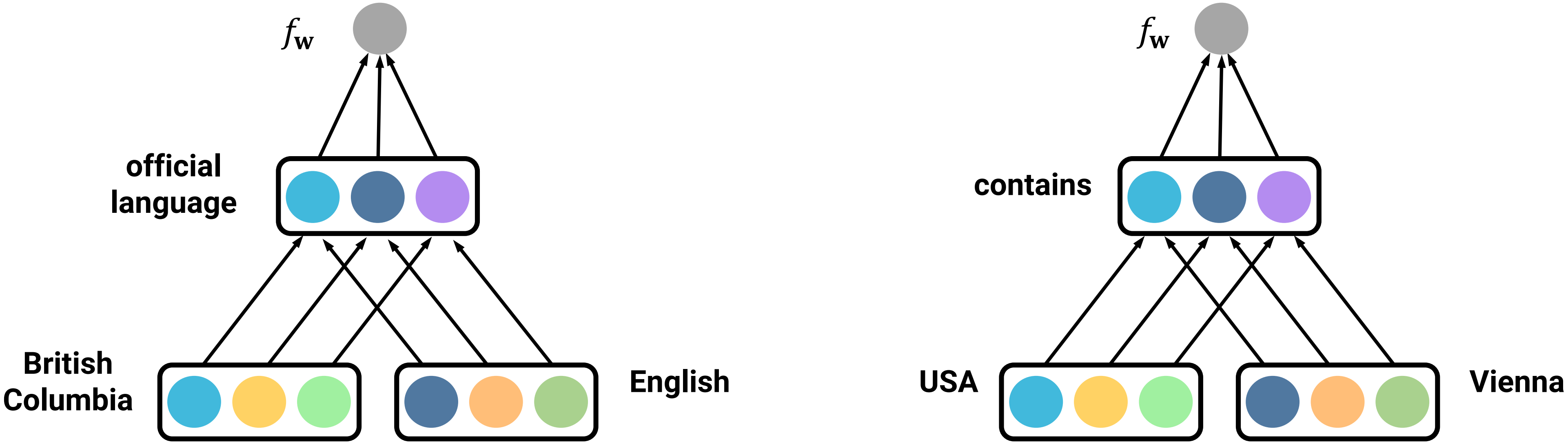


Calculating scores of **(British Columbia, official language, English)** and **(USA, contains, Vienna)**

02 Semantic Matching Decoder: DistMult

- The score of (h, r, t) is computed by the sum of the **Hadamard product** of the individual components of the triplet

$$f_{\mathbf{w}}(h,r,t)[j] = \boldsymbol{H}^{(0)}[h,:]\boldsymbol{T}_{\text{ent}}^{<j>}\operatorname{diag}\left(\boldsymbol{R}^{(0)}[r,:]\boldsymbol{T}_{\text{rel}}^{<j>}\right)\boldsymbol{T}_{\text{ent}}^{<j>\top}\left(\boldsymbol{H}^{(0)}[t,:]\right)^{\top}$$



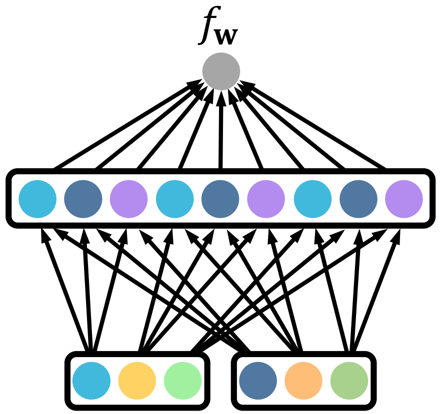
Calculating scores of **(British Columbia, official language, English)** and **(USA, contains, Vienna)**

Semantic Matching Decoder: Summary

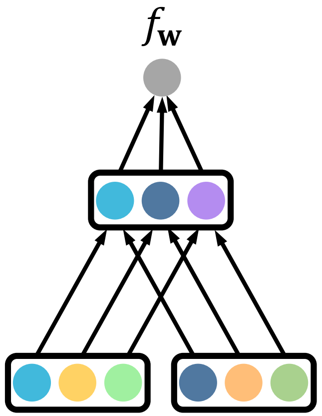
- The score of (h, r, t) is computed by the **similarity** between the individual components of the triplet

$$f_{\mathbf{w}}(h,r,t)[j] = \boldsymbol{H}^{(0)}[h,:]\,\bar{\boldsymbol{U}}_r^{\langle j\rangle}\left(\boldsymbol{H}^{(0)}[t,:]\right)^{\top}$$

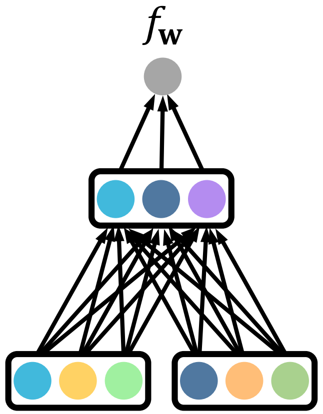
RESCAL
(ICML 2011)



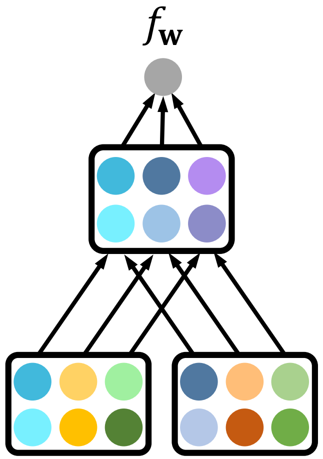
DistMult
(ICLR 2015)



HolE
(AAAI 2016)



ComplEx
(ICML 2016)

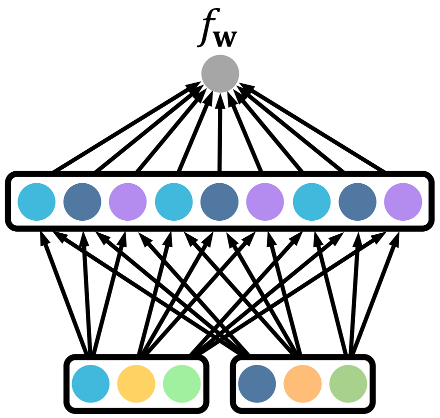


02 Semantic Matching Decoder: Summary

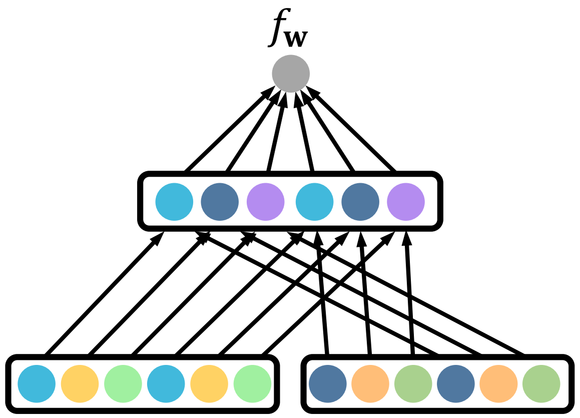
- The score of (h, r, t) is computed by the **similarity** between the individual components of the triplet

$$f_{\mathbf{w}}(h,r,t)[j] = \boldsymbol{H}^{(0)}[h,:]\,\overline{\boldsymbol{U}}_r^{\langle j \rangle}\left(\boldsymbol{H}^{(0)}[t,:]\right)^{\top}$$

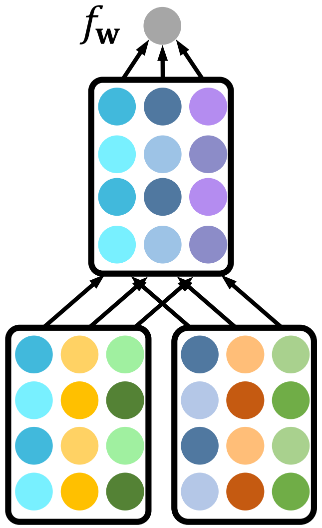
ANALOGY
(ICML 2017)



