Experiments in Graph Structure and Knowledge Graph Embeddings

Antrea Christou, Brandon Dave and Cogan Shimizu
Department of Engineering and Computer Science, Wright State University, OH, USA
E-mails: christou.2@wright.edu, dave.15@wright.edu, 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 examine how various perturbations of graph structures impact downstream tasks. These perturbations are sourced from how various methodologies (or design practices) would impact the model, starting with simple inclusions of schema and basic reification constructions. We assess these changes across synthetic graphs and FB15k-237, a common benchmark. We provide visualizations, graph metrics, and performance on the link prediction task as exploration results using various KGE models.

Keywords: Knowledge Graphs, Knowledge Graph Embeddings, Graph Metrics

1. Introduction

Knowledge graphs (KGs) are an established paradigm for effectively and efficiently integrating heterogeneous data [21, 24, 37]. Many methodologies for creating KGs (and the ontologies that act as their schema [22]) have been developed over the years [12], which recommend or otherwise emphasize the use of various techniques. These range from the use of upper ontologies [14, 46], the use of ontology design patterns [5, 15, 43], or even the use of LLMs alone [35], or combined with other methods [45].

Evaluating of KGs (or the ontologies that act as their schemas) can be done in many ways [17, 40], including the use of large language models [50], logical and mathematical characteristics [18], heuristics [39], or competency questions [34]. On the other hand, validation tools (e.g., SHACL [29] or ShEx [4]) can measure whether or not the KG adheres to the schema.

As such, these also widely vary along which dimensions the evaluation occurs (e.g., is the ontology well-formed?) and how the quality is reported (i.e., quantitative or qualitative reporting). Of particular importance, in any case, is determining whether or not the resultant KG after executing a methodology indeed serves the needs of the stakeholders. For example, competency questions act as both a guide during the development (in many methodologies) and also as a mechanism to confirm if the KG appropriately models – and returns – the correct data [3].

Beyond these particular assessments of quality, however, is also whether or not a KG is appropriate for *downstream* tasks, such as KG embeddings (KGEs) [28]. Depending on the model, entities and relationships are somehow vectorized, which allow, for example, predicting relationships between entities [42]. A recent study revealed that KGE model performance for link prediction can be impacted by the underlying structure of KGs [9].

The work presented in this paper explores how various graphical structures, such as those that would produced via different KG or ontology development methodologies, impact how various KGE models are impacted by those changes, when evaluated against the link prediction task. To the authors' knowledge, beyond their own work [8] that this paper extends, there has yet to be any comprehensive investigation in this area (although recently a pipeline for *conducting* such assessment has been developed [20]).

Specifically,

- instance graph, which we call SKG-4;
- SKG-4 plus type annotations, which we call SKG-5;
- SKG-5 plus superclasses for each type, which we call SKG-6;
- SKG-5 with reified properties, which we call SKG-5r;
- SKG-5r with shortcuts, which we call SKG-5rs; and
- SKG-5rs with added contextual nodes, which we call SKG-5rsc.

We furthermore note that these various representations span complexity. On one hand, they represent a richer ontological reality, but on the other hand simpler semantics (and thus KG structures) are easier to consume and query. This is inline with how patterns can be used to flatten or expand *views* of data to aid in data publishing and consumption [30, 31, 41]. Similarly, it is worth exploring how the various *views* over a KG can be used for human consumption, but tied directly to a version that is easier for different KGE models to learn.

Concretely, this paper contributes: **(a)** various synthetic graphs and mechanism for their generation. **(b)** the FB15k isotopes: FB15k-238 and FB15k-239,[1] **(c)** the scripts and configuration files to generate these datasets, **(d)** 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], **(e)** the creation of SKG-237, a graph mimicking FB15k-237 in structure as far as node, edge , node/ratio and degree centrality, that is trained and validated in the same way as the ones above on TransE, **(f)** the creation of synthetic knowledge graphs (SKGs) of increasing complexity, showcasing their generation, training, and evaluation on different hyperparameters from our original isotopes, along with visualizations using t-SNE and UMAP. and **(g)** a discussion of results and insights.

2. Background

2.1. Related Work

In [25], 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 [23]) 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.

The importance of evaluating knowledge graph embeddings (KGEs) with respect to the underlying semantics of the graph is brought up in recent research. When evaluating embedding performance, for instance, Jain et al. [26] mentions the importance of evaluating how well these embeddings preserve the semantic links within the knowledge graph in addition to using ordinary metrics. Our goal of understanding embedding behavior in synthetic KGs is in support of this. Gutierrez et al. work [19] additionally points out the significance of matching vector space representations to basic ontology rules and terminology, claiming that

[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.

