Modeling Relational Data with Graph Convolutional Networks

Michael Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, Max Welling

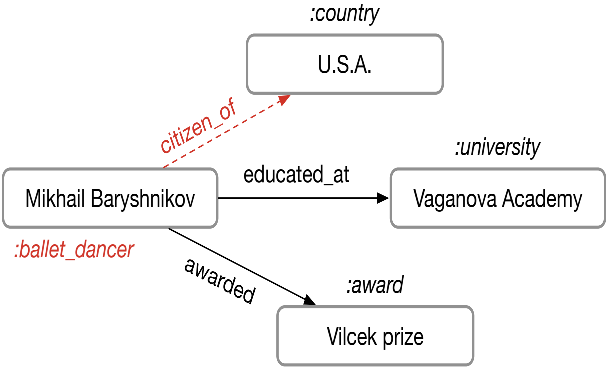
Presenter: Arshdeep Sekhon
https://qdata.github.io/deep2Read

Motivation

- Knowledge graphs are multi-relational data
- completing large databases : lots of missing information in knowledge graphs
- Method: R-GCN
- Statistical relational learning: predicting missing attributes in knowledge bases

Knowledge Graphs

- Knowledge databases store subject-predicate-object
- the subject and object are entities that have types
- Knowledge Graphs are multigraphs

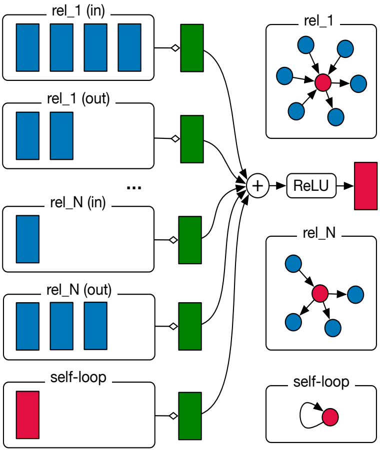


Method: Relational Graph Convolutional Networks

$$h_i^{(l+1)} = \sigma\left(\sum_{r\in\mathcal{R}}\sum_{j\in\mathcal{N}_i^r}\frac{1}{c_{i,r}}W_r^{(l)}h_j^{(l)} + W_0^{(l)}h_i^{(l)}\right), \tag{1}$$

- $\mathcal{N}_i^r$ denotes the set of neighbor indices of node i under relation $r \in \mathcal{R}$.
- $c_{i,r}$ is a problem-specific normalization constant that can either be learned or chosen in advance (such as $c_{i,r} = |\mathcal{N}_i^r|$).

The Model



rel_1 (in)
rel_1 (out)
...
rel_N (in)
rel_N (out)
self-loop
rel_1
ReLU
rel_N
self-loop

Regularization

- As the number of types of relations increases in a graph, number of parameters increase: overfitting
- basis and block-diagonal decomposition
- Basis: Shared basis vectors for all W_r (shared weights)
- Block Diagonal matrices compose to make W_r (sparsity)

Basis

$$W_r^{(l)} = \sum_{b=1}^{B} a_{rb}^{(l)} V_b^{(l)}, \tag{2}$$

effective weight sharing between different relation types

can alleviate overfitting on rare relations, as parameter updates are shared between both rare and more frequent relations.

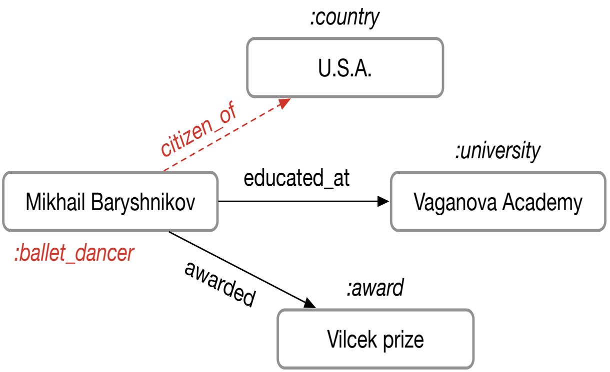
Block Diagonal

$$W_r^{(l)} = \bigoplus_{b=1}^{B} Q_{br}^{(l)}. \tag{3}$$

Thereby, $W_r^{(l)}$ are block-diagonal matrices: $\mathrm{diag}(Q_{1r}^{(l)}, \ldots, Q_{Br}^{(l)})$ with $Q_{br}^{(l)} \in \mathbb{R}^{(d^{(l+1)}/B) \times (d^{(l)}/B)}$.