SimplE
(NeurIPS 2018)

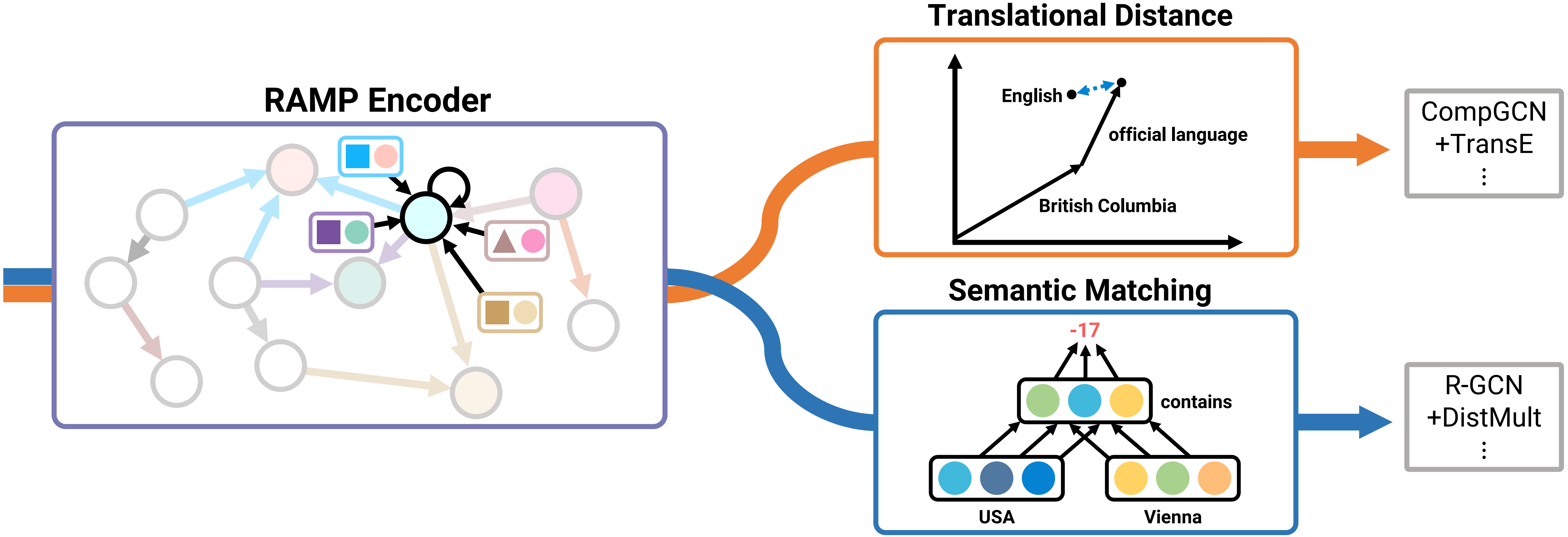


QuatE
(NeurIPS 2019)



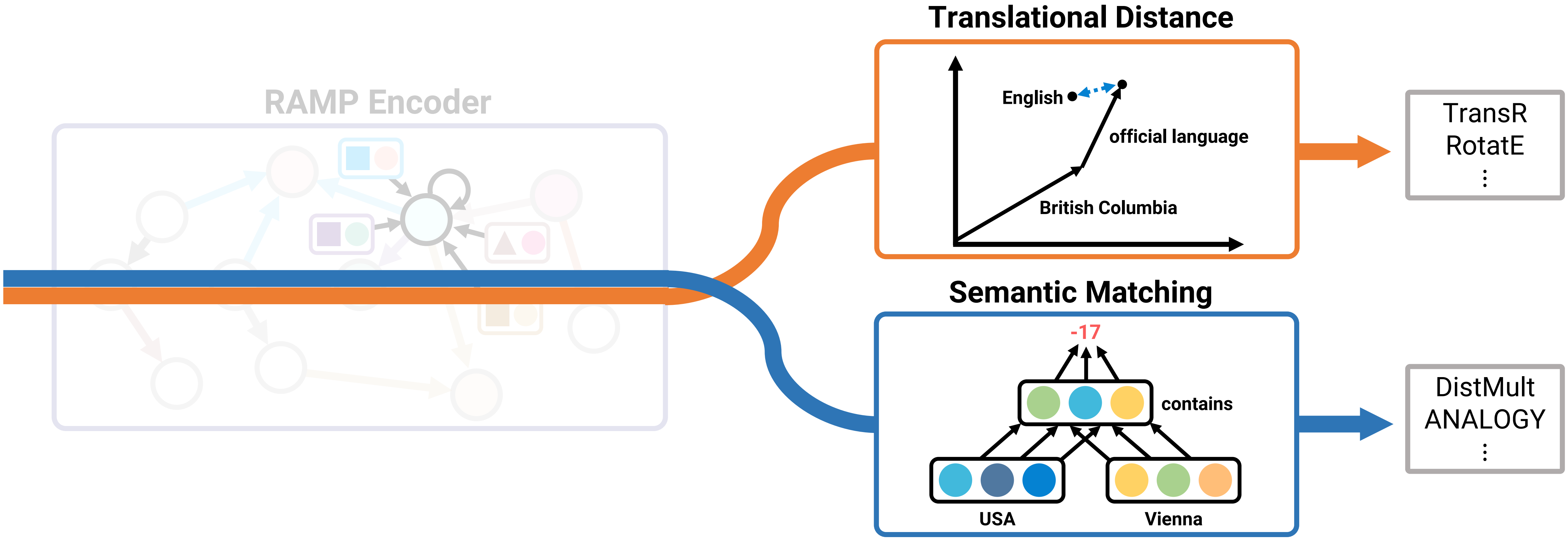
02 Instantiations of ReED

- ReED can express various KGRL methods using **different instantiations and configurations** of the RAMP encoder and the triplet classification decoder



02 Instantiations of ReED

- A triplet classification decoder can also be **used standalone**

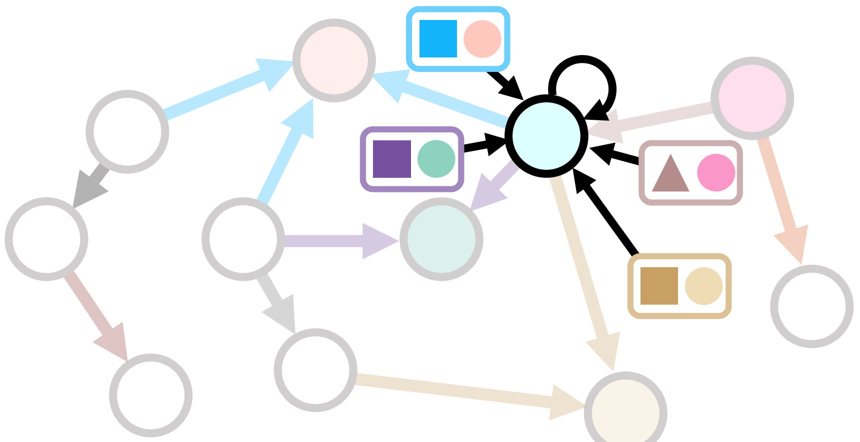


02 Instantiations of ReED

- Our ReED Framework can express at least 15 different existing KGRL models

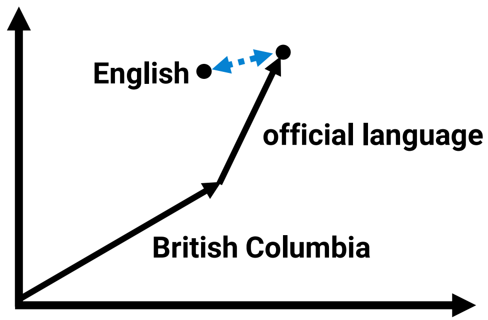
Graph Neural Network-based models

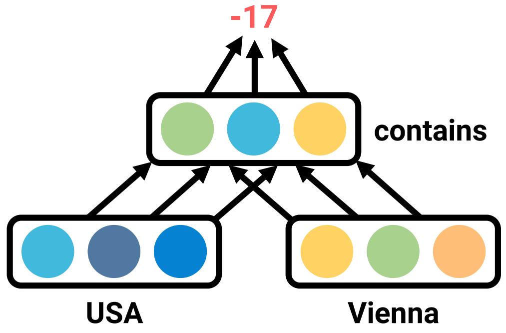
- **R-GCN** (ESWC 2018)
- **WGCN** (AAAI 2019)
- **CompGCN** (ICLR 2020)