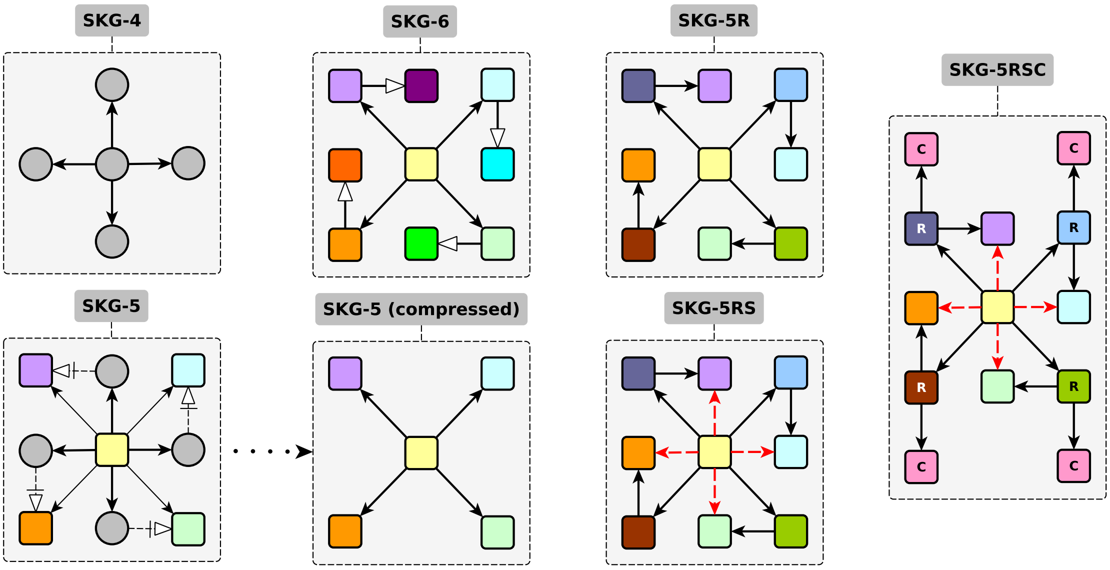


Fig. 1. This figure shows the various schema diagrams for the synthetic KG isotopes. We have used consistently coloring across all figures to demonstrate correspondence. For clarity, in SKG-5RSC, we denote *contextual* nodes with a C and *reification* nodes with an R.

a more thorough examination of how well embedding methods work with complex semantic structures is essential—a realization that directs our investigation of synthetic knowledge graphs.

Additionally, Kang et al. [27] showed how conditional information can make connections in data more clear, which motivated us to use t-SNE visualizations to uncover important patterns in our embeddings. Their research on demonstrating dataset properties guided our approach for using these visualizations to identify patterns, improve our understanding of groups, and identify data cluster divisions. Building on this visualization method, Damrich et al. [7] reveals how UMAP and t-SNE can be used to effectively study high-dimensional data. Their usage of similar learning methods to modify embeddings offers a helpful perspective on how visualizing models such as TransE might highlight structural ties in the data, which we apply into our own visualizations of our synthetic KGs.

2.2. *Knowledge Graph Embedding Models*

We utilize the DGL-KE library for scalable training and evaluation of KGE models [3]. KGE models that implement an additive scoring function can be categorized as *Translational Distance (TrD) Models*. Tested TrD Models include **TransE** [6], **TransR** [32], and **RotatE** [47]. KGE models that apply tensor decomposition (TeD) techniques for scoring can be categorized similarly as *TeD Models*. Tested TeD Models that fall under this category include **RESCAL** [36], **DistMult** [52], and **ComplEx** [49].

3. Methodology

In this section, we describe how we created the various synthetic KG and FB15k isotopes developed for our evaluation. Specific implementation details, including hyper-parameters, are detailed in Section 3.4.

3.1. *Creating SKG-4, SKG-5 and SKG-6*

We created a total of six synthetic datasets to further our investigation regarding the graph structure of a KG and how that may affect the link prediction aspect of KGEs. The structure of, or template for, each of these synthetic KGs (SKGs) is shown in Figure 1.

We describe the *templates* for the SKG isotopes.

[3]https://dglke.dgl.ai/doc/

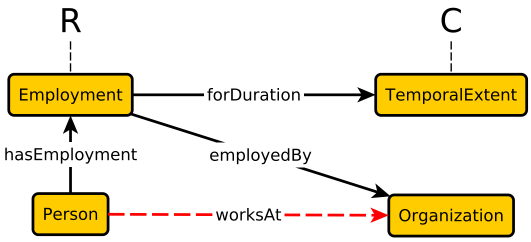
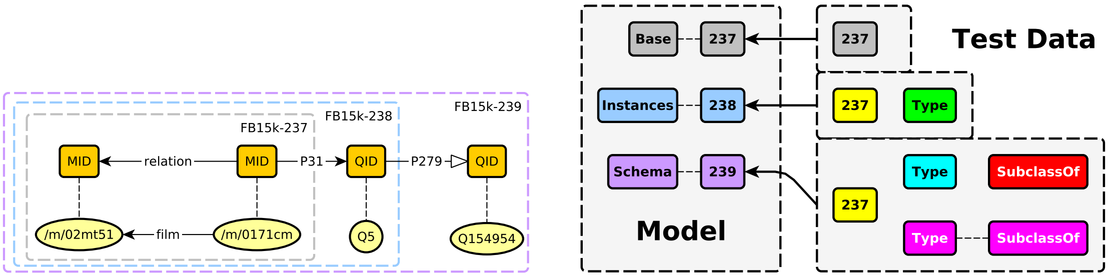


Fig. 2. This figure shows a schema diagram modeling a reified node. It includes a contextualization (the



(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 3b.

(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. 3. Graphical overview of adding semantics to FB15k and the method of testing trained models.

- **SKG-4:** This isotope consists of a central node connected to four peripheral nodes via unique properties. That there are four unique properties in this isotope gives meaning to the numerical assignment (i.e., SKG-*4*).
- **SKG-5:** Each node in the template for SKG-4 is assigned a type. In doing so, we introduce a new predicate (rdf:type) and thus increase the isotope numeral. The type remains constant (and thus tied to) the unique property. This is depicted using a consistent color in Figure 1. For example, the top node in SKG-4 always has the "lavender" type.
- **SKG-6:** Each type node in SKG-5 is assigned a corresponding superclass. In doing so, we introduce another new predicate (rdfs:subClassOf). In Figure 1, this corresponds to the "lavender" type having "purple" superclass.
- **SKG-5r:** This isotope is built from SKG-5. Essentially, each unique property is no reified with a consistently typed node. Reification can be interpreted a few different ways [13, 16]. In this case, intend it to be a node that will somehow be used to attach context to a property. A modeling example is show in Figure 2. This metapattern can be viewed in detail in [44].
- **SKG-5rs:** For each reified property, we include the original unique property, thus providing a *shortcut* back to the original target node. This is the red dashed lines in Figure 1.
- **SKG-5rsc:** For each reification node, we attach a contextual node of specific type. This is labeled C in the figure.

