1 Source Code

The source code of ERDNS is available at https://github.com/for4ever44/ERDNS.

2 Proof

2.1 The Proof for Proposition 1

Proof. When substitute N with $N_{(e,r)}$, the NS loss function with $w = 1$ can be written as:

$$L = -\frac{1}{|\mathcal{F}|} \sum_{\left(q,(e,r)\right) \in \mathcal{F}} \left[\log \sigma\Big(g\big(q,(e,r)\big)\Big) + \sum_{\bar{q_i} \sim p_n\left(\bar{q_i}|(e,r)\right)}^{N_{(e,r)}} \log \sigma\Big(-g\big(\bar{q_i},(e,r)\big)\Big) \right] \tag{1}$$

We can reformulate this Equation using the Monte Carlo method leading to the following expression.

$$\begin{aligned} L &\approx -\sum_{\left(q,(e,r)\right)} \left[p_d\big(q,(e,r)\big) \log \sigma\Big(g\big(q,(e,r)\big)\Big) + N_{(e,r)} p_n\big(\bar{q}|(e,r)\big) p_d(e,r) \log \sigma\Big(-g\big(\bar{q},(e,r)\big)\Big) \right] \\ &= -\sum_{(e,r)} p_d(e,r) \left\{ \sum_q \left[p_d\big(q|(e,r)\big) \log \sigma\Big(g\big(q,(e,r)\big)\Big) + N_{(e,r)} p_n\big(q|(e,r)\big) \log \sigma\Big(-g\big(q,(e,r)\big)\Big) \right] \right\} \end{aligned} \tag{2}$$

We follow the proving process proposed by Yang et al. [1] to demonstrate that minimizing the NS loss function is equivalent to minimizing the loss for each Entity-Relation pair (e, r).

$$L^{(e,r)} \approx -\sum_q \left[p_d\big(q|(e,r)\big) \log \sigma\Big(g\big(q,(e,r)\big)\Big) + N_{(e,r)} p_n\big(q|(e,r)\big) \log \sigma\Big(-g\big(q,(e,r)\big)\Big) \right] \tag{3}$$

According to proof process in [1], we can define two Bernoulli distributions $A_{(q,(e,r))}(z)$ and $B_{(q,(e,r))}(z)$ for each q and (e, r), where $A_{(q,(e,r))}(z = 1) = \frac{p_d\left(q|(e,r)\right)}{p_d\left(q|(e,r)\right) + N_{(e,r)} p_n\left(q|(e,r)\right)} < 0$ and $B_{(q,(e,r))}(z = 1) = \sigma\big(g(q,(e,r))\big) < 0$. With these definitions, we can transform the loss for each (e, r) as follows:

$$\begin{aligned} L^{(e,r)} &= -\sum_q \left[p_d\big(q|(e,r)\big) \log \sigma\Big(g\big(q,(e,r)\big)\Big) + N_{(e,r)} p_n\big(q|(e,r)\big) \Big(1 - \log \sigma\Big(g\big(q|(e,r)\big)\Big)\Big) \right] \\ &= -\sum_q \Big(p_d\big(q|(e,r)\big) + N_{(e,r)} p_n\big(q|(e,r)\big) \Big) \Big[A(z=1) \log B(z=1) + A(z=0) \log B(z=0) \Big] \\ &= -\sum_q \left[\Big(p_d\big(q|(e,r)\big) + N_{(e,r)} p_n\big(q|(e,r)\big) \Big) H(A, B) \right] \end{aligned} \tag{4}$$

The cross-entropy between two discrete distributions is given by $H(A, B) = -\sum_{z \in (0,1)} A(z) \log B(z)$, where A and B are the two distributions being compared. According to Gibbs Inequality, for each q and (e, r), the minimum value of $L^{(e,r)}$ occurs when A is equal to B.

$$\begin{aligned} \sigma\Big(g\big(q,(e,r)\big)\Big) &= \frac{p_d\big(q|(e,r)\big)}{p_d\big(q|(e,r)\big)+N_{(e,r)}p_n\big(q|(e,r)\big)} \\ \frac{1}{1+\exp\Big(-g\big(q,(e,r)\big)\Big)} &= \frac{p_d\big(q|(e,r)\big)}{p_d\big(q|(e,r)\big)+N_{(e,r)}p_n\big(q|(e,r)\big)} \\ f^*\big(q,(e,r)\big) &= \log\frac{p_d\big(q|(e,r)\big)}{N_{(e,r)}exp(\kappa)p_n\big(q|(e,r)\big)} \end{aligned} \tag{5}$$