Shallow-architecture Models

- **TransE** (NeurIPS 2013)
- **TransH** (AAAI 2014)
- **TransR** (AAAI 2015)
- **RotatE** (ICLR 2019)
- **PairRE** (ACL 2021)
- **RESCAL** (ICML 2011)
- **DistMult** (ICLR 2015)
- **HolE** (AAAI 2016)
- **ComplEx** (ICML 2016)
- **ANALOGY** (ICML 2017)
- **SimplE** (NeurIPS 2018)
- **QuatE** (NeurIPS 2019)





03 Empirical Loss of a Triplet Classifier

- **γ-margin Loss:** take into account when the score of **the ground-truth label** is less than or equal to that of **the other label** with a **margin** of γ

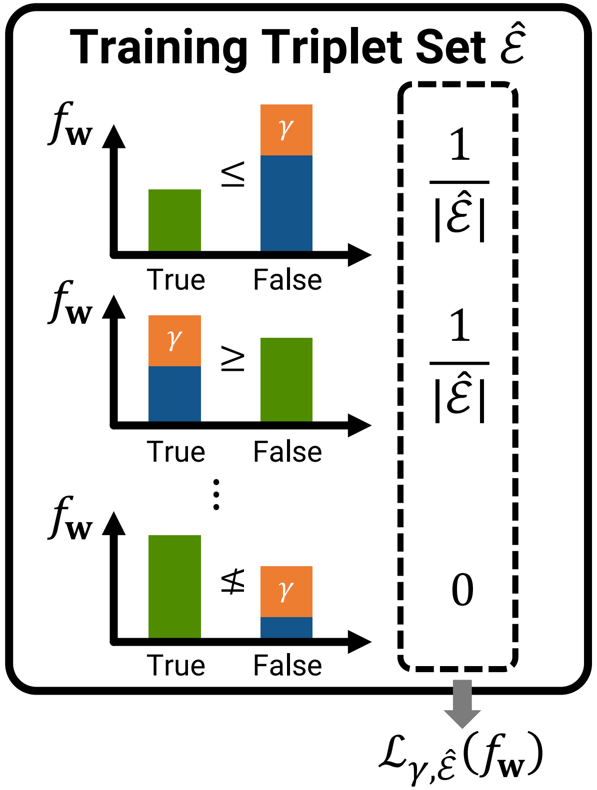
Definition (γ-margin loss of Triplet Classifier)

$$\mathcal{L}_{\gamma,\mathcal{Z}}(f_{\mathbf{w}}) = \frac{1}{|\mathcal{Z}|}\sum_{(h,r,t)\in\mathcal{Z}} \mathbf{1}\left[f_{\mathbf{w}}(h,r,t)[y_{hrt}] \le \gamma + f_{\mathbf{w}}(h,r,t)[1-y_{hrt}]\right]$$

Measured on a **training triplet set**

Definition (**Empirical Loss** of Triplet Classifier)

$$\mathcal{L}_{\gamma,\hat{\mathcal{E}}}(f_{\mathbf{w}}) = \frac{1}{|\hat{\mathcal{E}}|}\sum_{(h,r,t)\in\hat{\mathcal{E}}} \mathbf{1}\left[f_{\mathbf{w}}(h,r,t)[y_{hrt}] \le \gamma + f_{\mathbf{w}}(h,r,t)[1-y_{hrt}]\right]$$



Score of the ground-truth label

Score of the other label

03 Expected Loss of a Triplet Classifier

- **Classification Loss**: take into account when the score of **the ground-truth label** is less than or equal to that of **the other label**

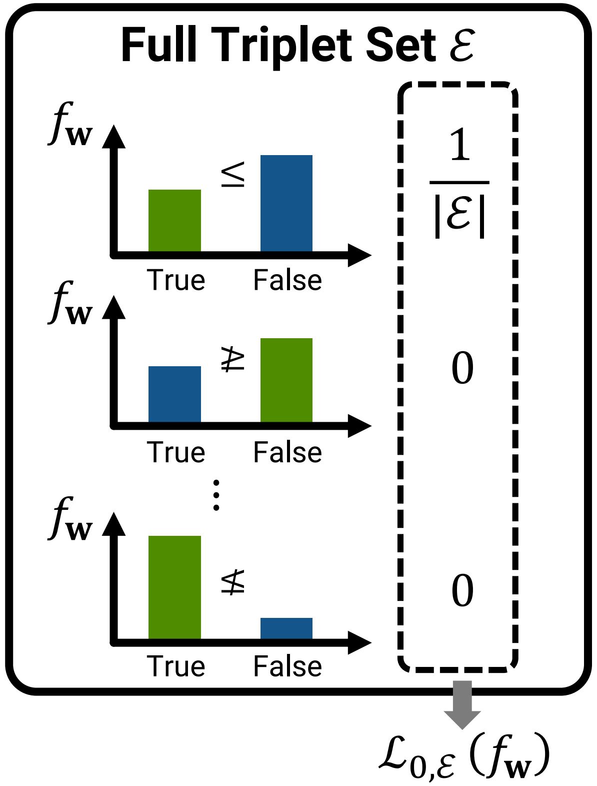
Definition (Classification Loss of Triplet Classifier)

$$\mathcal{L}_{0,\mathcal{Z}}(f_{\mathbf{w}}) = \frac{1}{|\mathcal{Z}|} \sum_{(h,r,t)\in\mathcal{Z}} \mathbf{1}\left[f_{\mathbf{w}}(h,r,t)[y_{hrt}] \le f_{\mathbf{w}}(h,r,t)[1-y_{hrt}]\right]$$

Measured on the **full triplet set**

Definition (**Expected Loss** of Triplet Classifier)

$$\mathcal{L}_{0,\mathcal{E}}(f_{\mathbf{w}}) = \frac{1}{|\mathcal{E}|} \sum_{(h,r,t)\in\mathcal{E}} \mathbf{1}\left[f_{\mathbf{w}}(h,r,t)[y_{hrt}] \le f_{\mathbf{w}}(h,r,t)[1-y_{hrt}]\right]$$



Score of the ground-truth label
Score of the other label

03 Transductive PAC-Bayesian Generalization Bounds

- Extends the **transductive PAC-Bayesian generalization bound** for the stochastic classifier to the **deterministic classifier**