In this study, we currently instantiate each template 1,000 times. This can be improved in the future to produce templates that interlink or somehow connect via nodes. As it stands, each SKG has 1,000 disconnected components.

3.2. 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

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

Table 1

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

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 2

This table details the hyper-parameter settings used for the training and evaluation during training of the KGE models with respect to FB15k-237, FB15k-238, and FB15k-239.

Hyper-Parameter	Setting-SKG
max_train_step	50
batch_size	10
neg_sample_size	10
learning rate	0.25
gamma	19.9
hidden_dim	10
log_interval	25
reg_coeff	1.00E-05

Table 3

This table details the standardized lowest hyper-parameter settings used for the traini the SKGs.

Wikidata and has the label instanceOf or is a.[4]. FB15k-239 extends from FB15k-238 and adds exactly one new relation: P279. P279 is taken from Wikidata and has the label "subclass of."[5] 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).[6] As such, our FB15k-238 dataset is missing the type information for 42 entities.

We provide Table 1 as a summary of the count of entities, edges, and triples per data split in FB15k-237 and our augmentations.

3.3. Creating SKG-237

We constructed a synthetic graph with the same number of unique nodes, unique predicates, and triple count as FB15k-237. However, the exact *facts* are not. Instead, triples are added in such a way to create synthetic version of FB15k-237 (which we creatively call SKG-237) that has the same graph centrality metrics: degree, betweenness, and closeness.

3.4. Implementation

Our graphs are generated using a set of scripts which can be found online. Research artifacts include the scripts for generating the SKG and FB15k isotopes, for calculating the ratio and centrality metrics, for generating the visualizations, and a container for training the KGE models, as well as each of the graphs

[4]https://www.wikidata.org/wiki/Property:P31
[5]https://www.wikidata.org/wiki/Property:P279
[6]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 [38].

themselves. They are provided through a Zenodo repository[7] and a GitHub repository[8] under the MIT License, which is also included in the repository.

The KGE models are trained through the Deep Graph Learning - Knowledge Embedding (DGL-KE) library [53].

Hyper-parameters play a crucial role in training machine learning models, and adjustments to hyper-parameters have a sizable impact on model performance, choosing them for knowledge graph embedding model training is a difficult but important issue [33].

Due to the smaller size of these synthetic KGs, we had different hyper-parameter configurations for them, due to incompatibilities between the graph size and the DGL-KE configuration. Further, we were not able to identify the hyper-parameters used in the initial publications of the KGE models, so we opted to standardize their values across our experimentation with the implemented models in DGL-KE. As used by DGL-KE, the list of hyper-parameters[9] are found in Table 2.

3.5. Evaluation

The experiment consists of four 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 10). These models are evaluated with the new data, challenging the models to perform link prediction on the semantics of the KGs.

(d) With using the same hyperparameters shown in table 2, for training and evaluation of SKG-237 to examine how the performance would be since the graphs are only the same in structure. We also include the t-sne and umap figures.

(e) We used drastically different hyperparameters from our original approach shown in 3 for training and evaluation on SKG-4, SKG-5,SKG-5r, SKG-5rs, SKG-5rsc and SKG-6 for the TransE model in order to examine how our isotope variations affect the link prediction aspect of learning when that structure is controlled.

As a straightforward and widely used model that offers a clear baseline for evaluating embedding accuracy and link prediction performance, we chose to focus on TransE as our starting point. TransE's translation-based approach fits in well with our goal of investigating how structural features affect model behavior, and initial testing showed that it is very sensitive to graph structure changes. To compare the synthetic knowledge graph with complex models created in future investigations, it was the perfect place to start when evaluating how well it represents underlying relationships.

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

[7] https://doi.org/10.5281/zenodo.10296229
[8] https://github.com/kastle-lab/kge-impact
[9] The description of the hyper-parameters can be found at https://dglke.dgl.ai/doc/train.html#arguments

3.7. Graph Metrics

Important insights into the dynamics and structure of the underlying data are obtained by investigating graph metrics in the context of KGs. The following explains each metric's selection and the implications of each datasets investigation.

The number of edges, nodes, and facts: The basic indicators of a KG's size and scope. It is easier to measure the complexity and range of the knowledge graph when one is aware of the quantity of facts (triples), nodes (entities), and edges (relationships). Examining these parameters enables us to classify KGs according on their sparsity and density. A dataset with a high degree of relation is shown by a graph with a large number of edges in relation to its nodes. On the other hand, a lower percentage can indicate a more domain-specific or scattered KG. Understanding the KG's size helps in adjusting procedures to the properties of the data, minimizing errors and increasing efficiency in applications like link prediction and node categorization [11].

Degree Centrality: Degree centrality provides an simple yet effective method of identifying significant nodes in the connections by counting the number of relationships that each node has. Knowing the degree distribution makes it easier to identify the entities in the graph that may be more important. In applications where integration is crucial, such as recommender systems or data retrieval, high-degree nodes can serve as indicators of significant concepts or entities. In domain-specific analysis, low-degree nodes can be helpful in identifying specialized or less well-known topics [11].

Inbetweeness Centrality: This metric measures how far a node is from other nodes along paths, indicating that it serves as a link in the network. Nodes with a high betweenness centrality are those that control the network's information flow. These nodes may serve as crucial connections or limitations, which is essential to comprehending the paths via which knowledge spreads. Examining betweenness centrality can help identify entities that could interfere with how the graph connects if eliminated. For operations involving pathway analysis or locating vulnerable positions in the graph, this is highly relevant [11].

