Out-of-Vocabulary Entities in Benchmarks for Link Prediction

Caglar Demir
Data Science Research Group, Paderborn University

Axel-Cyrille Ngonga Ngomo
Data Science Research Group, Paderborn University

ABSTRACT
Knowledge graph embedding techniques are key to making knowledge graphs amenable to the plethora of machine learning approaches based on vector representations. Link prediction is often used as a proxy to evaluate the quality of these embeddings. Given that the creation of benchmarks for link prediction is a time-consuming endeavor, most work on the subject matter uses only a few benchmarks. As benchmarks are crucial for the fair comparison of algorithms, ensuring their quality is tantamount to providing a solid ground for developing better solutions to link prediction and related tasks embedding knowledge graphs. First studies of benchmarks pointed to limitations pertaining to information leaking from the development to the test fragments of some benchmark datasets. We spotted a further common limitation of three of the benchmarks commonly used for evaluating link prediction approaches: out-of-vocabulary entities in the test and validation sets. We provide an implementation of an approach for spotting and removing such entities and provide corrected versions of the datasets WN18RR, FB15K-237, and YAGO3-10. Our experiments on the corrected versions of WN18RR, FB15K-237, and YAGO3-10 suggest that the measured performance of state-of-the-art approaches is altered significantly with p-values < 1%, < 1.4%, and < 1%, respectively. Overall, state-of-the-art approaches gain on average absolute 3.29 ± 0.24% in all metrics on WN18RR. This means that some of the conclusions achieved in previous works might need to be revisited. We provide an open-source implementation of our experiments and corrected datasets at https://github.com/dice-group/OOV-In-Link-Prediction.

KEYWORDS
Link Prediction, Knowledge Graph Embedding

1 INTRODUCTION
Knowledge Graphs (KGs) represent structured collections of facts modelled in typed relationships between entities [13]. These collections of facts have been used in a wide range of applications, including web search [12], cancer research [26], and even entertainment [18]. However, most KGs on the Web are far from being complete [21]. For instance, birthplaces for 71% of people in Freebase and 66% of people in DBpedia are not found in the respective KGs. Additionally, more than 58% of scientists in DBpedia are not linked to the predicate that describes what they are known for [17]. Link prediction on KGs refers to identifying missing information [11]. Knowledge Graph Embedding (KGE) approaches have been particularly successful at tackling the link prediction task [21]. Hence, link prediction is often used as a proxy to measure the quality of KGE approaches.

To quantify link prediction performances of KGE approaches, the KGE research community often relies on the datasets WN18, WN18RR, FB15K, FB15K-237, and YAGO3-10. Tautanova and Chen [30] note the test leakage problem of FB15K and WN18, i.e., a large number of triples in the test sets can be derived by inverting triples in the train set [11]. Tautanova and Chen [30] construct FB15K-237 via removing near-duplicate and inverse-duplicate relations from FB15K. Dettmers et al. [11] investigate the severity of this problem and show that a simple rule-based model exploiting the test leakage problem of FB15K and WN18 achieves state-of-the-art link prediction performance on both datasets. Dettmers et al. [11] create WN18RR that cannot be exploited using a single rule and caution against using FB15K and WN18. Two recent studies point out that the same benchmark datasets may suffer from another problem: the validation and test sets of WN18RR, FB15K-237, and YAGO3-10 contain entities that do not occur during training [6, 10].

The goal of this paper is two-fold: first, we aim to provide new, corrected versions of all three datasets. We dub these corrected versions WN18RR*, FB15K-237* and YAGO3-10*. Second, we investigate the impact of out-of-vocabulary entities (entities that do not occur in the train set) on the link prediction performance of a selection of state-of-the-art KGE techniques. Our experiments indicate that discrepancies in link prediction performances of approaches are statistically significant with p-values < 1%, < 1.4%, and < 1% on WN18RR, FB15K-237, and YAGO3-10 (see Section 6). For instance, approaches achieve absolute gains of 3.29 ± 0.24% on average, in all metrics on WN18RR*, while the link prediction performance of TransE on FB15K-237* is increased by absolute 9% in all metrics. During our experiments, we observed that out-of-vocabulary entities often occur with particular relations (e.g., hypernym on WN18RR). Hence, previously reported link prediction per relation results do not fully reflect actual performances of approaches. In turn, we observe that the impact of out-of-vocabulary entities is not as severe in ranking missing relations as in ranking missing head and tail entities.