Theorem 4.3 Let $f_{\mathbf{w}}: \mathcal{V} \times \mathcal{R} \times \mathcal{V} \to \mathbb{R}^2$ be a **deterministic triplet classifier** with parameters $\mathbf{w}$, and $\mathcal{P}$ be any prior distribution on $\mathbf{w}$. Let us consider the finite full triplet set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{R} \times \mathcal{V}$. Construct a posterior distribution $\mathcal{Q}_{\mathbf{w}+\ddot{\mathbf{w}}}$ by adding any random perturbation $\ddot{\mathbf{w}}$ to $\mathbf{w}$ such that $\mathbb{P}\left(\max_{(h,r,t)\in\mathcal{E}} \|f_{\mathbf{w}+\ddot{\mathbf{w}}}(h,r,t) - f_{\mathbf{w}}(h,r,t)\|_\infty < \frac{1}{4}\right) > \frac{1}{2}$. Then, for any $\gamma, \delta > 0$, with probability $1-\delta$ over the choice of a training triplet set $\hat{\mathcal{E}}$ drawn from the full triplet set $\mathcal{E}$ (such that $20 \le |\hat{\mathcal{E}}| \le |\mathcal{E}| - 20$ and $|\mathcal{E}| \ge 40$) without replacement, for any $\mathbf{w}$, we have

$$\mathcal{L}_{0,\mathcal{E}}(f_{\mathbf{w}}) \le \mathcal{L}_{\gamma,\hat{\mathcal{E}}}(f_{\mathbf{w}}) + \sqrt{\frac{1-\frac{|\hat{\mathcal{E}}|}{|\mathcal{E}|}}{2|\hat{\mathcal{E}}|}\left[2D_{KL}(\mathcal{Q}_{\mathbf{w}+\ddot{\mathbf{w}}}||\mathcal{P}) + \ln\frac{4\theta\left(|\hat{\mathcal{E}}|,|\mathcal{E}|\right)}{\delta}\right]}$$

where $D_{KL}(\mathcal{Q}_{\mathbf{w}+\ddot{\mathbf{w}}}||\mathcal{P})$ is the KL-divergence of $\mathcal{Q}_{\mathbf{w}+\ddot{\mathbf{w}}}$ from $\mathcal{P}$, and $\theta\left(|\hat{\mathcal{E}}|,|\mathcal{E}|\right) = 3\sqrt{|\hat{\mathcal{E}}|\left(1-\frac{|\hat{\mathcal{E}}|}{|\mathcal{E}|}\right)}\ln|\hat{\mathcal{E}}|$

03 Generalization Bounds for ReED: Proof Sketch

Transductive PAC-Bayesian Generalization Bounds

Add **random perturbations** to the fixed parameters

Transductive PAC-Bayesian Generalization Bounds for a **Deterministic Classifier**

Prior & Posterior Distributions on the Hypothesis Space

Assume the Gaussian distributions with the **same standard deviation**

Unroll **two-step recursions** considering **interactions** between **entities and relations**

Architecture of the **RAMP Encoder**

Perturbation Bound of ReED

Calculate the **KL-divergence**

Generalization Bound for ReED with **TD**

Covering Ball Analysis

Generalization Bound for ReED with **SM**

03 Generalization Bounds for ReED: Assumptions

Assumption 1

All activation functions are **Lipschitz-continuous** with respect to the Euclidean norm of input/output vectors.

Assumption 2

The training triplets in $\hat{\mathcal{E}}$ are **sampled** from the finite full triplet set $\mathcal{E}$ **without replacement**.

Assumption 3

Regarding the sizes of $\mathcal{E}$ and $\hat{\mathcal{E}}$, we assume $|\mathcal{E}| \geq 40$ and $20 \leq |\hat{\mathcal{E}}| \leq |\mathcal{E}| - 20$

Generalization Bound for ReED with TD

- Compute the **generalization bound** of a model that uses the **RAMP encoder** and the **TD decoder**

Theorem 4.4 For any $L \geq 0$, let $f_{\mathbf{w}}: \mathcal{V} \times \mathcal{R} \times \mathcal{V} \to \mathbb{R}^2$ be a triplet classifier designed by the combination of the RAMP encoder with L-layers and the TD decoder. Let k_r be the maximum of the infinity norms for all possible $\boldsymbol{S}_r^{(l)}$ in the RAMP encoder. Then, for any $\delta, \gamma > 0$, with probability at least $1 - \delta$ over a training triplet set $\hat{\mathcal{E}}$, for any $\mathbf{w}$, we have

$$\mathcal{L}_{0,\mathcal{E}}(f_{\mathbf{w}}) \leq \mathcal{L}_{\gamma,\hat{\mathcal{E}}}(f_{\mathbf{w}}) + \mathcal{O}\left(\sqrt{\frac{1 - \frac{|\hat{\mathcal{E}}|}{|\mathcal{E}|}}{|\hat{\mathcal{E}}|}\left[\frac{N_{\mathbf{w}}\, L^2\, \zeta_L^2\, s^{2L}\, d \ln(N_{\mathbf{w}} d)}{\gamma^2} + \ln \frac{\theta\left(|\hat{\mathcal{E}}|, |\mathcal{E}|\right)}{\delta}\right]}\right)$$

where $\theta\left(|\hat{\mathcal{E}}|, |\mathcal{E}|\right) = 3\sqrt{|\hat{\mathcal{E}}|\left(1 - \frac{|\hat{\mathcal{E}}|}{|\mathcal{E}|}\right)} \ln|\hat{\mathcal{E}}|$, $\zeta_L = 2\tau^L \|\boldsymbol{X}_{\text{ent}}\|_2 + 2\kappa \|\boldsymbol{X}_{\text{ent}}\|_2 \left(\sum_{i=0}^{L-1} \tau^i\right) + \|\boldsymbol{X}_{\text{rel}}\|_2$, $\tau = C_\phi + \kappa$, $\kappa = C_\phi C_\rho C_\psi \sum_{r \in \mathcal{R}} k_r$, $N_{\mathbf{w}}$ is the total number of learnable matrices, d is the maximum dimension, and s is the maximum Frobenius norm of the learnable matrices

03 Generalization Bound for ReED with TD

- Generalization bound **increases** as the total **number of learnable matrices increases**
  - Explains the effectiveness of the **parameter-sharing strategies** and the **basis/block decomposition**

Theorem 4.4 For any $L \geq 0$, let $f_{\mathbf{w}}: \mathcal{V} \times \mathcal{R} \times \mathcal{V} \rightarrow \mathbb{R}^2$ be a triplet classifier designed by the combination of the RAMP encoder with L-layers and the TD decoder. Let k_r be the maximum of the infinity norms for all possible $\boldsymbol{S}_r^{(l)}$ in the RAMP encoder. Then, for any $\delta, \gamma > 0$, with probability at least $1 - \delta$ over a training triplet set $\hat{\mathcal{E}}$, for any $\mathbf{w}$, we have

$$\mathcal{L}_{0,\mathcal{E}}(f_{\mathbf{w}}) \leq \mathcal{L}_{\gamma,\hat{\mathcal{E}}}(f_{\mathbf{w}}) + \mathcal{O}\left(\sqrt{\frac{1 - \frac{|\hat{\mathcal{E}}|}{|\mathcal{E}|}}{|\hat{\mathcal{E}}|}\left[\frac{N_{\mathbf{w}} L^2 \zeta_L^2 s^{2L} d \ln(N_{\mathbf{w}} d)}{\gamma^2} + \ln\frac{\theta(|\hat{\mathcal{E}}|, |\mathcal{E}|)}{\delta}\right]}\right)$$

where $\theta(|\hat{\mathcal{E}}|, |\mathcal{E}|) = 3\sqrt{|\hat{\mathcal{E}}|\left(1 - \frac{|\hat{\mathcal{E}}|}{|\mathcal{E}|}\right)} \ln|\hat{\mathcal{E}}|$, $\zeta_L = 2\tau^L \|\boldsymbol{X}_{\text{ent}}\|_2 + 2\kappa \|\boldsymbol{X}_{\text{ent}}\|_2 \left(\sum_{i=0}^{L-1} \tau^i\right) + \|\boldsymbol{X}_{\text{rel}}\|_2$, $\tau = C_\phi + \kappa$, $\kappa = C_\phi C_\rho C_\psi \sum_{r \in \mathcal{R}} k_r$, $N_{\mathbf{w}}$ is **the total number of learnable matrices**, d is the maximum dimension, and s is the maximum Frobenius norm of the learnable matrices