Closeness Centrality: This metric quantifies a node's closeness to every other node in the graph, indicating whether efficiently information can get to it. Relevance: High closeness centrality nodes are positioned to enable fast access to data from the whole graph. This aids in locating basic concepts that serve as points of focus for the distribution or retrieval of information. In applications like query-answering systems or KG-based search engines, where immediate access to scattered information is essential, knowing which nodes have high proximity centrality can help with decision-making processes [11].

4. Results

We report our results along three dimensions, the graph centrality metrics, KGE model performance on the link prediction task (including both the evaluation for the SKG isotopes and the ablation-like study for FB15k isotopes), and visualizations using t-SNE [27] and UMAP [7].

4.1. Graph Metrics of the Isotopes

Table 4, Table 6 and Table 7 present important metrics, such as the total number of facts, nodes, edges, and edge-to-node ratio, for the datasets and synthetic KGs. Additionally, they offers broad information for degree centrality, betweenness centrality, and closeness centrality, displaying the average, maximum, and minimum values for each of these metrics across the datasets.

As can be seen at table 5, even though SKG-237 is the same in structure with FB15k-237 as far as node, fact, edges and degree centrality values, the train/evaluation and visualization results are so different as shown in 11.

4.2. KGE Evaluation Results

Table 8 refers to the evaluation results of TransE.

Metric	FB15k-237	FB15k-238	FB15k-239
Total Facts	310114	360210	368439
Number of Nodes	14541	16414	17507
Number of Edges	248608	270023	273403
Edge-to-Node Ratio	17.10	16.45	15.62
Degree Centrality (Min)	6.90E-05	6.10E-05	5.70E-05
Degree Centrality (Max)	0.4782	0.4240	0.3975
Degree Centrality (Avg)	0.0024	0.0020	0.0018
Betweenness Centrality (Min)	0.00	0.00	0.00
Betweenness Centrality (Max)	0.2837	0.2555	0.2477
Betweenness Centrality (Avg)	0.0001	0.0001	0.0001
Closeness Centrality (Min)	0.0001	0.0001	0.1123
Closeness Centrality (Max)	0.6155	0.5681	0.5304
Closeness Centrality (Avg)	0.3732	0.3473	0.3261

Table 4

The table reports the graph metrics for FB15k-237, FB15k-238, FB15k-239.

Metric	SKG-237
Total Facts	310114
Number of Nodes	14541
Number of Edges	248608
Edge-to-Node Ratio	17.10
Degree Centrality (Min)	6.90E-05
Degree Centrality (Max)	0.4782
Degree Centrality (Avg)	0.0024
Betweenness Centrality (Min)	0.00
Betweenness Centrality (Max)	0.0100
Betweenness Centrality (Avg)	0.0001
Closeness Centrality (Min)	0.2234
Closeness Centrality (Max)	0.3720
Closeness Centrality (Avg)	0.2757

Table 5

The table reports the graph metrics for SKG-237.

Metric	SKG-4	SKG-5	SKG-6	Difference
Total Facts	4000	9000	9004	↑
Number of Nodes	5000	5005	5009	↑
Number of Edges	4000	9000	9004	↑
Edge-to-Node Ratio	0.80	1.80	1.80	↑
Degree Centrality (Min)	0.0002	0.0004	0.0002	↑ and ↓
Degree Centrality (Max)	0.0008	0.1998	0.1999	↑
Degree Centrality (Avg)	0.0003	0.0007	0.0007	↑
Betweenness Centrality (Min)	0.00	0.0001	0.00	↑ and ↓
Betweenness Centrality (Max)	0.00	0.3591	0.3586	↑ and ↓
Betweenness Centrality (Avg)	0.00	0.0004	0.0004	↑
Closeness Centrality (Min)	0.0046	0.2962	0.2920	↑
Closeness Centrality (Max)	0.0005	0.2943	0.2939	↑
Closeness Centrality (Avg)	0.0008	0.5553	0.5547	↑ and ↓

Table 6

The table reports the graph metrics for SKG-4, SKG-5, SKG-6.

Metric	SKG-4	SKG-5	SKG-5r	SKG-5rs	SKG-5rsc	Difference
Total Facts	4000	9000	13000	17000	25001	↑
Number of Nodes	5000	5005	9005	9005	13007	↑
Number of Edges	4000	9000	13000	17000	25001	↑
Edge-to-Node Ratio	0.80	1.80	1.44	1.89	1.92	↑
Degree Centrality (Min)	0.0002	0.0004	0.0001	0.0002	7.70E-05	↑ and ↑
Degree Centrality (Max)	0.0008	0.1998	0.1111	0.1111	0.3076	↑
Degree Centrality (Avg)	0.0003	0.0007	0.0003	0.00042	0.0003	↑
Betweenness Centrality (Min)	0.00	0.0001	0.00	0.00	0.00	↑ and ↓
Betweenness Centrality (Max)	0.00	0.3591	0.3081	0.5299	0.4748	↑ and ↓
Betweenness Centrality (Avg)	0.00	0.0004	0.0003	0.0003	0.0002	↑
Closeness Centrality (Min)	0.0046	0.2962	0.2047	0.2649	0.2602	↑
Closeness Centrality (Max)	0.0005	0.2943	0.4285	0.5293	0.4813	↑
Closeness Centrality (Avg)	0.0008	0.5553	0.2388	0.2814	0.2949	↑ and ↓

Table 7

The table reports the graph metrics for SKG-4, SKG-5, SKG-5r,SKG-5rs,SKG-5rsc.