2 RELATED WORK
A wide range of works have investigated KGE to address various tasks such as type prediction, relation prediction, link prediction,
question answering, item recommendation and knowledge graph completion [8, 9, 14, 23]. We refer to [7, 15, 21, 24, 34] for recent surveys and briefly overview selected KGE techniques.

RESCAL [23] is a bilinear model that computes a three-way factorization of a third-order adjacency tensor representing the input KG. RESCAL captures various types of relations in the input KG but is limited in its scalability as it has quadratic complexity in the factorization rank [31]. DistMult [35] can be seen as an efficient extension of RESCAL with a diagonal matrix per relation to reduce the complexity of RESCAL [3]. DistMult performs poorly on antisymmetric relations while performing well on symmetric relations [31]. Note that through applying the reciprocal data augmentation technique, this incapability of DistMult is alleviated [25]. TuckER [3] performs a Tucker decomposition on the binary tensor representing the input KG which enables multi-task learning through parameter sharing between different relations via the core tensor. ComplEx [33] extends DistMult by learning representations in a complex vector space. ComplEx can infer both symmetric and antisymmetric relations via a Hermitian inner product of complex-valued embeddings to generate scores of relations captured by 2D convolution, ConvE [11] yields a state-of-the-art performance on complex-valued embeddings that involves the conjugate-transpose of one of the two input vectors. ComplEx yields state-of-the-art performance on the link prediction task while leveraging the linear space and time complexity of the dot products. Trouillon et al. [32] showed that ComplEx is equivalent to HolE [22]. ConvE [11] is a convolutional neural model that relies on a 2D convolution operation to capture the interactions between entities and relations. Through interactions captured by 2D convolution, ConvE yields a state-of-the-art performance in link prediction. ConEx [10] combines a 2D convolution followed by an affine transformation with a Hermitian inner product of complex-valued embeddings to generate scores of triples. Hence, ConEx can be considered a multiplicative inclusion of ConvE into ComplEx.

3 PRELIMINARIES

Knowledge Graphs. Let $\mathcal{E}$ and $\mathcal{R}$ represent the set of entities and relations, respectively. Then, a KG $\mathcal{G} = \{(h, r, t) \in \mathcal{E} \times \mathcal{R} \times \mathcal{E}\}$ can be formalised as a set of triples where each triple contains two entities $h, t \in \mathcal{E}$ and a relation $r \in \mathcal{R}$.

Link Prediction. The link prediction task addresses the problem of predicting whether unseen triples (i.e., triples not found in $\mathcal{G}$) are true [15]. The task is often formalised by learning a parameterised scoring function $\phi_\Theta : \mathcal{E} \times \mathcal{R} \times \mathcal{E} \mapsto \mathbb{R}$ [15, 21] ideally characterised by $\phi_\Theta(h, r, t) = \phi_\Theta(x, y, z)$ if $(h, r, t)$ is true and $(x, y, z)$ is not. An approach’s link prediction performance is evaluated using its ranking of missing entities in unseen triples as a proxy [11, 21, 23, 33]. Consequently, benchmark datasets for link prediction consist of disjoint sets of triples denoted by $\mathcal{G}^{\text{Train}}, \mathcal{G}^{\text{Val}},$ and $\mathcal{G}^{\text{Test}}$. Vocabulary of entities for the training phase can be defined as $\mathcal{E}^{\text{Train}} = \{x | (x, r, t) \in \mathcal{G}^{\text{Train}}\}$. Similarly, the vocabulary of relations can be defined as $\mathcal{R}^{\text{Train}} = \{r | (h, r, t) \in \mathcal{G}^{\text{Train}}\}$. Analogously, vocabulary of entities and relations for validation and test phases can be obtained.

Evaluation metrics. Link prediction performances of approaches are often measured via the filtered mean reciprocal rank (MRR) and Hits@N metrics based on missing head and tail entity rankings [11, 21, 33]. Formally, the filtered MRR is defined as

$$\text{MRR} = \frac{1}{|\mathcal{G}^{\text{Test}}|} \sum_{(h, r, t) \in \mathcal{G}^{\text{Test}}} \left( \frac{1}{\text{rank}(t|h, r)} + \frac{1}{\text{rank}(h|r, t)} \right),$$