03 Generalization Bound for ReED with TD

- Generalization bound **increases** as the **number of layers** in the RAMP encoder **increases**

Theorem 4.4 For any $L \geq 0$, let $f_{\mathbf{w}}: \mathcal{V} \times \mathcal{R} \times \mathcal{V} \rightarrow \mathbb{R}^2$ be a triplet classifier designed by the combination of **the RAMP encoder with L-layers** and the TD decoder. Let k_r be the maximum of the infinity norms for all possible $\boldsymbol{S}_r^{(l)}$ in the RAMP encoder. Then, for any $\delta, \gamma > 0$, with probability at least $1 - \delta$ over a training triplet set $\hat{\mathcal{E}}$, for any $\mathbf{w}$, we have

$$\mathcal{L}_{0,\mathcal{E}}(f_{\mathbf{w}}) \leq \mathcal{L}_{\gamma,\hat{\mathcal{E}}}(f_{\mathbf{w}}) + \mathcal{O}\left(\sqrt{\frac{1-\frac{|\hat{\mathcal{E}}|}{|\mathcal{E}|}}{|\hat{\mathcal{E}}|}\left[\frac{N_{\mathbf{w}}\, L^2\, \zeta_L^2\, s^{2L}\, d \ln(N_{\mathbf{w}} d)}{\gamma^2} + \ln\frac{\theta\left(|\hat{\mathcal{E}}|, |\mathcal{E}|\right)}{\delta}\right]}\right)$$

where $\theta\left(|\hat{\mathcal{E}}|, |\mathcal{E}|\right) = 3\sqrt{|\hat{\mathcal{E}}|\left(1 - \frac{|\hat{\mathcal{E}}|}{|\mathcal{E}|}\right)} \ln|\hat{\mathcal{E}}|$, $\zeta_L = 2\tau^L \|\boldsymbol{X}_{\text{ent}}\|_2 + 2\kappa\|\boldsymbol{X}_{\text{ent}}\|_2\left(\sum_{i=0}^{L-1}\tau^i\right) + \|\boldsymbol{X}_{\text{rel}}\|_2$, $\tau = C_\phi + \kappa$, $\kappa = C_\phi C_\rho C_\psi \sum_{r\in\mathcal{R}} k_r$, $N_{\mathbf{w}}$ is the total number of learnable matrices, d is the maximum dimension, and s is the maximum Frobenius norm of the learnable matrices

03 Generalization Bound for ReED with TD

- Generalization bound **increases** as the **infinity norms of the diffusion matrices increase**
  - A **mean aggregator** is a better option than a **sum aggregator** in reducing the generalization bound

Theorem 4.4 For any $L \geq 0$, let $f_{\mathbf{w}}: \mathcal{V} \times \mathcal{R} \times \mathcal{V} \to \mathbb{R}^2$ be a triplet classifier designed by the combination of the RAMP encoder with L-layers and the TD decoder. Let k_r be **the maximum of the infinity norms for all possible $S_r^{(l)}$ in the RAMP encoder**. Then, for any $\delta, \gamma > 0$, with probability at least $1 - \delta$ over a training triplet set $\hat{\mathcal{E}}$, for any $\mathbf{w}$, we have

$$\mathcal{L}_{0,\mathcal{E}}(f_{\mathbf{w}}) \leq \mathcal{L}_{\gamma,\hat{\mathcal{E}}}(f_{\mathbf{w}}) + \mathcal{O}\left(\sqrt{\frac{1 - \frac{|\hat{\mathcal{E}}|}{|\mathcal{E}|}}{|\hat{\mathcal{E}}|}\left[\frac{N_{\mathbf{w}}\, L^2\, \boxed{\zeta_L^2}\, s^{2L}\, d \ln(N_{\mathbf{w}} d)}{\gamma^2} + \ln\frac{\theta\left(|\hat{\mathcal{E}}|, |\mathcal{E}|\right)}{\delta}\right]}\right)$$

where $\theta(|\hat{\mathcal{E}}|, |\mathcal{E}|) = 3\sqrt{|\hat{\mathcal{E}}|\left(1 - \frac{|\hat{\mathcal{E}}|}{|\mathcal{E}|}\right)} \ln|\hat{\mathcal{E}}|$, $\zeta_L = 2\tau^L \|\boldsymbol{X}_{\text{ent}}\|_2 + 2\kappa\|\boldsymbol{X}_{\text{ent}}\|_2\left(\sum_{i=0}^{L-1} \tau^i\right) + \|\boldsymbol{X}_{\text{rel}}\|_2$, $\tau = C_\phi + \kappa$, $\kappa = C_\phi C_\rho C_\psi \sum_{r \in \mathcal{R}} k_r$, $N_{\mathbf{w}}$ is the total number of learnable matrices, d is the maximum dimension, and s is the maximum Frobenius norm of the learnable matrices

03 Generalization Bound for ReED with TD

- Generalization bound **increases** as the **norms of the learnable matrices increase**
  - Provides theoretical justification for **weight normalization** & **normalization of entity representations**

Theorem 4.4 For any $L \geq 0$, let $f_{\mathbf{w}}: \mathcal{V} \times \mathcal{R} \times \mathcal{V} \to \mathbb{R}^2$ be a triplet classifier designed by the combination of the RAMP encoder with L-layers and the TD decoder. Let k_r be the maximum of the infinity norms for all possible $\boldsymbol{S}_r^{(l)}$ in the RAMP encoder. Then, for any $\delta, \gamma > 0$, with probability at least $1 - \delta$ over a training triplet set $\hat{\mathcal{E}}$, for any $\mathbf{w}$, we have

$$\mathcal{L}_{0,\mathcal{E}}(f_{\mathbf{w}}) \leq \mathcal{L}_{\gamma,\hat{\mathcal{E}}}(f_{\mathbf{w}}) + \mathcal{O}\left(\sqrt{\frac{1 - \frac{|\hat{\mathcal{E}}|}{|\mathcal{E}|}}{|\hat{\mathcal{E}}|}\left[\frac{N_{\mathbf{w}}\, L^2\, \zeta_L^2\, \boxed{s^{2L}}\, d \ln(N_{\mathbf{w}} d)}{\gamma^2} + \ln\frac{\theta\left(|\hat{\mathcal{E}}|, |\mathcal{E}|\right)}{\delta}\right]}\right)$$

where $\theta\left(|\hat{\mathcal{E}}|, |\mathcal{E}|\right) = 3\sqrt{|\hat{\mathcal{E}}|\left(1 - \frac{|\hat{\mathcal{E}}|}{|\mathcal{E}|}\right)}\ln|\hat{\mathcal{E}}|$, $\zeta_L = 2\tau^L\|\boldsymbol{X}_{\mathrm{ent}}\|_2 + 2\kappa\|\boldsymbol{X}_{\mathrm{ent}}\|_2\left(\sum_{i=0}^{L-1}\tau^i\right) + \|\boldsymbol{X}_{\mathrm{rel}}\|_2$, $\tau = C_\phi + \kappa$, $\kappa = C_\phi C_\rho C_\psi \sum_{r\in\mathcal{R}} k_r$, $N_{\mathbf{w}}$ is the total number of learnable matrices, d is the maximum dimension, and s is **the maximum Frobenius norm of the learnable matrices**

