Biomedical Knowledge Graph Embeddings with Negative Statements - Supplementary Material

Rita T. Sousa[1][0000-0002-7241-8970], Sara Silva[1][0000-0001-8223-4799], Heiko Paulheim[2][0000-0003-4386-8195], and Catia Pesquita[1][0000-0002-1847-9393]

[1] LASIGE, Faculdade de Ciências da Universidade de Lisboa, Portugal
{risousa,sgsilva,clpesquita}@ciencias.ulisboa.pt
[2] Data and Web Science Group, Universität Mannheim, Germany
heiko.paulheim@uni-mannheim.de

1 Parameters for Knowledge Graph Embedding Methods

We used nine different knowledge graph embedding approaches as baselines in our work: TransE, TransH, TransR, distMult, DeepWalk, node2vec, metapath2vec, OWL2Vec*, and RDF2Vec. Each method is run with two different KGs, one with only positive statements and one with both positive and negative statements. All of these methods had entity embedding sizes set to 200. Table S1 contains links to the implementations used for the various methods. For DeepWalk, we modified the RDF2Vec implementation to account for the differences between the two methods.

Table S1: GitHub implementations used for the different methods.

Methods	**Implementation**
TransE, TransH, TransR, distMult	`https://github.com/thunlp/OpenKE`
node2vec	`https://github.com/eliorc/node2vec`
metapath2vec	`https://github.com/loginaway/Metapath2vec`
OWL2Vec*	`https://github.com/KRR-Oxford/OWL2Vec-Star`
RDF2Vec	`https://github.com/IBCNServices/pyRDF2Vec`

The parameters for TransE, TransH, TransR, and distMult are shown in Table S2 and were selected as they are provided as examples in the OpenKE toolkit.

The parameters used for DeepWalk, node2vec, metapath2vec, RDF2Vec, OWL2Vec* are outlined in Table S3. Additional parameters for OWL2Vec* are provided in Table S4. However, to the best of our knowledge, the option of declaring inverse axioms is not implemented.

Table S2: Parameters for TransE, TransH, TransR, and distMult.

Parameter	**TransE**	**TransH**	**TransR**	**distMult**
Train times	500	500	500	500
Number batches	100	100	100	100
Learning rate	0.001	0.001	0.001	0.05
Optimization method	SGD	SGD	SGD	Adagrad
Embedding size	200	200	200	200
Entity neg rate	1	1	1	1
Relation neg rate	0	0	0	0
Bern	1	0	–	0
Lambda	0.05	–	4	–
Margin	–	1	1	1

Table S3: Parameters for methods based on random walks: node2vec, metapath2vec, RDF2Vec, OWL2Vec*, and TrueWalks.

Parameter	**Value**
Embedding size	200
Maximum number of walks per entity	100
Maximum depth of walks	4
Word2vec model	skip-gram
Epochs for Word2vec model	5
Window for Word2vec model	5
Minimum count for Word2vec model	1
Optimization for softmax	negative sampling
Number of noise words drawn in negative sampling	5
Learning rate	0.025

Table S4: Parameters for OWL2Vec*.

Parameter	**Value**
Ontology projection	No
Project only taxonomy	No
Avoiding OWL constructs	No
Axiom reasoner	None
URI Doc	Yes
Lit Doc	Yes
Mix Doc	No

2 Parameters for Machine Learning

To run the random forest classifier, we employed scikit-learn [1] to optimize certain parameters, specifically the number of estimators and the maximum depth of the tree. The parameters are supplied in Table S5.

Table S5: Parameters for Random Forest Classifier.

Parameter	**Value**
Number of estimators	50,100,200
Criterion to measure the quality of a split	Gini impurity
Maximum depth of the tree	2,4,6, None
Number of samples required to split	2
Minimum number of samples required to be at a leaf node	1

3 Data sources

The versions (and the link where available) of the files used to construct the KGs are listed below:

- The GO was downloaded on September 2021 and is available at `http://release.geneontology.org/2021-09-01/ontology/index.html`;
- The GO positive annotations were downloaded on January 2021 and is available at `http://release.geneontology.org/2021-01-01/annotations/index.html`;
- The GO negative annotations were downloaded from `https://lab.dessimoz.org/20_not`;
- The HP was downloaded on October 2022, but a link to this version is no longer available. The version we employed is given as part of the supplementary data and code;
- The HP annotations were downloaded on November 2021, but a link to this version is also no longer available. The version we employed is given as part of the supplementary data and code.