(1)

where $\text{rank}(t|h, r)$ (equivalently $\text{rank}(h|r, t)$) is computed in three consecutive steps. (a) Scores are computed $\forall x \in \mathcal{E} : \phi_\Theta(h, r, x)$. (b) Scores of triples that occurred in train, validation, and test sets are filtered except for the input test triple $(h, r, t)$. (c) Entities are ranked in descending order of corresponding scores. Given $(h, r, t) \in \mathcal{G}^{\text{Test}}$, rank$(t|h, r)$ denotes the rank of missing $t$ in the ordered entities. To further investigate link prediction performances of approaches, link prediction per relation performances are often measured as

$$\text{MRR}_{r_t} = \frac{1}{|\mathcal{G}^{\text{Test}}|} \sum_{(h, r, t) \in \mathcal{G}^{\text{Test}}} \left( \frac{1}{\text{rank}(t|h, r_t)} + \frac{1}{\text{rank}(h|r_t, t)} \right),$$

(2)

Equation (2) categorises Mean Reciprocal Rank (MRR) scores per relation [1, 27, 36]. Similarly, the filtered Hits@N is defined as

$$\text{MRR} = \frac{1}{|\mathcal{G}^{\text{Test}}|} \sum_{(h, r, t) \in \mathcal{G}^{\text{Test}}} \left( \frac{1}{\text{rank}(t|h, r)} \right),$$

(3)

where the binary function $I$ returns 1 if the condition is true, otherwise 0. Link prediction performances can also be measured in terms of missing relation prediction [34]. To this end, Equation (1) is altered as

$$\text{MRR} = \frac{1}{|\mathcal{G}^{\text{Test}}|} \sum_{(h, r, t) \in \mathcal{G}^{\text{Test}}} \left( \frac{1}{\text{rank}(r|h, t)} \right).$$

(4)

Analogous to Equation (4), Equation (3) is modified to quantify relation prediction performances [8].

4 PROBLEM

Learning a scoring function. To tackle the link prediction problem, the common practice is to learn a parameterized scoring function $\phi_\Theta : \mathcal{E} \times \mathcal{R} \times \mathcal{E} \mapsto \mathbb{R}$, where $\Theta$ corresponds to parameters of the scoring function. $\Theta$ involves embeddings of entities and relations and may involve additional parameters depending on the architecture of the scoring function, e.g., kernels in convolution operation, affine transformations, linear transformations or even tensors [2, 3, 8, 10, 11, 20, 23]. The standard workflow of learning $\phi_\Theta$ to tackle the link prediction problem consists of three parts:

(1) learning $\Theta$ by minimizing a set loss function (e.g., a margin or entropy-based loss functions) on $\mathcal{G}^{\text{Train}}$

(2) determining hyperparameters of $\phi_\Theta$ on $\mathcal{G}^{\text{Val}}$, and

(3) measuring the link prediction performance of $\phi_\Theta$ on $\mathcal{G}^{\text{Test}}$.

This workflow entails that $\mathcal{E}^{\text{Train}}$ and $\mathcal{R}^{\text{Train}}$ are known a priori since $\Theta$ must involve embeddings of all entities and relations seen during the training phase. Hence, $\Theta$ can be learned as a byproduct of minimizing the set loss function through an optimizer (e.g. ADAM [16], RMSprop [28]). In this workflow, the following two conditions must hold:

(1) $\mathcal{E}^{\text{Val}} \subseteq \mathcal{E}^{\text{Train}} \wedge \mathcal{E}^{\text{Test}} \subseteq \mathcal{E}^{\text{Train}}$ and

(2) $\mathcal{R}^{\text{Val}} \subseteq \mathcal{R}^{\text{Train}} \wedge \mathcal{R}^{\text{Test}} \subseteq \mathcal{R}^{\text{Train}}$.

An OOV-entity is an entity $e \in (\mathcal{E}^{\text{Val}} \cup \mathcal{E}^{\text{Train}} \cup \mathcal{E}^{\text{Test}}) \setminus \mathcal{E}^{\text{Train}}$. OOV relations are defined analogously. If these two conditions
Table 1: Overview of WN18RR, FB15K-237 and YAGO3-10 benchmark datasets. \(|G|, |E|, \text{ and } |R|\) denote number of triples, entities, and relation. Indegr. and Outdegr. (M±SD) \(G\) stand for mean and standard deviation of node indegrees and node outdegrees, respectively.