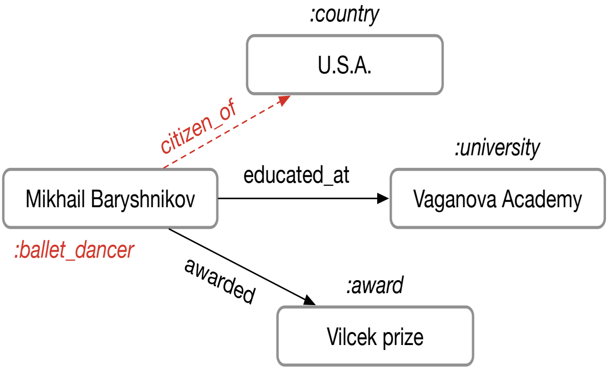
- sparsity constraint on the weight matrices for each relation type
- latent features can be grouped into sets of variables which are more tightly coupled within groups than across groups

Task: Entity Classification



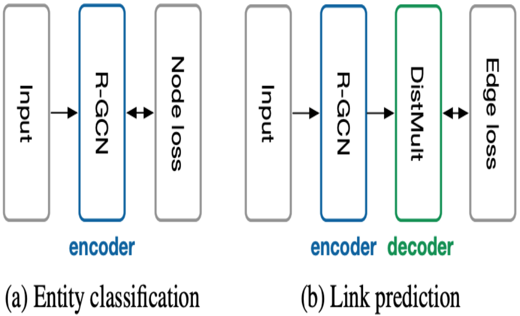
Mikhail Baryshnikov was educated at the Vaganova Academy implies that Mikhail Baryshnikov should have the label person

Task: Link Prediction



knowing that Mikhail Baryshnikov was educated at the Vaganova Academy implies that the triple (Mikhail Baryshnikov, lived in, Russia) must belong to the knowledge graph.

RGCN for Tasks



Task: Entity Classification

- Stack R-GCN layers with softmax layers in the end
- minimize cross entropy loss

$$\mathcal{L} = -\sum_{i \in \mathcal{Y}} \sum_{k=1}^{K} t_{ik} \ln h_{ik}^{(L)} , \tag{4}$$

where $\mathcal{Y}$ is the set of node indices that have labels and $h_{ik}^{(L)}$ is the k-th entry of the network output for the i-th labeled node. t_{ik} denotes its respective ground truth label.

Task: Link Prediction

- Only given $\hat{E}$ from a total number of edges E
- Assign scores to $f(s, r, o)$, how likely is edge s, r, o to belong to E
- Model (a) Encoder (b) Decoder/Scorer

Task: Link Prediction

- Encoder output : each v_i mapped to $e_i \in R^d = h_i^L$
- Decode / score s, r, o $g : R^d \times R \times R^d \to R$
- $f(s, r, o) = e_s^T R_r e_o$ where R_r is a diagonal matrix
- trained with negative sampling by randomly corrupting the subject or the object
- For each observed example we sample ω negative ones
- trained in a supervised manner

$$\mathcal{L} = -\frac{1}{(1+\omega)|\hat{\mathcal{E}}|} \sum_{(s,r,o,y)\in\mathcal{T}} y \log l\big(f(s,r,o)\big) + (1-y)\log\big(1 - l\big(f(s,r,o)\big)\big), \tag{5}$$