03 Generalization Bound for ReED with TD

- Generalization bound **increases** as the **dimensions increase**

Theorem 4.4 For any $L \geq 0$, let $f_{\mathbf{w}}: \mathcal{V} \times \mathcal{R} \times \mathcal{V} \rightarrow \mathbb{R}^2$ be a triplet classifier designed by the combination of the RAMP encoder with L-layers and the TD decoder. Let k_r be the maximum of the infinity norms for all possible $\boldsymbol{S}_r^{(l)}$ in the RAMP encoder. Then, for any $\delta, \gamma > 0$, with probability at least $1 - \delta$ over a training triplet set $\hat{\mathcal{E}}$, for any $\mathbf{w}$, we have

$$\mathcal{L}_{0,\mathcal{E}}(f_{\mathbf{w}}) \leq \mathcal{L}_{\gamma,\hat{\mathcal{E}}}(f_{\mathbf{w}}) + \mathcal{O}\left(\sqrt{\frac{1 - \frac{|\hat{\mathcal{E}}|}{|\mathcal{E}|}}{|\hat{\mathcal{E}}|}\left[\frac{N_{\mathbf{w}}\, L^2\, \zeta_L^2\, s^{2L}\, d \ln(N_{\mathbf{w}} d)}{\gamma^2} + \ln \frac{\theta\left(|\hat{\mathcal{E}}|, |\mathcal{E}|\right)}{\delta}\right]}\right)$$

where $\theta\left(|\hat{\mathcal{E}}|, |\mathcal{E}|\right) = 3\sqrt{|\hat{\mathcal{E}}|\left(1 - \frac{|\hat{\mathcal{E}}|}{|\mathcal{E}|}\right)} \ln|\hat{\mathcal{E}}|$, $\zeta_L = 2\tau^L \|\boldsymbol{X}_{\text{ent}}\|_2 + 2\kappa \|\boldsymbol{X}_{\text{ent}}\|_2 \left(\sum_{i=0}^{L-1} \tau^i\right) + \|\boldsymbol{X}_{\text{rel}}\|_2$, $\tau = C_\phi + \kappa$, $\kappa = C_\phi C_\rho C_\psi \sum_{r \in \mathcal{R}} k_r$, $N_{\mathbf{w}}$ is the total number of learnable matrices, d is **the maximum dimension**, and s is the maximum Frobenius norm of the learnable matrices

03 Generalization Bounds for ReED: Proof Sketch

Transductive PAC-Bayesian Generalization Bounds

Add **random perturbations** to the fixed parameters

Transductive PAC-Bayesian Generalization Bounds for a **Deterministic Classifier**

Unroll **two-step recursions** considering **interactions** between **entities and relations**

Architecture of the **RAMP Encoder**

Perturbation Bound of ReED

Calculate the **KL-divergence**

Generalization Bound for ReED with **TD**

Covering Ball Analysis

Generalization Bound for ReED with **SM**

Prior & Posterior Distributions on the Hypothesis Space

Assume the Gaussian distributions with the **same standard deviation**

03 Generalization Bound for ReED with SM

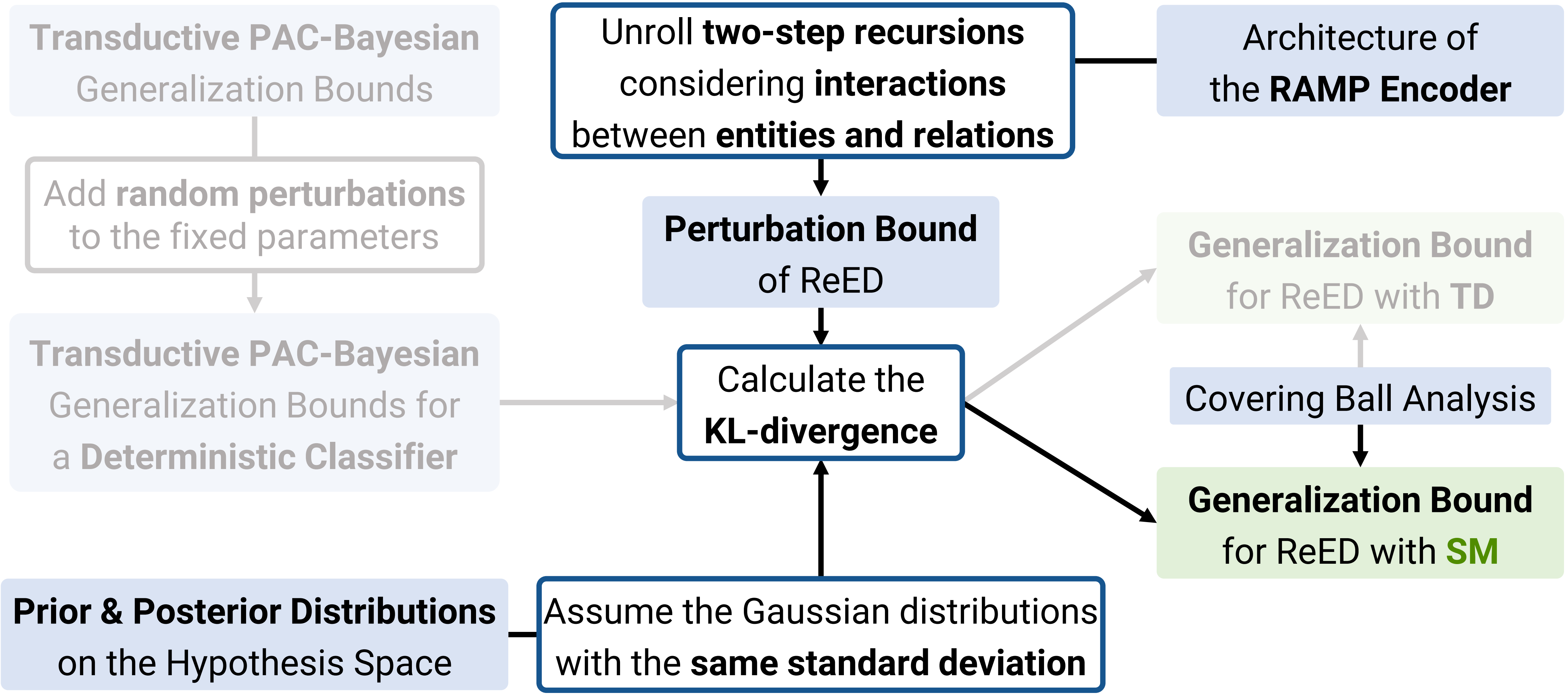
- Compute the **generalization bound** of a model that uses the **RAMP encoder** and the **SM decoder**
  - While the **magnitude** may vary, the increasing and decreasing trends of the factors are same with TD

Theorem 4.5 For any $L \geq 0$, let $f_{\mathbf{w}}: \mathcal{V} \times \mathcal{R} \times \mathcal{V} \to \mathbb{R}^2$ be a triplet classifier designed by the combination of the RAMP encoder with L-layers and the SM decoder. Let k_r be the maximum of the infinity norms for all possible $\boldsymbol{S}_r^{(l)}$ in the RAMP encoder. Then, for any $\delta, \gamma > 0$, with probability at least $1 - \delta$ over a training triplet set $\hat{\mathcal{E}}$, for any $\mathbf{w}$, we have

$$\mathcal{L}_{0,\mathcal{E}}(f_{\mathbf{w}}) \leq \mathcal{L}_{\gamma,\hat{\mathcal{E}}}(f_{\mathbf{w}}) + \mathcal{O}\left(\sqrt{\frac{1 - \frac{|\hat{\mathcal{E}}|}{|\mathcal{E}|}}{|\hat{\mathcal{E}}|}\left[\frac{N_{\mathbf{w}}\, L^2\, \eta_L^4\, s^{4L}\, d \ln(N_{\mathbf{w}} d)}{\gamma^2} + \ln\frac{\theta\left(|\hat{\mathcal{E}}|, |\mathcal{E}|\right)}{\delta}\right]}\right)$$

where $\theta\left(|\hat{\mathcal{E}}|, |\mathcal{E}|\right) = 3\sqrt{|\hat{\mathcal{E}}|\left(1 - \frac{|\hat{\mathcal{E}}|}{|\mathcal{E}|}\right)}\ln|\hat{\mathcal{E}}|$, $\eta_L = \tau^L \|\boldsymbol{X}_{\mathrm{ent}}\|_2 + \kappa\|\boldsymbol{X}_{\mathrm{rel}}\|_2\left(\sum_{i=0}^{L-1} \tau^i\right)$, $\tau = C_\phi + \kappa$, $\kappa = C_\phi C_\rho C_\psi \sum_{r\in\mathcal{R}} k_r$, $N_{\mathbf{w}}$ is the total number of learnable matrices, d is the maximum dimension, and s is the maximum Frobenius norm of the learnable matrices