| Dataset       | \(G\)\(^{\text{Train}}\) | \(E\)\(^{\text{Train}}\) | \(R\)\(^{\text{Train}}\) | Indegr. (M±SD) \(G\)\(^{\text{Train}}\) | Outdegr. (M±SD) \(G\)\(^{\text{Train}}\) |
|---------------|----------------|----------------|----------------|--------------------------------|--------------------------------|
| WN18RR        | 86,835         | 272,115        | 1,079,040      | 2.72±7.74                      | 20.34±98.54                    |
| FB15K-237     | 40,559         | 14,505         | 123,143        | 11                             | 237                            |
| YAGO3-10      | 11             | 233            | 37             | 5.25                           | 1.59±5.25                      |

Table 2: Overview of out-of-vocabulary entities in WN18RR, FB15K-237 and YAGO3-10 benchmark datasets. OOV in \(G\) denotes number of triples containing at least an out-of-vocabulary entity and its percentage in the respective set.

| Dataset       | \(|E|\)\(^{\text{Train}}\) - \(|E|\)\(^{\text{Test}}\) | OOV in \(G\)\(^{\text{Test}}\) | \(|E|\)\(^{\text{Val}}\) - \(|E|\)\(^{\text{Train}}\) | OOV in \(G\)\(^{\text{Val}}\) |
|---------------|--------------------------------|----------------|--------------------------------|----------------|
| WN18RR        | 209 (6.70%)                   | 28 (0.14%)     | 198 (6.92%)                   | 9 (0.05%)      |
| FB15K-237     | 28 (0.14%)                   | 18 (0.36%)     | 22                             | 22 (0.44%)     |
| YAGO3-10      | 18 (0.36%)                   | 22 (0.44%)     |                                |                |

In our experiments, we are interested in quantifying the severity of these situations.

In our experiments, we are interested in quantifying the severity of these situations.

5 EXPERIMENTAL SETUP

Baselines. In our experiments, we relied on pretrained RESCAL, TransE, DistMult, ComplEx, ConvE, ConEx, and TuckER approaches provided in [6, 10, 25]. This decision stems from the fact that Ruffinelli et al. [25] conducted an extensive analysis on the impact of hyperparameter optimization and training strategies in link prediction performances. Their findings indicate that the relative performance differences between various approaches often shrink and sometimes even reverse when compared to prior results, provided that approaches are optimized properly. Hence, we alleviated the impact of different training strategies in our experiments by relying on the results of [25].

Datasets. Bordes et al. [5] construct WN18 and FB15K datasets to quantify the link prediction performance of KGE approaches. WN18 is a subset of WordNet [19] that contains lexical relations, while FB15K is a subset of Freebase [4]. Toutanova and Chen [29] highlight the test leakage problem of benchmark datasets and construct the FB15K-237 dataset by removing triples containing near-duplicate and inverse-duplicate relations from the FB15K dataset. The FB15K-237 dataset is constructed by limiting the set of relations in FB15K to the most frequent 401 relations. Next, near-duplicate and inverse relations are detected and removed from the train set of FB15K. This process detects many inverse relationships in relations, e.g., award_nominee/award

Evaluation metrics. We employ the standard metrics filtered MRR and hits at \(N\) (H@\(N\)) for link prediction and relation prediction [8, 11, 33].

6 EVALUATION

Tables 3 to 5 report link prediction results on WN18RR, FB15K-237, YAGO3-10 and their respective vectorized versions. Our results on WN18RR\(^*\) indicate that state-of-the-art approaches achieve on average absolute 3.29 ± 0.24% gains in all metrics. Such a distinct difference is not observed on FB15K-237\(^*\) and YAGO3-10\(^*\).

These results also indicate that shallow models (e.g., DistMult and ComplEx) perform particularly well on FB15K-237, YAGO3-10,
### Table 3: Link prediction results on WN18RR and WN18RR*.

|       | WN18RR | WN18RR* |
|-------|--------|---------|
|       | Hits@1 | Hits@3 | Hits@10 | MRR | Hits@1 | Hits@3 | Hits@10 |
| RESCAL | .467   | .439   | .480    | .517 | .499   | .470   | .513    | .552 |
| TransE | .175   | .044   | .227    | .484 | .187   | .047   | .243    | .517 |
| DistMult | .452  | .413   | .466    | .530 | .483   | .442   | .499    | .566 |
| ComplEx | .475  | .438   | .490    | .547 | .509   | .470   | .525    | .587 |
| ConvE  | .442   | .411   | .451    | .504 | .472   | .440   | .482    | .538 |
| ConEx  | .481   | .448   | .493    | .550 | .512   | .477   | .525    | .584 |
| TuckER | .466   | .441   | .476    | .515 | .488   | .471   | .509    | .550 |