Model	Metrics	SKG-4	SKG-5	SKG-6	SKG-5r	SKG-5rs	SKG-5rsc
TransE	MRR	0.3068	0.2311	0.2366	0.2486	0.2410	0.2482
	MR	5.6925	6.8772	6.7836	6.5969	6.6403	6.6325
	HITS@1	0.1275	0.0683	0.06992	0.0800	0.0682	0.07620
	HITS@3	0.2963	0.1972	0.2109	0.2269	0.2275	0.2371
	HITS@10	0.9163	0.8678	0.8574	0.8881	0.8541	0.8419

Table 8

This table reports the performance results for the SKG isotopes using TransE with the standardized hyperparameters.

Table 9 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 10 reports the result of the aforementioned *ablation-like study*.

4.3. Visualizations of TransE Embeddings

The figures are included in the Appendix so as to not overwhelm the narrative.

5. Discussion

5.1. KGE Performance over SKG Isotopes

For SKG-4, which has no hierarchical relationships or any sort of additional "semantic complexity," shows the best results overall. We note that these values will still be limited due to the low connectivity between template structure instantiations. Yet, when we begin adding semantic annotations (in the form of rdf:type triples) which should thereby connect

SKG-5 and its variations, adding reification, shortcut and contextual information show promising results. For SKG-5, MRR drops showing a struggle in the prediction task but Hits@10 demonstrates that it eventually can achieve satisfactory numbers.

Model	Metrics	Evaluation across KGs FB15k-237	FB15k-238	FB15k-239	Evaluation with T_{237} 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
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

Table 9

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}).

Adding the reification relationship adds on the structural depth but makes it difficult to perform, the same way with the shortcut addition. However, the contextual aspect of the information assists on the link prediction capabilities.

Additional exploration, especially in the style of the ablation-like study below, for training between

5.2. KGE Performance over FB15k Isotopes

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.

Model	Metrics	$M_{238} \leftarrow$ $T_{238-237}$	$M_{239} \leftarrow$ $T_{238-237}$	$T_{239-238}$	$T_{239-237}$
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

Table 10

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.

Metric	SKG-237
MRR	0.01209
MR	309.91
Hits1	0.0018
Hits3	0.0058
Hits10	0.0186

Table 11

This table reports the results of our evaluation of SKG-237 for TransE, using the standardized hyperparameters. We note a *drastically* negative impact on performance.

5.3. KGE Performance over SKG-237

Despite utilizing a KG that has the same structural characteristics as FB15k-237 (nodes, predicates, and triples), the TransE model does not do well on link prediction, according to the results . Low HITS@1, HITS@3, and HITS@10 scores, along with poor Mean Reciprocal Rank (MRR) and Mean Rank (MR) values, show that the model has struggled properly ranking pertinent entities, even among the top 10 predictions. This suggests that although the synthetic KG shares structural similarities with FB15k-237, it does not contain of the semantic relationships that underlie the original dataset. Thus, we note, that to some extent TransE requires that the KG indeed more closely mimic real-world data. Further exploration is required to determine the exact connection between recurring entities in triple and the appearance of entities consistently in appropriate domains and ranges of relations. That is to say, that we suspect in order for a KG to be TransE-learnable, a minimum semantics is required in the graph.

5.4. Ablation-like Study with FB15k Isotopes

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 other hand, given that performance still improves when removing said assertions (Tab. 9, 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.

5.5. Discussion of Graph Metrics

The graphs for SKG-4, SKG-5, and SKG-6 are getting more complex as reflected by their increased nodes, edges, and edge-to-node ratio, which indicates their higher degree of connectivity. Degree and betweenness centrality show that center nodes that are important are becoming more frequent, even while many nodes are still less connected. Nodes become easier to locate in SKG-5, but somewhat less so in SKG-6, according to closeness centrality.

When reification relationships (r), shortcuts (s), and contextual information (c) are added, the metrics for SKG-4 through SKG-5rsc clearly show a pattern of a growing complexity. A significant increase is seen in the overall number of facts, nodes, and edges; denser graphs are demonstrated by a higher edge-to-node ratio. There is a range of degree centrality values, with some nodes growing closer together while others stay just moderately connected. Although betweenness centrality points to the rise in important nodes, especially in SKG-5rs, overall average values are still low, suggesting that there is no dominant centralization. As shortcuts and context are introduced, nodes become easier to access, thus increasing total graph connection, according to closeness centrality values. These trends demonstrate the graphs' increasing structural changes and depth as more semantic layers are added.

With the most facts, nodes, and edges, FB15k-239 has the largest graph structures, according to the metrics. In bigger datasets, the edge-to-node ratio drops from FB15k-237 to FB15k-239, indicating a lower relative graph density. While average values drop across the datasets, degree centrality measurements indicate that FB15k-237 has a greater range with higher maximum values, suggesting better balanced connection in larger graphs. Although betweenness centrality varies throughout the datasets, FB15k-237 has somewhat higher maximum values, suggesting that some nodes are essential for connecting. More direct node interaction is suggested by FB15k-237's greater maximum and average closeness centrality scores.

So far we managed to replicate the graph structure of FB15k-237, as far as node, edge count, node/edge ratio and degree centrality.

Most nodes don not act as important information-transfer facilitators, given the low average and maximum betweenness centrality numbers, thereby pointing to a graph structure in which no single node controls the shortest paths.

Also, nodes appear to be in a similar location with respect to their average distance to every other node, based on the very small variety of closeness centrality values. This suggests that a lot of nodes in the graph are relatively easy to find, indicating a balanced connectivity pattern.

5.6. Discussion of Visualization Results

The distribution of the training embeddings for FB15k-237, FB15k-238 and FB15k-239 can be seen in Figures **??**, **??** and **??**, showing discrete clusters within each dataset. The visualizations show distinct regions with dense node clusters and with minimal areas with scattered nodes. This suggests that the model has effectively discovered significant connections between the knowledge graph's entities.

The entities and relations are typically well-separated, suggesting that the model obtained distinct representations of different entity and relation types. The relationships between entities in the embedding space appear to be implied by the model, as shown by the red crosses that depict relations appearing between clusters.