03 Generalization Bounds for ReED: Proof Sketch



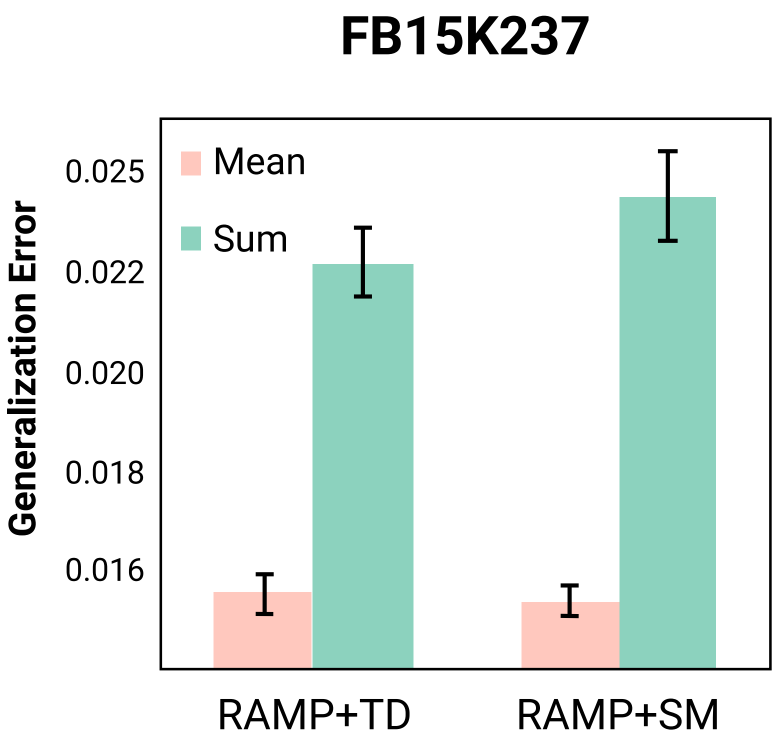
Experimental Results

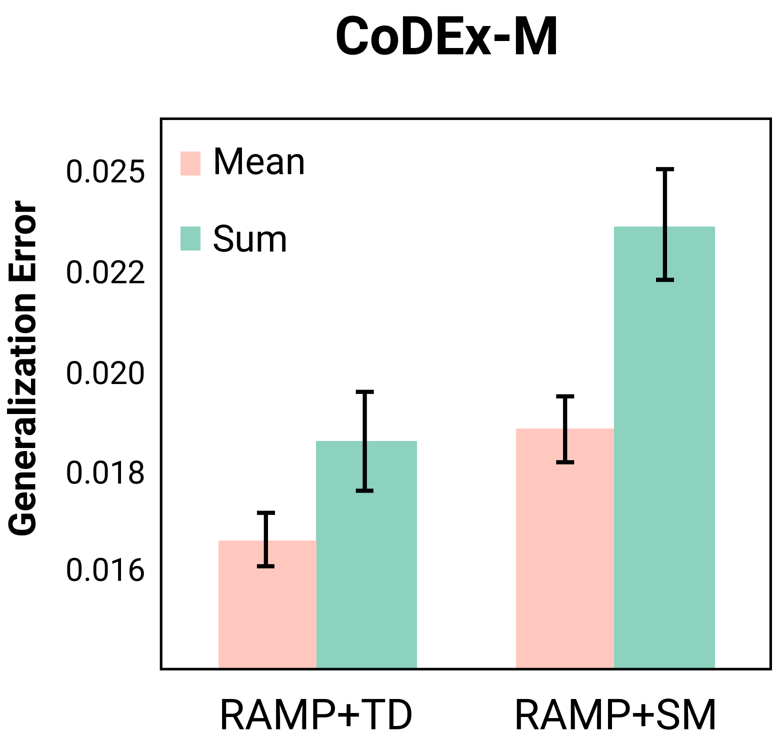
- **Datasets**
  - Sampled from **three real-world knowledge graphs**
  - FB15K237, CoDEx-M, UMLS-43

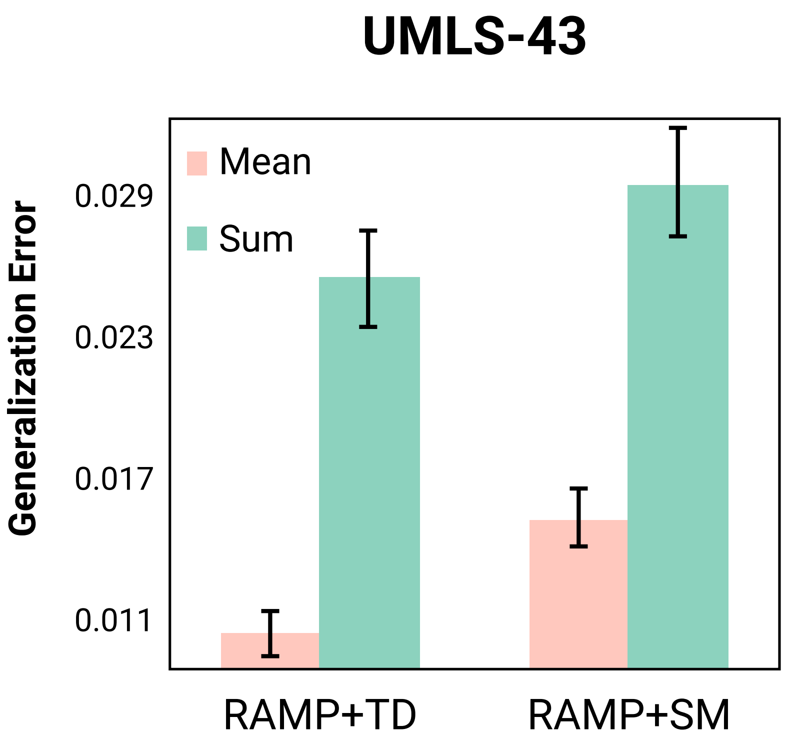
- **Experimental Details**
  - Create a training triplet set by sampling without replacement from the full triplet set
  - **Measure the generalization errors** on real-world datasets
    - Generalization error: an **actual error** observed in a particular experiment
    - Generalization bound: the **theoretical upper bound** of a generalization error

04 Varying the Aggregator: Mean vs Sum

- Generalization errors of **sum aggregators** are **higher** than **mean aggregators**