2.2 The Proof for Proposition 2

Proof. The number of negative samples $N_{(t,r)}$ and $N_{(h,r)}$ can be determined according to the distribution of $p_d(h|(t,r))$ and $p_d(t|(h,r))$.

$$\begin{aligned} f^*\big(h,(t,r)\big) &= f^*\big(t,(h,r)\big) \\ \log\frac{p_d\big(h|(t,r)\big)}{N_{(t,r)}exp(\kappa)p_n\big(h|(t,r)\big)} &= \log\frac{p_d\big(t|(h,r)\big)}{N_{(h,r)}exp(\kappa)p_n\big(t|(h,r)\big)} \\ \frac{p_d\big(h|(t,r)\big)}{N_{(t,r)}} &= \frac{p_d\big(t|(h,r)\big)}{N_{(h,r)}} \\ \frac{N_{(t,r)}}{N_{(h,r)}} &= \frac{p_d\big(h|(t,r)\big)}{p_d\big(t|(h,r)\big)} \approx \frac{\#(h,r)}{\#(t,r)} \end{aligned} \tag{6}$$

2.3 The Proof for Proposition 3

Proof. The number of negative samples $N_{(h_1,r)},\ldots,N_{(h_j,r)}$ for Head-Relations in $S_{(r,t)}$, and $N_{(r,t)}$ should follow:

$$\begin{aligned} &\frac{N_{(h_1,r)}}{N_{(t,r)}} = \frac{\#(t,r)}{\#(h_1,r)},\ldots,\frac{N_{(h_j,r)}}{N_{(t,r)}} = \frac{\#(t,r)}{\#(h_j,r)} \\ &N_{(h_1,r)}\#(h_1,r) = \ldots = N_{(h_j,r)}\#(h_j,r) = N_{(t,r)}\#(t,r) \end{aligned} \tag{7}$$

3 Scoring Functions of KGE Models

Table 1. Some KGE models and their scoring functions. $\circ$ denotes the Hadamard (element-wise) product, $\tilde{\cdot}$ denotes a 2D reshaping of $\cdot$, ω is filters of a 2D convolutional layer, and $\mathbf{W}$ is a linear transformation matrix.

Category	Model	Scoring function
TD	TransE	$-\|\mathbf{h}+\mathbf{r}-\mathbf{t}\|_1$
	RotatE	$-\|\mathbf{h}\circ\mathbf{r}-\mathbf{t}\|_1$
SM	DistMult	$\langle\mathbf{h},\mathbf{r},\mathbf{t}\rangle$
	ComplEx	$Re\langle\mathbf{h},\mathbf{r},\bar{\mathbf{t}}\rangle$
CNN	ConvE	$f(vec(f([\tilde{\mathbf{h}};\tilde{\mathbf{r}}]*\omega))\mathbf{W})\mathbf{t}$

4 Results of 10-Fold Cross-Validation

The full results of ERDNS in the 10-fold cross-validation are shown in Table 2. The value of 0.00 is rounded, the actual p-values are extremely small, and most are less than 0.0001.

Table 2. Full Results of ERDNS in the 10-Fold Cross-Validation.

