Towards Understanding the Impact of Graph Structure on Knowledge Graph Embeddings

Brandon Dave(✉), Antrea Christou, and Cogan Shimizu

Wright State University, Dayton, Ohio, USA
dave.15@wright.edu
{dave.15,christou.2,cogan.shimizu}@wright.edu

Abstract. Knowledge graphs (KGs) are an established paradigm for integrating heterogeneous data and representing knowledge. As such, there are many different methodologies for producing KGs, which span notions of expressivity, and are tailored for different use-cases and domains. Now, as neurosymbolic methods rise in prominence, it is important to understand how the development of KGs according to these methodologies impact downstream tasks, such as link prediction using KG embeddings (KGE). In this paper, we modify FB15k-237 in several ways (e.g., by increasingly including semantic metadata). This significantly changes the graph structure (e.g., centrality). We assess how these changes impact the link prediction task, using six KGE models.

Keywords: Knowledge Graph · Knowledge Graph Embedding · FB15k-237

1 Introduction

Knowledge Graphs (KGs) allow for efficiently integrating heterogeneous data [7,14]. Several techniques for creating KGs exist [6,17], and KGs are frequently evaluated with competency questions (CQs) to assess the qualitative aspects to its construction [2,18]. CQs are designed to fit the intended use-case of the KG; yet, the increasing popularity in integrating KGs into applications necessitate a more comprehensive assessment of quantitative quality. Assessing KGs is essential for subsequent tasks such as KG embeddings (KGEs) [9,11]. Depending on the model, entities and relationships are vectorized, which allow for, e.g., predicting relationships between entities [16]. A recent study revealed that KGE model performance for link prediction can be impacted by the underlying structure of KGs [4].

The work presented in this paper explores varying levels of semantic inclusion and the impact to KGE model performance as a method for quantitative evaluation of KG structure. To the authors' knowledge, there have yet to be any previous comprehensive investigations in this area. The accessibility of KGE models

T. R. Besold et al. (Eds.): NeSy 2024, LNAI 14980, pp. 41–50, 2024.
https://doi.org/10.1007/978-3-031-71170-1_5

are made possible with the use of DGL's DGL-KE library which facilitates the training and evaluation of KGE models [24]. The KG chosen for this research is the widely utilized FB15k-237 KG [21]. This research augments the base entity typing and class hierarchy to these newly described typings; thus, formulating FB15k-238 and FB15k-239, respectively. Concretely, this paper contributes: **(a)** the FB15k isotopes: FB15k-238 and FB15k-239,[1] **(b)** the scripts and configuration files to generate these datasets, **(c)** a thorough evaluation of the effects that the incorporation of increasing metadata has on the performance of the KGE models in the link prediction task[2]; and **(d)** a brief discussion of our initial insights. The research artifacts are provided through a Zenodo repository and[3] a GitHub repository[4] under the MIT License.

1.1 Related Work

The authors were not able to find previous attempts to explore KG structure and its impact on KGE model performance. In [10], Iferroudjene et al. argue that the removal of Freebase *Compound Value Types (CVTs)* from the FB15k and FB15k-237 datasets, consequently, removes valuable information from the KG. They create *FB15k-CVT* that re-introduces an exact subset of Freebase with CVTs, which allow KGs to create more structured and detailed representation of entities with multiple values of a type of data. When evaluating KGE models against FB15k-237 and FB15k-CVT, FB15k-CVT underperformed on link prediction tasks. This work indicates that current KGE models may not effectively incorporate semantic data and additional research can be done to understand the limitations.

Overall, we see that deductive reasoning is quite difficult outside of the symbolic algorithms dedicated to it. In particular, neurosymbolic methods (e.g., as found in [8]) struggle quite a bit. As deductive reasoning is a major hurdle for approaching human-level cognition, this provides further motivation for understanding the impact of how the presence (or lack thereof) of semantic information impacts KGEs.