4 Statistical Tests

Statistically significant differences between TrueWalks and the other methods are determined using the non-parametric Wilcoxon test at $p < 0.05$. Tables S6 and S7 display the p-values obtained from comparing the other methods with TrueWalks and TrueWalksOE, respectively. TrueWalks performance values are italicized/underlined in Table 2 of the paper when improvements over all other methods are statistically significant, except when comparing TrueWalks with OWL2Vec* for GDA, since in this particular case the improvement is not statistically significant.

References

1. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12**, 2825–2830 (2011)

Table S6: p-values obtained from comparing TrueWalks with the other methods.

	Method	**PPI Prediction**			**GDA Prediction**		
		Precision	**Recall**	**F-measure**	**Precision**	**Recall**	**F-measure**
Pos	TransE	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	TransH	0.0000	0.0000	0.0000	0.0000	0.0002	0.0000
	TransR	0.0000	0.0000	0.0000	0.0000	0.0734	0.0000
	distMult	0.0000	0.0000	0.0000	0.0000	0.0145	0.0000
	DeepWalk	0.0000	0.9997	0.0000	0.0000	0.3457	0.0004
	node2vec	0.0000	0.0000	0.0000	0.0205	0.0977	0.0204
	metapath2vec	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	OWL2Vec*	0.0000	0.0003	0.0000	0.0083	0.7352	0.0532
	RDF2Vec	0.0000	0.7077	0.0000	0.0000	0.0429	0.0000
Pos+Neg	TransE	0.0000	0.0000	0.0000	0.0000	0.0100	0.0000
	TransH	0.0000	0.0000	0.0000	0.0000	0.0011	0.0000
	TransR	0.0000	0.0000	0.0000	0.0015	0.1608	0.0025
	distMult	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	DeepWalk	0.0000	0.9963	0.1268	0.0000	0.0000	0.0000
	node2vec	0.0004	0.0000	0.0000	0.0001	0.3681	0.0004
	metapath2vec	0.0000	0.0000	0.0000	0.0004	0.0222	0.0003
	OWL2Vec*	0.0044	0.1546	0.0044	0.0783	0.0889	0.0338
	RDF2Vec	0.0000	1.0000	0.4795	0.0003	0.8611	0.0012

Table S7: p-values obtained from comparing TrueWalksOA with the other methods.

	Method	**PPI Prediction**			**GDA Prediction**		
		Precision	**Recall**	**F-measure**	**Precision**	**Recall**	**F-measure**
Pos	TransE	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	TransH	0.0000	0.0000	0.0000	0.0000	0.0010	0.0000
	TransR	0.0000	0.0000	0.0000	0.0001	0.0662	0.0001
	distMult	0.0000	0.0000	0.0000	0.0000	0.0415	0.0001
	DeepWalk	0.0000	0.7415	0.0000	0.0009	0.6676	0.0074
	node2vec	0.0000	0.0000	0.0000	0.0360	0.2502	0.0272
	metapath2vec	0.0000	0.0000	0.0000	0.0000	0.0001	0.0000
	OWL2Vec*	0.0000	0.0000	0.0000	0.0239	0.9701	0.0954
	RDF2Vec	0.0000	0.0343	0.0000	0.0004	0.2224	0.0005
Pos+Neg	TransE	0.0000	0.0000	0.0000	0.0001	0.0389	0.0000
	TransH	0.0000	0.0000	0.0000	0.0000	0.0003	0.0000
	TransR	0.0000	0.0000	0.0000	0.0083	0.3892	0.0135
	distMult	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	DeepWalk	0.0003	0.2720	0.0005	0.0000	0.0000	0.0000
	node2vec	0.0000	0.0000	0.0000	0.0005	0.6200	0.0038
	metapath2vec	0.0000	0.0000	0.0000	0.0005	0.0376	0.0005
	OWL2Vec*	0.0008	0.0016	0.0001	0.1495	0.2189	0.1311
	RDF2Vec	0.0000	0.8182	0.0066	0.0018	0.9091	0.0142