### Table 4: Link prediction results on FB15K-237 and FB15K-237*.

|       | FB15K-237 | FB15K-237* |
|-------|-----------|------------|
|       | Hits@1    | Hits@3    | Hits@10 | MRR  | Hits@1 | Hits@3 | Hits@10 |
| RESCAL | .351      | .260      | .387    | .531 | .354   | .262   | .391    | .537 |
| TransE | .150      | .090      | .148    | .284 | .246   | .175   | .263    | .390 |
| DistMult | .343    | .249      | .378    | .531 | .343   | .249   | .379    | .532 |
| ComplEx | .348   | .253      | .384    | .536 | .348   | .253   | .384    | .536 |
| ConvE  | .329      | .244      | .359    | .501 | .334   | .246   | .364    | .501 |
| ConEx  | .366      | .271      | .403    | .555 | .366   | .271   | .403    | .555 |
| TuckER | .363      | .268      | .400    | .553 | .363   | .268   | .400    | .553 |

### Table 5: Link prediction results on YAGO3-10 and YAGO3-10*.

|       | YAGO3-10 | YAGO3-10* |
|-------|----------|-----------|
|       | MRR       | Hits@1    | Hits@3    | Hits@10 | MRR       | Hits@1    | Hits@3    | Hits@10 |
| DistMult | .543   | .466      | .590      | .683    | .545     | .467      | .592      | .686    |
| ComplEx | .547    | .468      | .594      | .690    | .549     | .470      | .596      | .692    |
| ConEx   | .553    | .474      | .601      | .696    | .555     | .476      | .603      | .698    |
| TuckER  | .427    | .331      | .476      | .609    | .429     | .332      | .477      | .611    |

### Table 6: Relation prediction results on WN18RR and WN18RR*.

|       | WN18RR | WN18RR* |
|-------|--------|---------|
|       | MRR    | Hits@1 | Hits@3 | Hits@10 | MRR    | Hits@1 | Hits@3 | Hits@10 |
| RESCAL | .578   | .335   | .784   | .892    | .587   | .343   | .797   | .891    |
| TransE | .547   | .360   | .682   | .865    | .525   | .328   | .669   | .859    |
| DistMult | .671  | .539   | .759   | .869    | .687   | .553   | .780   | .889    |
| ComplEx | .785   | .705   | .822   | .989    | .820   | .749   | .857   | .994    |
| ConvE   | .353   | .145   | .405   | .857    | .358   | .148   | .414   | .853    |

### Table 7: Relation prediction results on FB15K-237 and FB15K-237*.

|       | FB15K-237 | FB15K-237* |
|-------|-----------|------------|
|       | MRR       | Hits@1    | Hits@3    | Hits@10 | MRR       | Hits@1    | Hits@3    | Hits@10 |
| RESCAL | .192     | .021      | .140      | .823    | .192     | .021      | .141      | .824    |
| TransE | .672     | .569      | .733      | .800    | .673     | .589      | .734      | .801    |
| DistMult | .568   | .425      | .660      | .823    | .569     | .425      | .661      | .824    |
| ComplEx | .632    | .506      | .717      | .855    | .633     | .507      | .718      | .856    |
| ConvE   | .466    | .352      | .732      | .874    | .467     | .352      | .732      | .874    |
and their corrected versions. These results hence do not corroborate the claim that shallow models do not perform well on knowledge graphs having high node in-degree [11]. In turn, we conjecture that this claim might have to be reconsidered. Although on average in-degree of nodes in FB15K-237 is circa 7.5 times larger than WN18RR, ComplEx and DistMult outperform ConvE in all metrics. In FB15K-237*, the link prediction performances of RESCAL, TransE, and ConvE are increased in all metrics, whereas link prediction performances of DistMult and ComplEx did not change.

**Link Prediction per Relation.** During our evaluation, we were interested in quantifying the impact of OVV-entities in the link prediction per relation task. Tables 8 to 10 report link prediction per relation results on all datasets. Results indicate that performances of approaches improve particularly well on hypernym, instance_hypernym, member_meronym, synset_domain_topic_of, has_part and member_of_domain_usage on WN18RR*, whereas such distinct improvements are not observed on the remaining relations.