KGE model	Dataset	FB15K237				WN18RR			
	Method	MRR	Hits@1	Hits@3	Hits@10	MRR	Hits@1	Hits@3	Hits@10
ComplEx	ERDNS(1)	42.1	30.6	48.1	64.3	41.4	38.2	42.8	47.6
	ERDNS(2)	41.8	30.4	47.6	63.8	41.6	38.1	43.1	48.4
	ERDNS(3)	41.8	30.4	47.7	63.9	41.9	38.6	43.4	48.3
	ERDNS(4)	41.9	30.5	48.0	64.1	41.0	37.6	42.3	47.8
	ERDNS(5)	41.8	30.3	47.9	64.2	41.0	37.8	42.3	47.2
	ERDNS(6)	41.6	30.2	47.8	63.7	41.6	38.0	42.9	48.5
	ERDNS(7)	41.7	30.2	47.7	63.9	40.6	37.2	42.0	47.2
	ERDNS(8)	42.0	30.5	47.9	64.4	41.3	38.0	42.7	47.8
	ERDNS(9)	41.7	30.3	47.6	64.1	41.3	37.9	42.8	47.8
	ERDNS(10)	41.6	30.3	47.4	63.7	40.8	37.5	42.2	47.3
RotatE	ERDNS(1)	40.6	28.4	47.5	63.6	41.9	37.6	43.2	50.5
	ERDNS(2)	40.2	28.3	47.0	63.2	42.3	38.1	43.5	51.0
	ERDNS(3)	40.4	28.4	46.9	63.3	42.4	38.2	43.8	50.8
	ERDNS(4)	40.4	28.4	47.3	63.2	41.6	37.3	43.0	50.3
	ERDNS(5)	40.5	28.4	47.3	63.3	41.7	37.6	42.9	49.9
	ERDNS(6)	40.2	28.3	47.0	63.1	41.9	37.6	43.1	50.5
	ERDNS(7)	40.4	28.2	47.2	63.4	41.4	36.9	42.8	50.3
	ERDNS(8)	40.6	28.5	47.4	63.4	42.0	37.8	43.3	50.4
	ERDNS(9)	40.3	28.2	47.0	63.4	42.0	37.7	43.4	50.2
	ERDNS(10)	40.2	28.4	47.0	63.1	41.7	37.3	43.4	50.3
ConvE	ERDNS(1)	39.1	28.3	44.1	60.4	34.6	31.5	36.0	40.7
	ERDNS(2)	38.6	27.8	43.7	59.7	35.4	32.2	36.7	41.2
	ERDNS(3)	38.5	27.8	43.6	59.7	34.9	32.0	36.0	40.8
	ERDNS(4)	38.7	27.9	43.7	59.8	34.6	31.7	35.8	40.5
	ERDNS(5)	38.5	27.7	43.5	59.9	34.6	31.8	35.6	40.0
	ERDNS(6)	38.5	27.9	43.5	59.7	35.0	31.9	36.2	41.1
	ERDNS(7)	38.7	27.9	43.7	59.9	34.0	30.9	35.1	40.0
	ERDNS(8)	38.8	28.0	43.8	60.0	34.7	31.5	36.2	40.9
	ERDNS(9)	38.7	28.0	43.6	59.8	34.9	32.0	36.0	40.7
	ERDNS(10)	38.7	28.0	43.5	59.7	34.3	31.2	35.5	40.6

5 Hyperparameters Setting of KGE Models

Table 3. Hyperparameters on DistMult, ComplEx, TransE and RotatE. Dim., Reg., and LR denote the dimension, the regularization, and the learning rate.

Dataset	Batch	Steps	Model	Dim.	Reg.	N_x	α	LR	Model	Dim.	Margin	N_x	α	LR
FB15K237	1024	100000	DistMult	2000	0.000002	256	1	0.001	TransE	1000	9	256	0.5	0.00005
			ComplEx	1000	0.000002	256	1	0.001	RotatE	1000	9	256	0.5	0.00005
WN18RR	512	80000	DistMult	1000	0.000005	50	1	0.002	TransE	500	6	512	1	0.00005
			ComplEx	500	0.000005	128	1	0.002	RotatE	500	6	128	1	0.00005
YAGO3-10	1024	200000	DistMult	1000	0.000002	512	1	0.001	TransE	500	24	1024	1	0.0002
			ComplEx	500	0.000002	512	1	0.001	RotatE	500	24	1024	1	0.0002

Table 4. Hyperparameters on ConvE and CompGCN with decoders (TransE, DistMult, and ConvE). OPN denotes the Composition Operation to be used in CompGCN.

Model	Dataset	Batch	Steps	Dim.	Reg.	N	LR	hidden-drop	input-drop	feat-drop
ConvE	FB15K237	1024	100000	500	0.005	4096	0.0001	0.1	0.3	0.3
	WN18RR	512	80000	500	0.001	4096	0.0001	0.2	0.2	0.2
CompGCN(TransE)	Dataset	Batch	Dim.	Margin	N	LR	OPN	hidden-drop	GCN-drop	GCN-layer
	FB15K237	1024	200	9	4096	0.001	mult	0.2	0.1	1
	WN18RR	512	200	9	1024	0.001	mult	0.2	0.1	1
CompGCN(DistMult)	Dataset	Batch	Dim.	Reg.	N	LR	OPN	hidden-drop	GCN-drop	GCN-layer
	FB15K237	1024	150	0	4096	0.001	mult	0.3	0.1	2
	WN18RR	512	150	0	8192	0.001	mult	0.3	0.1	2
CompGCN(ConvE)	Dataset	Batch	Dim.	Reg.	N	LR	OPN	hidden-drop	GCN-drop	GCN-layer
	FB15K237	1024	200	0	4096	0.001	mult	0.3	0.1	1
	WN18RR	512	200	0	4096	0.001	mult	0.3	0.1	1
	input-drop	feat-drop								
	0.3	0.3								

References

1. Yang, Z., Ding, M., Zhou, C., Yang, H., Zhou, J., Tang, J.: Understanding negative sampling in graph representation learning. In: Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining. pp. 1666–1676 (2020)