TransE t-SNE and UMAP visualizations are showcased in **??** and **??**. Notably the nature of the clustering is quite different. Of course, both t-SNE and UMAP are not appropriate for strictly defining cluster membership, they can give an understanding of what clusters might exist. In this case, we might inspect that the centrality metrics for SKG-237 are misleading. Future work should include different investigations to the centrality metric beyond the average, per se.

The dense mixing observed here may lead to the model miss-ranking predictions because of similar embeddings for different entities. As opposed to this, the UMAP plot in **??** displays elements within close clusters, although this separation might simply represent key structural differences rather than expressing the more complicated semantic connections required for accurate predictions.

Plots for the SKG isotopes are shown in Figures **??**-**??**, displaying the TransE training results and giving us an insight on the overall clustering and that the lack of interconnections between template structure instantiations has a negative impact. Yet, in higher isotopes, we also notice a distinct lack of clustering based on type (i.e., that consistent use of type for the range of a property does not seem to overtly influence the distribution of embeddings).

The t-SNE and UMAP visualizations both demonstrate the creation of clusters that is similar for SKG-4 seen in figure **??**, suggesting that entities with similar semantic properties are grouped together. Based on embedding values, the color coding indicates that different entities have different semantic characteristics. Along with SKG-5 and SKG-6 shown in figure **??** and **??** respectively, the clusters show the most separation, confirming the results we discussed above about the highest evaluation results.

The rest of the visualizations of the extended versions of SKG-5 that are SKG-5r/5rs/5rsc in figures **??**, **??** and **??** respectively, show many small tight clusters again reflecting the evaluation results.

6. Conclusion

Surprisingly, creating SKG-237 with exact triple, node, predicate and degree centrality number as FB15k-237 was not enough as a controlled environment in terms of training and evaluation results.

The semantic connections represented in the synthetic graph may not capture the complex patterns found in FB15k-237, despite its structural features (such as node/edge counts and centrality measurements) being identical. Hence, we took the next step in creating SKG-4/5/6 and the variations of SKG-5 (SKG-5/5r/5rs/5rsc).

In summary, controlling graph structure has yielded important information on KGE's performance. With its simple structure, SKG-4 provides the best link prediction results, indicating that model performance is improved by minimal complexity. While adding complexity enhances semantic depth, it also makes prediction more difficult. This is true for SKG-5 and its modified versions, which include reification, shortcuts, and contextual information.

Simpler structure, such as SKG-4, are shown to form more distinct clusters, while more complicated graphs create a balance between relationships and group formation. According to these findings, adjusting graph complexity affects how effective KGEs are; simplicity and structural depth must be balanced. 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. Understanding the effects of graph metrics and structure on embedding outcomes is essential. Getting the best results out of knowledge graph embeddings can be complex as demonstrated by the impact of these metrics and the careful tuning of training and evaluation parameters.

Future Work

We have identified some next steps in this line of research:

1. Replicate the experiment on other benchmarks (e.g., YAGO [48] or WN18RR [6]).
2. Replicate the experiment using additional models (e.g., Deep Learning techniques for KGEs [10]), which may better incorporate semantics, as well as establish no differences between implementations (e.g., [2])
3. Increase the number of tested isotopes by adding even more semantic metadata and varying graph structures.
4. Examine the impact on other downstream tasks (e.g., entity clustering [51]).

Acknowledgement. Antrea Christou and Cogan Shimizu acknowledges support 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 support 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 supporting 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] Ali, M., Berrendorf, M., Hoyt, C.T., Vermue, L., Sharifzadeh, S., Tresp, V., Lehmann, J.: PyKEEN 1.0: A Python Library for Training and Evaluating Knowledge Graph Embeddings. Journal of Machine Learning Research **22**(82), 1–6 (2021), http://jmlr.org/papers/v22/20-825.html

[3] Antia, M.J., Keet, C.: Automating the generation of competency questions for ontologies with agocqs (10 2023).

[4] Baker, T., Prud'hommeaux, E. (eds.): Shape Expressions (ShEx) 2.1 Primer. Final Community Group Report 09 October 2019 (2019), http://shex.io/shex-primer/index.html

[5] Blomqvist, E., Hammar, K., Presutti, V.: Engineering Ontologies with Patterns – The eXtreme Design Methodology. In: Hitzler, P., Gangemi, A., Janowicz, K., Krisnadhi, A., Presutti, V. (eds.) Ontology Engineering with Ontology Design Patterns – Foundations and Applications, Studies on the Semantic Web, vol. 25, pp. 23–50. IOS Press (2016)

[6] 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 - Volume 2. p. 2787–2795. NIPS'13, Curran Associates Inc., Red Hook, NY, USA (2013)

[7] Damrich, S., Böhm, J.N., Hamprecht, F.A., Kobak, D.: From *t*-sne to umap with contrastive learning. arXiv preprint arXiv:2206.01816 (2022)

[8] Dave, B., Christou, A., Shimizu, C.: Towards understanding the impact of graph structure on knowledge graph embeddings. In: Besold, T.R., d'Avila Garcez, A., Jiménez-Ruiz, E., Confalonieri, R., Madhyastha, P., Wagner, B. (eds.) Neural-Symbolic Learning and Reasoning - 18th International Conference, NeSy 2024, Barcelona, Spain, September 9-12, 2024, Proceedings, Part II. Lecture Notes in Computer Science, vol. 14980, pp. 41–50. Springer (2024). , https://doi.org/10.1007/978-3-031-71170-1_5

[9] Dave, B., Shimizu, C.: Towards understanding the impact of schema on knowledge graph embeddings (invited) (2023), in press

[10] Dettmers, T., Minervini, P., Stenetorp, P., Riedel, S.: Convolutional 2d knowledge graph embeddings (2018)