**Statistical Hypothesis Testing.** We carried out a Wilcoxon signed-rank test to check whether the impact of OOV-entities in link prediction performances is significant. Our null hypothesis was that the link prediction performances of state-of-the-art approaches on the datasets with and without OOV entities and relations come from the same distribution. The alternative hypothesis was that these results come from different distributions, i.e., removing triples containing OOV entities from the test set has a significant impact on link prediction performances. To perform the Wilcoxon signed-rank test (two-sided), we used the MRR, Hits@1, Hits@3, and Hits@10 performances on a benchmark dataset and its corrected version (e.g. WN18RR-WN18RR*, FB15K-237-FB15K-237*, and YAGO3-10-YAGO3-10*). We were able to reject the null hypothesis with p-values < 1%, < 1.4%, and < 1%, on all three tests.  

7 CONCLUSION

In this work, we investigated the impact of out-of-vocabulary entities in the link prediction and relation prediction problem. Our analysis shows that 6.70% and 6.92% triples in the test and validation splits of the WN18RR benchmark dataset do not serve for benchmarking link prediction performances of approaches. Our experiments also showed that state-of-the-art approaches gain on average absolute 3.29±0.24% in all metrics on WN18RR. Findings of our statistically hypothesis test indicates that link prediction performances of state-of-the-art approaches are significantly altered. In turn, the impact of out-of-vocabulary entities in ranking missing relations is not as distinct as in ranking missing entities. We provide an open-source implementation of our experiments along with corrected datasets in the project page.

REFERENCES

[1] Ivana Balazevic, Carl Allen, and Timothy Hospedales. 2019. Multi-relational Poincaré graph embeddings. In Advances in Neural Information Processing Systems. 4465–4475.

[2] Ivana Balazevic, Carl Allen, and Timothy M Hospedales. 2019. Hypernetwork knowledge graph embeddings. In International Conference on Artificial Neural Networks. Springer, 553–565.

[3] Ivana Balazevic, Carl Allen, and Timothy M Hospedales. 2019. Tucker: Tensor factorization for knowledge graph completion. arXiv preprint arXiv:1901.09590 (2019).

[4] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In Proceedings of the 2008 ACM SIGMOD international conference on Management of data. 1247–1250.

[5] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and RainerGemulla. 2013. TransE: a simple algorithm for learning embeddings of relations and entities. In Advances in neural information processing systems. 2787–2795.

[6] Samuel Bowers, Daniel Ruffinelli, Adrian Kochsiek, Patrick Betz, and RainerGemulla. 2020. LibKGE - A Knowledge Graph Embedding Library for Reproducible Research. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. 165–174. https://www.aclweb.org/anthology/2020.emnlp-demos.22

[7] Hongyun Cai, Vincent W Zheng, and Kevin Chen-Chuan Chang. 2018. A robust neural model for relation prediction. arXiv preprint arXiv:2001.09980 (2021).

[8] Caspar Derim, Diego Moussallem, and Axel-Cyrille N'gonga N'gomo. 2021. A shallow neural model for relation prediction. arXiv preprint arXiv:2001.09980 (2021).

[9] Caspar Derim and Axel-Cyrille N'gonga N'gomo. 2019. A physical embedding model for knowledge graphs. In Joint International Semantic Technology Conference. Springer, 192–209.

[10] Caspar Derim and Axel-Cyrille N'gonga N'gomo. 2021. Convolutional Complex Knowledge Graph Embeddings. In Eighteenth Extended Semantic Web Conference - Research Track. https://openreview.net/forum?id=T4S-4TFqX

[11] Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2d knowledge graph embeddings. In Thirty-Second AAAI Conference on Artificial Intelligence.

[12] Jeffrey Scott Eder. 2012. Knowledge graph based search system. US Patent App. 13/404,109.

[13] Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard de Melo, Claudio Gutierrez, José Emilio Lahra Gayo, Sabina Kerrane, Sebastian Neumaier, Axel Pollieres, et al. 2020. Knowledge graphs. arXiv preprint arXiv:2003.02520 (2020).

[14] Xiao Huang, Jingyuan Zhang, Dingcheng Li, and Ping Li. 2019. Knowledge graph embedding based question answering. In Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining. 105–113.

[15] Shaoshiong Ji, Shriui Pan, Erik Cambria, Pekka Marttinen, and Philip S Yu. 2020. A Survey on Knowledge Graphs: Representation, Acquisition and Applications. arXiv preprint arXiv:2002.00388 (2020).