1.2 Knowledge Graph Embedding Models

We utilize the DGL-KE library for scalable training and evaluation of KGE models[5]. KGE models that implement an additive scoring function can be categorized as *Translational Distance (TrD) Models*. Tested TrD Models include **TransE** [3], **TransR** [12], and **RotatE** [19]. KGE models that apply tensor decomposition (TeD) techniques for scoring can be categorized similarly as *TeD*

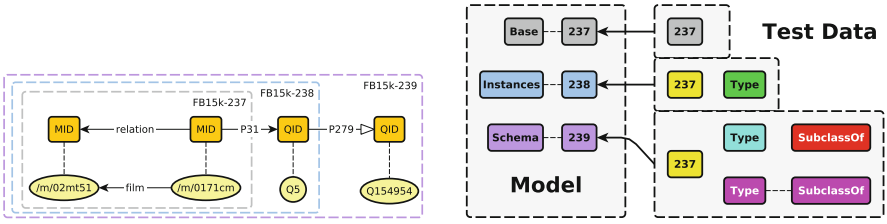
[1] This is intended to be reminiscent of Uranium-238 or Plutonium-239.
[2] For the remainder of the paper, when we say *performance of a model*, we mean specifically for the link prediction task.
[3] https://doi.org/10.5281/zenodo.10296229.
[4] https://github.com/kastle-lab/kge-impact.
[5] https://dglke.dgl.ai/doc/.

Models. Tested TeD Models that fall under this category include **RESCAL** [13], **DistMult** [23], and **ComplEx** [22].

2 Methodology

2.1 Creating FB15k-238 and FB15k-239

FB15k-237 is published with the data split to allow for training, evaluation, and validation of KGE models. This research introduces *FB15k-238* and *FB15k-239* as augmentations of the FB15k-237 dataset. Expanding from FB15k-237, FB15k-238 includes exactly one new relation, P31 (hence the 238). P31 is taken from Wikidata and has the label instanceOf or is a.[6] FB15k-239 extends from FB15k-238 and adds exactly one new relation: P279. P279 is taken from Wikidata and has the label "subclass of."[7] FB15k-238 is constructed by iterating through each Freebase entity (MID) and querying Wikidata for its entity typing via the P31 property. The found facts of P31 are appended to the respective data split files of FB15k-237. FB15k-239 is constructed by iterating through each new entity typing entry from FB15k-238 and querying Wikidata for the entity typing's superclass relationship via the P279 property. The found facts of P279 are appended to the data split files of FB15k-238. We note that not every MID remains incorporated into Wikidata from the original transfer (either they were never transferred or, over time, were for some reason removed).[8] As such, our FB15k-238 dataset is missing the type information for 42 entities.



(a) This represents the types of triples contained in each of the datasets. The yellow ellipses are a set of triples extracted from T_{239}. The dashed boxes correspond to the colors in Figure 1b.

(b) A graphical overview of the different KGE models and their corresponding augmentations. The right hand side shows the different sets of test data used to evaluate the models.

Fig. 1. Graphical overview of adding semantics to FB15k and the method of testing trained models.

[6] https://www.wikidata.org/wiki/Property:P31.

[7] https://www.wikidata.org/wiki/Property:P279.

[8] For example, Freebase MID /m/01sxq9, Wikidata had once listed this entity as "Bebe Neuwirth" but has since removed the MID Property from the respective page. During the initial data migration, technical and non-technical challenges resulted in some missing MIDs in Wikidata [15].

Table 1. This table shows a comparison of different counts for the Freebase subset and the created augmentations.