Decoder/Scorer: DistMul factorization

$$\mathbf{y}_{e_1} = f\left(\mathbf{W}\mathbf{x}_{e_1}\right), \;\; \mathbf{y}_{e_2} = f\left(\mathbf{W}\mathbf{x}_{e_2}\right)$$

$$g_r^a(\mathbf{y}_{e_1}, \mathbf{y}_{e_2}) = \mathbf{A}_r^T \begin{pmatrix} \mathbf{y}_{e_1} \\ \mathbf{y}_{e_2} \end{pmatrix} \;\text{ and }\; g_r^b(\mathbf{y}_{e_1}, \mathbf{y}_{e_2}) = \mathbf{y}_{e_1}^T \mathbf{B}_r \mathbf{y}_{e_2},$$

$$g_r^b(\mathbf{y}_{e_1}, \mathbf{y}_{e_2}) = \mathbf{y}_{e_1}^T \mathbf{M}_r \mathbf{y}_{e_2}$$

Entity classification experiments

Dataset	AIFB	MUTAG	BGS	AM
Entities	8,285	23,644	333,845	1,666,764
Relations	45	23	103	133
Edges	29,043	74,227	916,199	5,988,321
Labeled	176	340	146	1,000
Classes	4	2	2	11

Datasets

Model	AIFB	MUTAG	BGS	AM
Feat	55.55	77.94	72.41	66.66
WL	80.55	**80.88**	86.20	87.37
RDF2Vec	88.88	67.20	**87.24**	88.33
R-GCN	**95.83**	73.23	83.10	**89.29**

Accuracy

Baselines

- RDF2Vec
- Weisfeiler-Lehman kernel
- hand designed feature extractors

Metrics:

- Mean Reciprocal Rank[1]
- Hits @ n

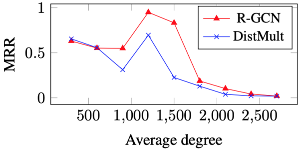
[1]The mean reciprocal rank is a statistic measure for evaluating any process that produces a list of possible responses to a sample of queries, ordered by probability of correctness. The reciprocal rank of a query response is the multiplicative inverse of the rank of the first correct answer: : $\frac{1}{|Q|} \frac{1}{rank_i}$

sample of queries Q

Link Prediction Experiments

Dataset	WN18	FB15K	FB15k-237
Entities	40,943	14,951	14,541
Relations	18	1,345	237
Train edges	141,442	483,142	272,115
Val. edges	5,000	50,000	17,535
Test edges	5,000	59,071	20,466

Datasets



Mean reciprocal rank (MRR) for R-GCN and DistMult on the FB15k validation data as a function of the node degree

Link Prediction

	FB15k					WN18				
	MRR		Hits @			MRR		Hits @		
Model	Raw	Filtered	1	3	10	Raw	Filtered	1	3	10
LinkFeat		0.779			0.804		0.938			0.939
DistMult	0.248	0.634	0.522	0.718	0.814	0.526	0.813	0.701	0.921	0.943
R-GCN	0.251	0.651	0.541	0.736	0.825	0.553	0.814	0.686	0.928	0.955
R-GCN+	**0.262**	**0.696**	**0.601**	**0.760**	**0.842**	0.561	0.819	0.697	0.929	**0.964**
CP*	0.152	0.326	0.219	0.376	0.532	0.075	0.058	0.049	0.080	0.125
TransE*	0.221	0.380	0.231	0.472	0.641	0.335	0.454	0.089	0.823	0.934
HolE**	0.232	0.524	0.402	0.613	0.739	**0.616**	0.938	0.930	**0.945**	0.949
ComplEx*	0.242	0.692	0.599	0.759	0.840	0.587	**0.941**	**0.936**	**0.945**	0.947

Figure: Results on the the Freebase and WordNet datasets

	MRR		Hits @		
Model	Raw	Filtered	1	3	10
LinkFeat		0.063			0.079
DistMult	0.100	0.191	0.106	0.207	0.376
R-GCN	**0.158**	0.248	**0.153**	0.258	0.414
R-GCN+	0.156	**0.249**	0.151	**0.264**	**0.417**
CP	0.080	0.182	0.101	0.197	0.357
TransE	0.144	0.233	0.147	0.263	0.398
HolE	0.124	0.222	0.133	0.253	0.391
ComplEx	0.109	0.201	0.112	0.213	0.388

Figure: Results on FB15k-237