[11] Dörpinghaus, J., Weil, V., Düing, C., Sommer, M.W.: Centrality measures in multi-layer knowledge graphs. arXiv preprint arXiv:2203.09219 (2022)

[12] 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)

[13] Galton, A.: Reified temporal theories and how to unreify them. In: Proceedings of the 12th International Joint Conference on Artificial Intelligence - Volume 2. p. 1177–1182. IJCAI'91, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1991)

[14] Gangemi, A., Guarino, N., Masolo, C., Oltramari, A., Schneider, L.: Sweetening ontologies with dolce. In: Gómez-Pérez, A., Benjamins, V.R. (eds.) Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web. pp. 166–181. Springer Berlin Heidelberg, Berlin, Heidelberg (2002)

[15] Gangemi, A., Presutti, V.: Ontology design patterns. In: Staab, S., Studer, R. (eds.) Handbook on Ontologies, pp. 221–243. International Handbooks on Information Systems, Springer (2009). , https://doi.org/10.1007/978-3-540-92673-3_10

[16] Gangemi, A., Presutti, V.: Multi-layered n-ary patterns. In: Hitzler, P., Gangemi, A., Janowicz, K., Krisnadhi, A., Presutti, V. (eds.) Ontology Engineering with Ontology Design Patterns - Foundations and Applications, Studies on the Semantic Web, vol. 25, pp. 105–131. IOS Press (2016). , https://doi.org/10.3233/978-1-61499-676-7-105

[17] Gómez-Pérez, A.: Ontology Evaluation, pp. 251–273. Springer Berlin Heidelberg, Berlin, Heidelberg (2004). , https://doi.org/10.1007/978-3-540-24750-0_13

[18] Guarino, N., Welty, C.A.: An Overview of OntoClean, pp. 151–171. Springer Berlin Heidelberg, Berlin, Heidelberg (2004). , https://doi.org/10.1007/978-3-540-24750-0_8

[19] Gutierrez Basulto, V., Schockaert, S.: From knowledge graph embedding to ontology embedding? an analysis of the compatibility between vector space representations and rules (2018)

[20] Heist, N., Hertling, S., Paulheim, H.: Kgreat: A framework to evaluate knowledge graphs via downstream tasks. In: Frommholz, I., Hopfgartner, F., Lee, M., Oakes, M., Lalmas, M., Zhang, M., Santos, R.L.T. (eds.) Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, CIKM 2023, Birmingham, United Kingdom, October 21-25, 2023. pp. 3938–3942. ACM (2023). , https://doi.org/10.1145/3583780.3615241

[21] Hitzler, P.: Semantic Web: A review of the field. Communications of the ACM (2021), to appear

[22] Hitzler, P., Krisnadhi, A.: On the roles of logical axiomatizations for ontologies. In: Hitzler, P., Gangemi, A., Janowicz, K., Krisnadhi, A., Presutti, V. (eds.) Ontology Engineering with Ontology Design Patterns - Foundations and Applications, Studies on the Semantic Web, vol. 25, pp. 73–80. IOS Press (2016). , https://doi.org/10.3233/978-1-61499-676-7-73

[23] 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

[24] Hogan, A., Blomqvist, E., Cochez, M., d'Amato, C., de Melo, G., Gutierrez, C., Kirrane, S., Gayo, J.E.L., Navigli, R., Neumaier, S., Ngomo, A.N., Polleres, A., Rashid, S.M., Rula, A., Schmelzeisen, L., Sequeda, J.F., Staab, S., Zimmermann, A.: Knowledge graphs. ACM Comput. Surv. **54**(4), 71:1–71:37 (2022). , https://doi.org/10.1145/3447772

[25] 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. pp. 381–394. Siena, Italy (Jul 2023), https://hal-emse.ccsd.cnrs.fr/emse-04081543

[26] Jain, N., Kalo, J.C., Balke, W.T., Krestel, R.: Do embeddings actually capture knowledge graph semantics? In: The Semantic Web: 18th International Conference, ESWC 2021, Virtual Event, June 6–10, 2021, Proceedings 18. pp. 143–159. Springer (2021)

[27] Kang, B., Garcia Garcia, D., Lijffijt, J., Santos-Rodríguez, R., De Bie, T.: Conditional t-sne: more informative t-sne embeddings. Machine Learning **110**, 2905–2940 (2021)

[28] 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

[29] Knublauch, H., Kontokostas, D. (eds.): Shapes Constraint Language (SHACL). W3C Recommendation 20 July 2017 (2017), https://www.w3.org/TR/shacl/

[30] Krisnadhi, A., Hu, Y., Janowicz, K., Hitzler, P., Arko, R.A., Carbotte, S., Chandler, C., Cheatham, M., Fils, D., Finin, T.W., Ji, P., Jones, M.B., Karima, N., Lehnert, K.A., Mickle, A., Narock, T.W., O'Brien, M., Raymond, L., Shepherd, A., Schildhauer, M., Wiebe, P.: The geolink modular oceanography ontology. In: Arenas, M., Corcho, Ó., Simperl, E., Strohmaier, M., d'Aquin, M., Srinivas, K., Groth, P.T., Dumontier, M., Heflin, J., Thirunarayan, K., Staab, S. (eds.) The Semantic Web - ISWC 2015 - 14th International Semantic Web Conference, Bethlehem, PA, USA, October 11-15, 2015, Proceedings, Part II. Lecture Notes in Computer Science, vol. 9367, pp. 301–309. Springer (2015). , https://doi.org/10.1007/978-3-319-25010-6_19

[31] Krisnadhi, A.A.: Ontology Pattern-Based Data Integration. Ph.D. thesis, Wright State University (2015)