Dataset	# Entities	# Rel.s	# Train	# Validation	# Test
FB15k-237	14541	237	272115	17535	20466
FB15k-238	16414	238	293471	31482	35257
FB15k-239	17494	239	296822	33879	37738

We provide Table 1 as a summary of the count of entities, edges, and triples per data split in FB15k-237 and our augmentations. Hyper-parameters play a crucial role in training machine learning models, and adjustments to hyper-parameters can result in a variety of results. Due to our inability to collect the hyper-parameters used in the initial publications of the KGE models, we opted to standardize the experimentation across the implemented models of DGL-KE and the augmented datasets by uniforming the hyper-parameters used to train our KGE models. As used by DGL-KE, the list of hyper-parameters[9] are found on Table 2

2.2 Evaluation

The experiment consists of three overall analyses: **(a)** We evaluate the performance of each model across the previously described KGs to examine the respective model's impact to semantic inclusion. In this experiment, the models are trained and evaluated with their respective training and test data. **(b)** We evaluate the performance of each model by training them on their own respective training data. We continue to evaluate them with the test data provided by FB15k-237. This allows for an examination of how models are trained with and without semantics when evaluating data. **(c)** We include an *ablation-like study* which experiments solely with models trained with FB15k-238 and FB15k-239 data (as respectively denoted by M_{238} and M_{239} in Table 4). These models are evaluated with the new data, challenging the models to perform link prediction on the semantics of the KGs.

2.3 Evaluation Metrics

The DGL-KE library provides an evaluation mechanism, configured with **Mean Rank (MR)**, **Mean Reciprocal Rank (MRR)**, and **Hits@K** [1]. **MR** is a statistical metric representing the average position or ordinal rank assigned to a set of items in a given ranking. A lower MR score indicates a better performing model. **MRR** is a statistical measure that assesses the average of the reciprocals of the ranks assigned to relevant items in a ranked list. A higher MRR score, constrained by {0,1}, indicates a better performing model. **Hits@K** is an evaluation metric that measures the number of relevant items present in the top-k

[9] The description of the hyper-parameters can be found at https://dglke.dgl.ai/doc/train.html#arguments.

Table 2. This table details the hyper-parameter settings used for the training and evaluation during training of the KGE models.

Hyper-Parameter	Setting
emb_size	400
max_train_step	500
batch_size	1000
neg_sample_size	1000
learning rate	0.25
gamma	19.9
double_ent	FALSE
double_rel	FALSE
neg_adversarial_sampling	TRUE
adversarial_temperature	1
regularization_coef	1.00E−09
regularization_norm	3

Table 3. This table reports the results of evaluating each of the models against their respective training data. This table reports the results of testing each of the models against solely the FB15k-237 training data (i.e., T_{237}).

Model	Metrics	Evaluation across KGs			Evaluation with T_{237}	
		FB15k-237	FB15k-238	FB15k-239	FB15k-238	FB15k-239
TransE	MRR	0.4143	0.7219	**0.7342**	0.5489	**0.5566**
	MR	33.9947	11.1185	**10.3057**	17.2292	**16.1160**
	HITS@1	0.2982	0.6440	**0.6569**	0.4349	**0.4440**
	HITS@3	0.4701	0.7729	**0.7836**	0.6152	**0.6195**
	HITS@10	0.6394	0.8577	**0.8714**	0.7575	**0.7668**
TransR	MRR	**0.2901**	0.2019	0.2361	0.3037	**0.3058**
	MR	**152.6647**	241.1533	209.9060	138.4772	**128.6830**
	HITS@1	**0.2247**	0.1538	0.1832	0.2401	**0.2404**
	HITS@3	**0.3148**	0.2153	0.2526	0.3266	**0.3293**
	HITS@10	**0.4066**	0.2871	0.3292	0.4188	**0.4226**
ComplEx	MRR	**0.3064**	0.1296	0.1909	0.0840	0.1297
	MR	**84.6207**	225.5332	129.6837	281.7108	179.7169
	HITS@1	**0.2034**	0.0809	0.1152	0.0425	0.0652
	HITS@3	**0.3532**	0.1331	0.2053	0.0859	0.1398
	HITS@10	**0.5001**	0.2206	0.3420	0.1612	0.2565

(*continued*)

positions of a ranked list. A higher value indicates a better performing model. Our evaluation uses k at 1, 3, and 10.