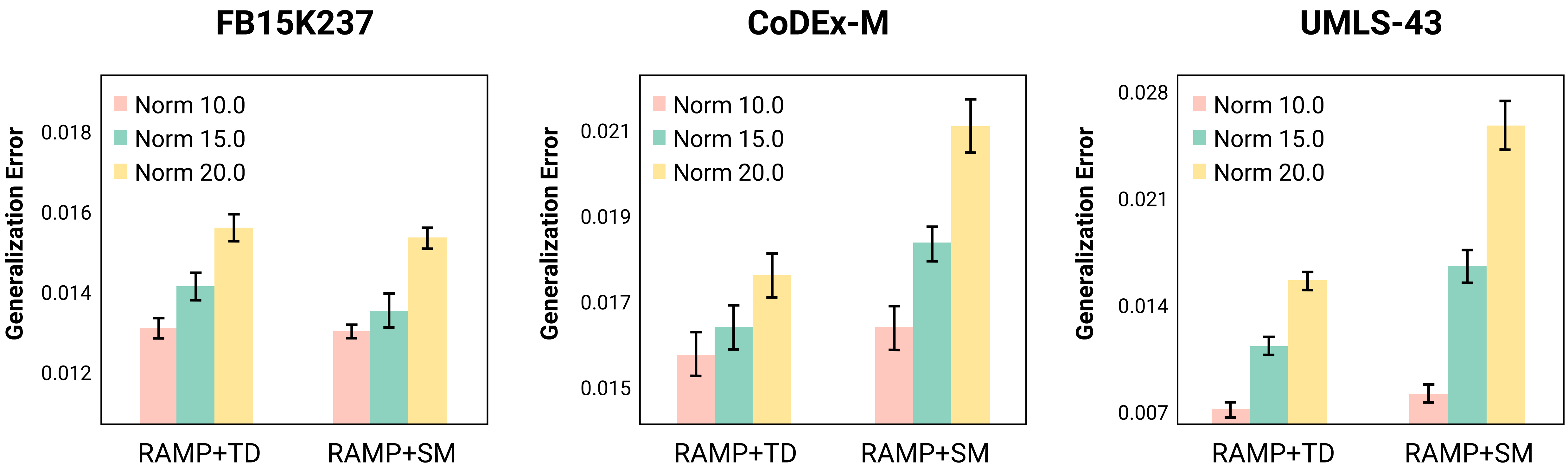






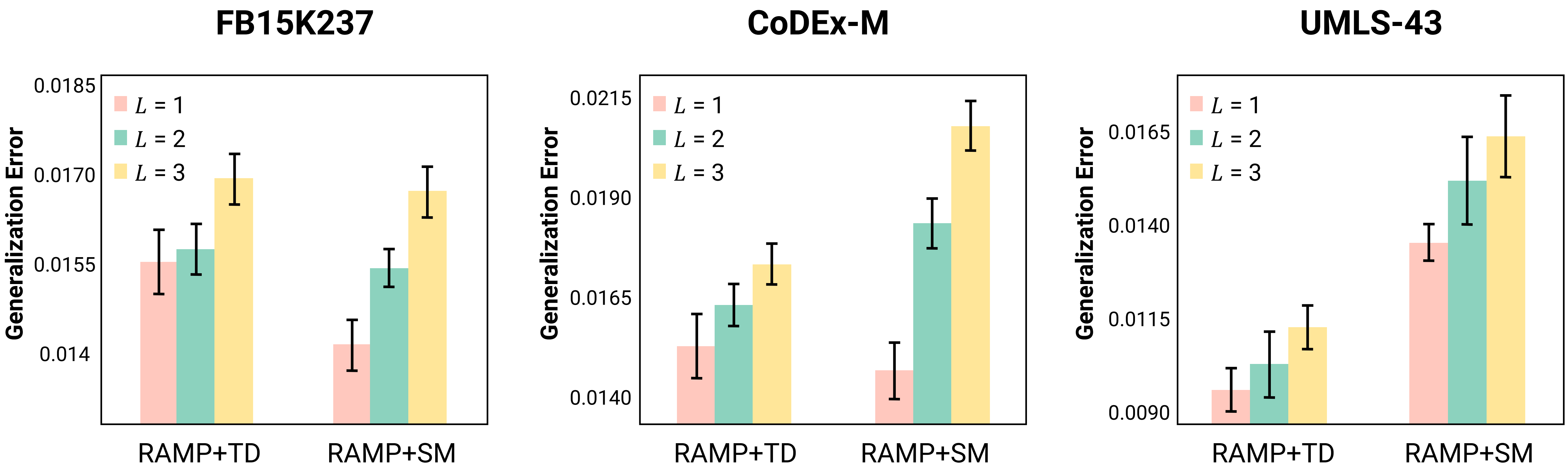
04 Varying the Norms of Weight Matrices

- Generalization errors **increase** as the **norm of weight matrices increases**



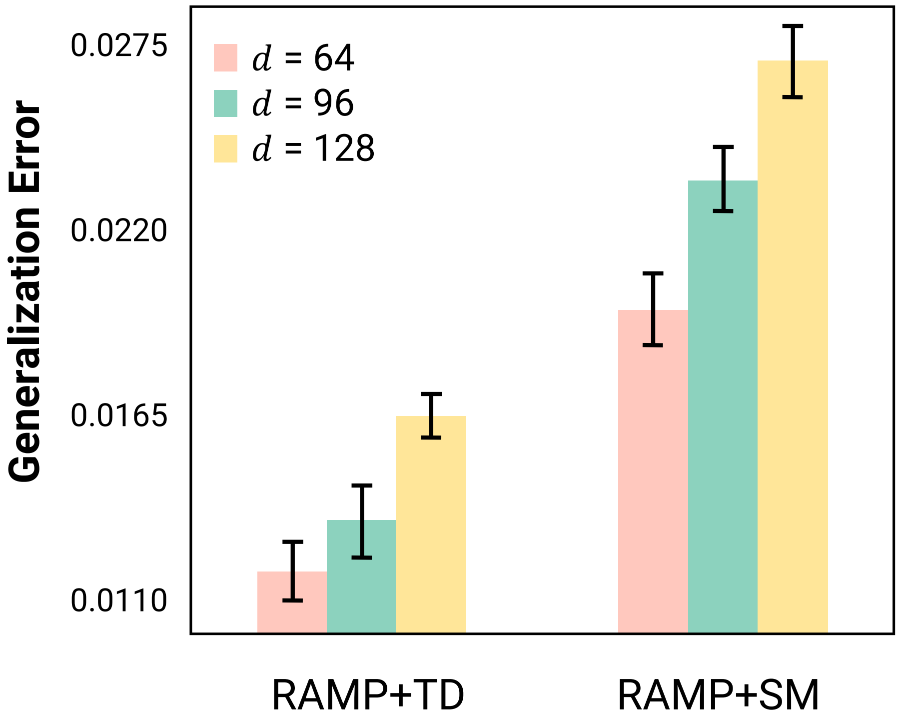
04 Varying the Number of Layers

- Generalization errors **increase** as the **number of layers** in the encoder **increases**



04 Varying the Maximum Dimension

- Generalization errors **increase** as the **maximum dimension increases**
  - Extract the **initial features** from **textual descriptions** of entities and relations in FB15K237



05 Conclusion

- A novel **ReED framework** expressing at least 15 KGRL models
  - Subsume both GNN-based models and shallow-architecture models
- The **first PAC-Bayesian generalization bounds** for ReED with different types of decoders
  - ReED with Translational Distance decoder and Semantic Matching decoder
- Provide **theoretical grounds** for commonly used tricks in KGRL
  - E.g., parameter-sharing and weight normalization schemes
- Empirically show the relationships between **the critical factors in the theoretical bounds** and **the actual generalization errors**
  - The critical factors explaining the generalization bounds also affect an actual generalization error

Thank You!

Our datasets and codes are available at:
https://github.com/bdi-lab/ReED

You can find us at:
{jjlee98, hminsung, jjwhang}@kaist.ac.kr
https://bdi-lab.kaist.ac.kr