[32] Lin, Y., Liu, Z., Sun, M., Liu, Y., Zhu, X.: Learning entity and relation embeddings for knowledge graph completion. Proceedings of the AAAI Conference on Artificial Intelligence **29**(1) (Feb 2015). , https://ojs.aaai.org/index.php/AAAI/article/view/9491

[33] Lloyd, O., Liu, Y., Gaunt, T.R.: Assessing the effects of hyperparameters on knowledge graph embedding quality. J. Big Data **10**(1), 59 (2023). , https://doi.org/10.1186/s40537-023-00732-5

[34] Mansfield, M., Tamma, V., Goddard, P., Coenen, F.: Capturing expert knowledge for building enterprise sme knowledge graphs. In: Proceedings of the 11th Knowledge Capture Conference. p. 129–136. K-CAP '21, Association for Computing Machinery, New York, NY, USA (2021). , https://doi.org/10.1145/3460210.3493569

[35] Meyer, L.P., Stadler, C., Frey, J., Radtke, N., Junghanns, K., Meissner, R., Dziwis, G., Bulert, K., Martin, M.: Llm-assisted knowledge graph engineering: Experiments with chatgpt (2023)

[36] 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. p. 809–816. ICML'11, Omnipress, Madison, WI, USA (2011)

[37] 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

[38] 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. p. 1419–1428. WWW '16, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE (2016). , https://doi.org/10.1145/2872427.2874809

[39] Poveda-Villalón, M., Gómez-Pérez, A., Suárez-Figueroa, M.C.: Oops! (ontology pitfall scanner!): An on-line tool for ontology evaluation. Int. J. Semantic Web Inf. Syst. **10**(2), 7–34 (2014). , https://doi.org/10.4018/ijswis.2014040102

[40] Raad, J., Cruz, C.: A survey on ontology evaluation methods. In: Fred, A.L.N., Dietz, J.L.G., Aveiro, D., Liu, K., Filipe, J. (eds.) KEOD 2015 - Proceedings of the International Conference on Knowledge Engineering and Ontology Development, part of the 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2015), Volume 2, Lisbon, Portugal, November 12-14, 2015. pp. 179–186. SciTePress (2015). , https://doi.org/10.5220/0005591001790186

[41] Rodríguez-Doncel, V., Krisnadhi, A.A., Hitzler, P., Cheatham, M., Karima, N., Amini, R.: Pattern-based linked data publication: The linked chess dataset case. In: Hartig, O., Sequeda, J.F., Hogan, A. (eds.) Proceedings of the 6th International Workshop on Consuming Linked Data co-located with 14th International Semantic Web Conference (ISWC 2105), Bethlehem, Pennsylvania, USA, October 12th, 2015. CEUR Workshop Proceedings, vol. 1426. CEUR-WS.org (2015), http://ceur-ws.org/Vol-1426/paper-05.pdf

[42] 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) (jan 2021). , https://doi.org/10.1145/3424672

[43] Shimizu, C., Hammar, K., Hitzler, P.: Modular ontology modeling. Semantic Web **14**(3), 459–489 (2023), https://doi.org/10.3233/SW-222886

[44] Shimizu, C., Hirt, Q., Hitzler, P.: MODL: A modular ontology design library. In: Janowicz, K., Krisnadhi, A.A., Poveda-Villalón, M., Hammar, K., Shimizu, C. (eds.) Proceedings of the 10th Workshop on Ontology Design and Patterns (WOP 2019) co-located with 18th International Semantic Web Conference (ISWC 2019), Auckland, New Zealand, October 27, 2019. CEUR Workshop Proceedings, vol. 2459, pp. 47–58. CEUR-WS.org (2019), http://ceur-ws.org/Vol-2459/paper4.pdf

[45] Shimizu, C., Hitzler, P.: Accelerating knowledge graph and ontology engineering with large language models (2024), https://arxiv.org/abs/2411.09601

[46] Smith, B.: The basic tools of formal ontology. In: Formal Ontology in Information Systems (1998)

[47] 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

[48] Thomas Pellissier Tanon, Gerhard Weikum, F.M.S.: Yago 4: A reason-able knowledge base (2020).

[49] Trouillon, T., Welbl, J., Riedel, S., Éric Gaussier, Bouchard, G.: Complex embeddings for simple link prediction (2016)

[50] Tsaneva, S., Vasic, S., Sabou, M.: Llm-driven ontology evaluation: Verifying ontology restrictions with chatgpt. In: Tiwari, S., Mihindukulasooriya, N., Osborne, F., Kontokostas, D., D'Souza, J., Kejriwal, M., Pellegrino, M.A., Rula, A., Gayo, J.E.L., Cochez, M., Alam, M. (eds.) Joint proceedings of the 3rd International workshop on knowledge graph generation from text (TEXT2KG) and Data Quality meets Machine Learning and Knowledge Graphs (DQMLKG) co-located with the Extended Semantic Web Conference (ESWC 2024), Hersonissos, Greece, May 26-30, 2024. CEUR Workshop Proceedings, vol. 3747, p. 15. CEUR-WS.org (2024), https://ceur-ws.org/Vol-3747/dqmlkg_paper3.pdf

[51] Wang, Q., Mao, Z., Wang, B., Guo, L.: Knowledge graph embedding: A survey of approaches and applications. IEEE Transactions on Knowledge and Data Engineering **29**(12), 2724–2743 (2017).

[52] Yang, B., tau Yih, W., He, X., Gao, J., Deng, L.: Embedding entities and relations for learning and inference in knowledge bases (2015)

[53] Zheng, D., Song, X., Ma, C., Tan, Z., Ye, Z., Dong, J., Xiong, H., Zhang, Z., Karypis, G.: Dgl-ke: Training knowledge graph embeddings at scale. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. p. 739–748. SIGIR '20, Association for Computing Machinery, New York, NY, USA (2020)