Table 3. (*continued*)

Model	Metrics	Evaluation across KGs			Evaluation with T_{237}	
		FB15k-237	FB15k-238	FB15k-239	FB15k-238	FB15k-239
RESCAL	MRR	0.3520	0.3132	**0.3939**	**0.3771**	0.3766
	MR	126.5442	134.8030	**104.7880**	119.2274	**116.6290**
	HITS@1	0.2766	0.2478	**0.3142**	**0.3108**	0.3055
	HITS@3	0.3884	0.3366	**0.4305**	0.4043	**0.4076**
	HITS@10	0.4789	0.4320	**0.5388**	0.4957	**0.5022**
DistMult	MRR	**0.3213**	0.1644	0.2344	0.1097	0.1537
	MR	**80.7576**	137.8459	108.8363	190.6282	158.1567
	HITS@1	**0.2180**	0.0865	0.1452	0.0508	0.0800
	HITS@3	**0.3672**	0.1785	0.2601	0.1147	0.1696
	HITS@10	**0.5176**	0.3216	0.4122	0.2235	0.2993
RotatE	MRR	**0.0769**	0.0751	0.0629	**0.0775**	0.0664
	MR	**277.2960**	287.1963	298.5854	265.2055	**257.6581**
	HITS@1	**0.0419**	0.0394	0.0344	0.0408	0.0341
	HITS@3	**0.0757**	0.0728	0.0592	**0.0767**	0.0625
	HITS@10	0.1331	**0.1361**	0.1068	**0.1387**	0.1189

3 Results

Table 3 reports the model performances when trained with their respective KGs. Models trained according to a specific FB15k-x are denoted as M_x, where x is the appropriate value. Test datasets are denoted T_x, analogously, and we may find their differences (e.g., $T_{238} - T_{237}$ contains only the entity type triples).

As a space saving measure, the evaluation of FB15k-237 is reported only once, as the second test to compare the various trained models repeat evaluation of FB15k-237 on its own test data. If there are no **bold** reports shown for a particular model, the optimal reported results are from models trained with FB15k-237. If some results are missing underline in the evaluation with T_{237}, the next best results come from models trained on FB15k-237.

Table 4 reports the result of the aforementioned *ablation-like study.*

Table 4. This table reports the results of our ablation-like study, where we change which component of the data against which we evaluate. M_x denotes a model being trained with FB15k-x. T_{x-y} denotes test data, where $x - y$ refers to the set difference resulting in data that can only be found in FB15k-x.

Model	Metrics	$M_{238} \leftarrow$	$M_{239} \leftarrow$		
		$T_{238-237}$	$T_{238-237}$	$T_{239-238}$	$T_{239-237}$
TransE	MRR	0.9590	0.9465	0.9495	0.9470
	MR	2.6546	3.1323	6.0362	3.4810
	HITS@1	0.9280	0.9122	0.9132	0.9128
	HITS@3	0.9910	0.9785	0.9851	0.9799
	HITS@10	0.9970	0.9958	0.9910	0.9955
TransR	MRR	0.0632	0.1643	0.0770	0.1522
	MR	383.1885	254.3162	618.0510	306.3611
	HITS@1	0.0377	0.1215	0.0611	0.1139
	HITS@3	0.0645	0.1744	0.0832	0.1617
	HITS@10	0.1034	0.2377	0.1002	0.2175
ComplEx	MRR	0.1904	0.2250	0.4818	0.2614
	MR	147.8925	75.8214	38.9972	70.3205
	HITS@1	0.1317	0.1347	0.3841	0.1702
	HITS@3	0.1957	0.2433	0.5261	0.2839
	HITS@10	0.3033	0.4047	0.6724	0.4436
DistMult	MRR	0.2399	0.2989	0.5142	0.3316
	MR	64.7679	53.2142	35.7648	50.5354
	HITS@1	0.1350	0.1896	0.4160	0.2245
	HITS@3	0.2652	0.3329	0.5661	0.3685
	HITS@10	0.4593	0.5212	0.6941	0.5479
RESCAL	MRR	0.2238	0.3830	0.5886	0.4127
	MR	156.3768	89.4667	99.9538	90.8974
	HITS@1	0.1583	0.2852	0.5412	0.3223
	HITS@3	0.2437	0.4319	0.6087	0.4571
	HITS@10	0.3440	0.5639	0.6764	0.5801
RotatE	MRR	0.0707	0.0679	0.0049	0.0584
	MR	317.6899	293.1596	669.9831	347.1890
	HITS@1	0.0364	0.0418	0.0012	0.0352
	HITS@3	0.0656	0.0645	0.0020	0.0549
	HITS@10	0.1329	0.1071	0.0068	0.0923