[16] Diedrik F Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014).

[17] Denis Krompaß, Stephan Baier, and Volker Tresp. 2015. Type-constrained representation learning in knowledge graphs. In International semantic web conference. Springer, 640–655.

[18] Stanislav Malychev, Markus Krötzsch, Larry González, Julius Gonsior, and Adrian Bielefeldt. 2018. Getting the most out of Wikidata: semantic technology usage in Wikipedia’s knowledge graph. In International Semantic Web Conference. Springer, 376–392.

[19] George A Miller. 1995. WordNet: a lexical database for English. Commun. ACM 38, 11 (1995), 39–41.

[20] Dai Quoc Nguyen, Tuan Dinh Nguyen, Dat Quoc Nguyen, and Dinh Phung. 2018. A Novel Embedding Model for Knowledge Base Completion Based on Convolutional Neural Network. In NAACL. 327–333.

[21] Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. 2015. A review of relational machine learning for knowledge graphs. Proc. IEEE 104, 1 (2015), 11–33.

[22] Maximilian Nickel, Lorenzo Rosasco, and Tomaso Poggio. 2015. Holographic embeddings of knowledge graphs. arXiv preprint arXiv:1510.04935 (2015).

[23] Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2011. A Three-Way Model for Collective Learning on Multi-Relational Data. In KSEM. Vol. 11. 809–816.

[24] Chuan Qin, Hengshu Zhu, Fuheen Zhuang, Qingyu Guo, Qi Zhang, Le Zhang, Chao Wang, Enhong Chen, and Hui Xiong. [n.d.]. A survey on knowledge graph based recommender systems. SCIENTIA SINICA Informationis 38, 11 (1995), 39–41.

[25] Chuan Qin, Hengshu Zhu, Fuheen Zhuang, Qingyu Guo, Qi Zhang, Le Zhang, Chao Wang, Enhong Chen, and Hui Xiong. [n.d.]. A survey on knowledge graph based recommender systems. SCIENTIA SINICA Informationis 38, 11 (1995), 39–41.

[26] Muhammad Saleem, Maulik R Kamdar, Aftab Iqbal, Shanmuga Sampath, Helena F Deus, and Axel-Cyrille Ngonga Ngomo. 2014. Big linked cancer data: a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning.
### Table 8: MRR link prediction per relation results on WN18RR and WN18RR*.

| Relation                      | RESCAL  | TransE  | DistMult | ComplEx | ConvE  | ConEx  | TuckER |
|-------------------------------|---------|---------|----------|---------|--------|--------|--------|
| **WN18RR**                    |         |         |          |         |        |        |        |
| hypernym                      | .149    | .071    | .117     | .157    | .090   | .150   | .121   |
| instance_hypernym             | .349    | .246    | .377     | .385    | .352   | .392   | .375   |
| member_meronym                | .225    | .168    | .148     | .232    | .198   | .171   | .181   |
| synset_domain_topic_of        | .325    | .301    | .348     | .335    | .311   | .373   | .344   |
| has_part                      | .164    | .095    | .155     | .212    | .146   | .192   | .171   |
| member_of_domain_usage        | .295    | .328    | .308     | .215    | .299   | .319   | .212   |
| member_of_domain_region       | .193    | .228    | .365     | .120    | .367   | .354   | .284   |
| derivationally_related_form   | .956    | .276    | .957     | .957    | .986   | .985   |        |
| also_see                      | .595    | .209    | .613     | .616    | .656   | .647   | .658   |
| verb_group                    | .974    | .337    | .975     | .974    | .974   | 1.00   | .987   |
| similar_to                    | .585    | .285    | 1.00     | 1.00    | 1.00   | 1.00   | 1.00   |
| **WN18RR***                   |         |         |          |         |        |        |        |
| hypernym                      | .173    | .083    | .135     | .184    | .105   | .169   | .140   |
| instance_hypernym             | .375    | .259    | .407     | .419    | .376   | .423   | .408   |
| member_meronym                | .231    | .173    | .152     | .238    | .203   | .176   | .186   |
| synset_domain_topic_of        | .333    | .312    | .355     | .347    | .318   | .376   | .350   |
| has_part                      | .167    | .097    | .158     | .216    | .148   | .191   | .169   |
| member_of_domain_usage        | .303    | .329    | .308     | .234    | .307   | .319   | .232   |
| member_of_domain_region       | .193    | .229    | .365     | .120    | .367   | .354   | .284   |
| derivationally_related_form   | .956    | .276    | .957     | .957    | .986   | .985   |        |
| also_see                      | .594    | .209    | .613     | .616    | .656   | .647   | .658   |
| verb_group                    | .974    | .337    | .975     | .974    | .974   | 1.00   | .987   |
| similar_to                    | .585    | .285    | 1.00     | 1.00    | 1.00   | 1.00   | 1.00   |