4 Initial Insights

First, across the different isotopes, we see that the inclusion of the additional semantic data drastically improves the performance of **TransE** and **RESCAL**, but otherwise impedes or has a marginal improvement in the other models, when tested with the full training data for each corresponding FB15k isotope. We believe this to largely be the product of the type of relationships being added. For example, **DistMult** works best with symmetric relationships, and neither P31 nor P279 are as such. Treating this work as a more traditional data science problem is slated for immediate next steps.

We also test if the presence of additional semantic metadata present during training improves link prediction *only in the case of non-semantic metadata relations* (i.e., not P31 or P279). For **TransE** and **TransR** this is the case relative to baseline.

The purpose of our ablation-like study is to determine how different training data influences the model and, subsequently, if the end results change for different test data. For example, the $T_{238} - T_{237}$ task may be also called type classification, as we are at this point simply predicting a P31 relation. We note that TransE does exceptionally well, outperforming itself when all data is present, and irrespective of the training data. Yet, this is not the case for any other model. One concern is that the relatively huge presence bias for P31 may be significantly skewing performance. On the otherhand, given that performance still improves when removing said assertions (Table 3, col.s 4–5).

Overall, we see that when looking to improve performance for link prediction, for simple assertional relationships, **TransE** is effective and much improved when semantic metadata is included during training, outperforming all other models.

5 Conclusion

The experiment described in this short paper invites further investigations towards understanding the impact of a KG's schema and KGE model performance. The reports of our experiment suggests a threshold of semantic inclusion exists that can assist in link prediction for all models. We have identified the next areas of research:

1. Replicate the experiment on other benchmarks (e.g., YAGO [20] or WN18RR [3]).
2. Replicate the experiment using additional models (e.g., Deep Learning techniques for KGEs [5]), which may better incorporate semantics.
3. Increase the number of isotopes by adding even more semantic metadata.

Acknowledgement. Antrea Christou and Cogan Shimizu acknowledges funding from the National Science Foundation (NSF) under Grant #2333532; Proto-OKN Theme 3: An Education Gateway for the Proto-OKN. Brandon Dave and Cogan Shimizu acknowledge funding from DAGSI/SOCHE and DAGSI/AFRL under award RX24-26.

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding institutions.