### Table 9: MRR link prediction per relation results based on some person related relations.

| Relation                      | RESCAL  | TransE  | DistMult | ComplEx | ConvE  | ConEx  | TuckER |
|-------------------------------|---------|---------|----------|---------|--------|--------|--------|
| **FB15K-237**                 |         |         |          |         |        |        |        |
| people/person/profession       | .371    | .134    | .388     | .402    | .386   | .374   | .375   |
| people/person/gender           | .585    | .452    | .528     | .518    | .505   | .593   | .605   |
| people/person/nationality      | .461    | .370    | .444     | .439    | .442   | .468   | .469   |
| people/person/languages        | .386    | .322    | .430     | .429    | .420   | .441   | .431   |
| people/person/religion         | .306    | .264    | .301     | .302    | .304   | .309   | .325   |
| **FB15K-237**                 |         |         |          |         |        |        |        |
| people/person/profession       | .377    | .340    | .388     | .402    | .386   | .374   | .375   |
| people/person/gender           | .586    | .452    | .528     | .518    | .505   | .593   | .605   |
| people/person/nationality      | .466    | .399    | .444     | .439    | .442   | .468   | .469   |
| people/person/languages        | .422    | .401    | .430     | .429    | .421   | .441   | .431   |
| people/person/religion         | .307    | .283    | .301     | .302    | .304   | .309   | .325   |

---

[29] Kristina Toutanova and Danqi Chen. 2015. Observed versus latent features for knowledge base and text inference. In Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality. 57–66.

[30] Kristina Toutanova, Danqi Chen, Patrick Pantel, Hoifung Poon, Pallavi Choudhury, and Michael Gamon. 2015. Representing text for joint embedding of text and knowledge bases. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. 1499–1509.

[31] Théo Trouillon, Christopher R Dance, Éric Gaussier, Johannes Welbl, Sebastian Riedel, and Guillaume Bouchard. 2017. Knowledge graph completion via complex tensor factorization. *The Journal of Machine Learning Research* 18, 1 (2017), 4735–4772.

[32] Théo Trouillon and Maximilian Nickel. 2017. Complex and holographic embeddings of knowledge graphs: a comparison. *arXiv preprint arXiv:1707.01475* (2017).

[33] Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex embeddings for simple link prediction. In *International Conference on Machine Learning*. 2071–2080.

[34] Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. 2017. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering* 29, 12 (2017), 2724–2743.

[35] Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015. Embedding entities and relations for learning and inference in knowledge bases. In *IJCAI*.

[36] Shuai Zhang, Yi Tay, Lina Yao, and Qijun Liu. 2019. Quaternion knowledge graph embeddings. In *Advances in Neural Information Processing Systems*. 2731–2741.
Table 10: MRR link prediction per relation results based on some person and place related relations.

| Relation          | DistMult | ComplEx | ConEx | TuckER |
|-------------------|----------|---------|-------|--------|
| **YAGO3-10**      |          |         |       |        |
| isLocatedIn       | .293     | .289    | .354  | .368   |
| happenedIn        | .505     | .499    | .473  | .434   |
| directed          | .509     | .504    | .527  | .480   |
| hasWonPrize       | .245     | .237    | .281  | .269   |
| isMarriedTo       | .985     | .985    | .985  | .936   |
| hasCapital        | .483     | .481    | .564  | .442   |
| hasNeighbor       | 1.00     | 1.00    | 1.00  | .406   |
| **YAGO3-10**      |          |         |       |        |
| isLocatedIn       | .302     | .297    | .365  | .379   |
| happenedIn        | .505     | .499    | .473  | .434   |
| directed          | .509     | .504    | .528  | .480   |
| hasWonPrize       | .247     | .237    | .282  | .271   |
| isMarriedTo       | .985     | .985    | .985  | .936   |
| hasCapital        | .571     | .568    | .667  | .523   |
| hasNeighbor       | 1.00     | 1.00    | 1.00  | .406   |