References

1. Akrami, F., Saeef, M.S., Zhang, Q., Hu, W., Li, C.: Realistic re-evaluation of knowledge graph completion methods: an experimental study (2020)
2. Bezerra, C., Freitas, F., Santana da Silva, F.: Evaluating ontologies with competency questions, pp. 284–285, November 2013. https://doi.org/10.1109/WI-IAT.2013.199
3. Bordes, A., Usunier, N., Garcia-Durán, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: Proceedings of the 26th International Conference on Neural Information Processing Systems, NIPS 2013, vol. 2, pp. 2787–2795. Curran Associates Inc., Red Hook (2013)
4. Dave, B., Shimizu, C.: Towards understanding the impact of schema on knowledge graph embeddings (invited) (2023, in press)
5. Dettmers, T., Minervini, P., Stenetorp, P., Riedel, S.: Convolutional 2d knowledge graph embeddings (2018)
6. Fernandez-Lopez, M., Gomez-Perez, A., Juristo, N.: Methontology: from ontological art towards ontological engineering. In: Proceedings of the AAAI97 Spring Symposium, pp. 33–40, March 1997
7. Hitzler, P.: Semantic Web: a review of the field. Comm. ACM (2021, to appear)
8. Hitzler, P., Rayan, R., Zalewski, J., Norouzi, S.S., Eberhart, A., Vasserman, E.Y.: Deep deductive reasoning is a hard deep learning problem (2023, under review)
9. Hogan, A., et al.: Knowledge graphs. ACM Comput. Surv. **54**(4), 71:1–71:37 (2022). https://doi.org/10.1145/3447772
10. Iferroudjene, M., Charpenay, V., Zimmermann, A.: FB15k-CVT: a challenging dataset for knowledge graph embedding models. In: NeSy 2023, 17th International Workshop on Neural-Symbolic Learning and Reasoning, Siena, Italy, pp. 381–394, July 2023. https://hal-emse.ccsd.cnrs.fr/emse-04081543
11. Kejriwal, M., Knoblock, C., Szekely, P.: Knowledge Graphs: Fundamentals, Techniques, and Applications. Adaptive Computation and Machine Learning series. MIT Press (2021). https://books.google.com/books?id=iqvuDwAAQBAJ
12. Lin, Y., Liu, Z., Sun, M., Liu, Y., Zhu, X.: Learning entity and relation embeddings for knowledge graph completion. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 29, no. 1, February 2015. https://doi.org/10.1609/aaai.v29i1.9491
13. Nickel, M., Tresp, V., Kriegel, H.P.: A three-way model for collective learning on multi-relational data. In: Proceedings of the 28th International Conference on International Conference on Machine Learning, ICML 211, pp. 809–816. Omnipress, Madison (2011)
14. Noy, N.F., Gao, Y., Jain, A., Narayanan, A., Patterson, A., Taylor, J.: Industry-scale knowledge graphs: lessons and challenges. Commun. ACM **62**(8), 36–43 (2019). https://doi.org/10.1145/3331166

15. Pellissier Tanon, T., Vrandečić, D., Schaffert, S., Steiner, T., Pintscher, L.: From freebase to wikidata: the great migration. In: Proceedings of the 25th International Conference on World Wide Web, WWW 2016, pp. 1419–1428. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE (2016). https://doi.org/10.1145/2872427.2874809
16. Rossi, A., Barbosa, D., Firmani, D., Matinata, A., Merialdo, P.: Knowledge graph embedding for link prediction: a comparative analysis. ACM Trans. Knowl. Discov. Data **15**(2) (2021). https://doi.org/10.1145/3424672
17. Shimizu, C., Hammar, K., Hitzler, P.: Modular ontology modeling. Semant. Web **14**(3), 459–489 (2023). https://doi.org/10.3233/SW-222886
18. Shimizu, C., et al.: The enslaved ontology 1.0: people of the historic slave trade. Technical report, Michigan State University, East Lansing, Michigan, April 2019
19. Sun, Z., Deng, Z., Nie, J., Tang, J.: Rotate: knowledge graph embedding by relational rotation in complex space. CoRR **abs/1902.10197** (2019). http://arxiv.org/abs/1902.10197
20. Pellissier Tanon, T., Weikum, G., Suchanek, F.: YAGO 4: a reason-able knowledge base. In: Harth, A., et al. (eds.) ESWC 2020. LNCS, vol. 12123, pp. 583–596. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-49461-2_34
21. Toutanova, K., Chen, D.: Observed versus latent features for knowledge base and text inference, July 2015. https://doi.org/10.18653/v1/W15-4007
22. Trouillon, T., Welbl, J., Riedel, S., Éric Gaussier, Bouchard, G.: Complex embeddings for simple link prediction (2016)
23. Yang, B., tau Yih, W., He, X., Gao, J., Deng, L.: Embedding entities and relations for learning and inference in knowledge bases (2015)
24. Zheng, D., et al.: DGL-KE: training knowledge graph embeddings at scale. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2020, pp. 739–748. Association for Computing Machinery, New York (2020)