Relation Specific Transformations for Open World Knowledge Graph Completion

Haseeb Shah
Department of Computer Science
University of Alberta
hshah1@ualberta.ca

Johannes Villmow, Adrian Ulges
DCSM Department
RheinMain University of Applied Sciences
{johannes.villmow, adrian.ulges}@hs-rm.de

Abstract

We propose an open-world knowledge graph completion model that can be combined with common closed-world approaches (such as ComplEx) and enhance them to exploit text-based representations for entities unseen in training. Our model learns relation-specific transformation functions from text-based embedding space to graph-based embedding space, where the closed-world link prediction model can be applied. We demonstrate state-of-the-art results on common open-world benchmarks and show that our approach benefits from relation-specific transformation functions (RST), giving substantial improvements over a relation-agnostic approach.

1 Introduction

Knowledge graphs are an interesting source of information that can be exploited by retrieval (Dong et al., 2014) and question answering systems (Ferrucci et al., 2010). They are, however, known to be inherently sparse (Paulheim, 2017). To overcome this problem, *knowledge graph completion (KGC)* enriches graphs with new triples. While most existing approaches require all entities to be part of the training graph, for many applications it is of interest to infer knowledge about entities *not* present in the graph i.e. *open-world* entities. Here, approaches usually assume some text describing the target entity to be given, from which an entity representations can be inferred, for example via text embeddings (Mikolov et al., 2013; Devlin et al., 2018). To the best of our knowledge, only a few such *open-world* KGC approaches have been proposed so far (Xie et al., 2016; Shi and Weninger, 2017a; Shah et al., 2019).

We suggest a simple yet effective approach towards open-world KGC[1]: Similar to Shah et al. (2019)'s OWE model, our approach enables *existing* KGC models to perform open-world prediction: Given an open-world entity, its name and description are aggregated into a text-based entity representation, and a transformation from text-based embedding space to graph-based embedding space is learned, where the closed-world KGC model can be applied. However, while OWE's transformation only takes the open-world *entity* into account, our approach also utilizes the target triple's *relation*, such that specific mappings are learned for different relations such as *birthplace*, *spouse*, or *located_in* (see Figure 1). We demonstrate that this extension comes with strong improvements, yielding state-of-the-art results on common open-world datasets.

2 Related Work

Interest in KGC has increased recently, with most of the work focusing on embedding-based approaches. Earlier approaches (Nickel et al., 2016) have recently been complemented by other models such as DistMult (Yang et al., 2014), TransR (Lin et al., 2015), ComplEx (Trouillon et al., 2016), ProjE (Shi and Weninger, 2017b) and RotatE (Sun et al., 2019). The above models estimate the probability of triples $(head, rel, tail)$ using a scoring function $\phi(u_{head}, u_{rel}, u_{tail})$, where u_x denotes the embedding of entity/relation x and is a real-valued or complex-valued vector. ϕ depends on the model and varies from simple translation (Bordes et al., 2013) over bilinear forms (Yang et al., 2014) to complex-valued

[1] We make our code available under https://github.com/haseebs/RST-OWE

Proceedings of the Graph-based Methods for Natural Language Processing (TextGraphs), pages 79–84
Barcelona, Spain (Online), December 13, 2020

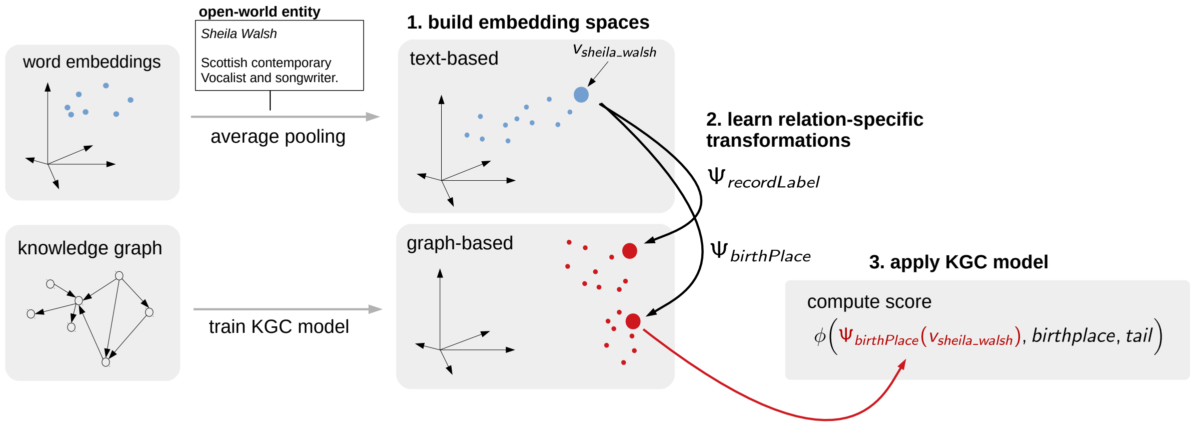

Figure 1: Our approach first trains a KGC model on the graph without using textual information (bottom left). For every annotated entity, we extract a text-based embedding v by aggregating the word embeddings for tokens in the entity's name and description (top left). A transformation Ψ is trained to map v to the space of graph-based embeddings (center). The learned mapping can then be applied to unknown entities, thus allowing the trained KGC model to be applied (right).

forms (Trouillon et al., 2016). Training happens by learning to discriminate real triples from perturbed ones, typically by negative sampling (Nickel et al., 2016).

While the knowledge graph completion models described above leverage only the structure of the graph, some approaches combine text with graph information, typically using embeddings that represent terms, sentences or documents (Goldberg, 2016). Embeddings are usually derived from language models, either in static form (Mikolov et al., 2013) or by contextualized models (Devlin et al., 2018). KGC models can employ such textual information for entities scarcely linked in the graph (Gesese et al., 2019). Most approaches combine a textual embedding with structural KGC approaches, either by initializing structural embeddings from text (Socher et al., 2013; Wang and Li, 2016), interpolating between textual and structural embeddings (Xu et al., 2017), sometimes with a joined loss (Toutanova and Chen, 2015) and gating mechanisms (Kristiadi et al., 2019). Others perform a fine-tuning for KGC based on textual labels of the entities and relations (Yao et al., 2019).

Only few other works have addressed *open-world KGC* so far. Xie et al. (2016) proposed DKRL with a joint training of graph-based embeddings (TransE) and text-based embeddings while regularizing both types of embeddings to be aligned using an additional loss. ConMask (Shi and Weninger, 2017a) is a text-centric approach where text-based embeddings for entities and relations are derived by an attention model over names and descriptions. Closest to our work is OWE (Shah et al., 2019), which trains graph and text embeddings independently and then learns a mapping between the two embedding spaces (more details are provided in Section 3). While OWE's mapping only take entities into account, our extended model's mapping is learned given both the entity *and relation* when predicting a triple. Orthogonal to our work, WOWE (Zhou et al., 2020) extends the OWE approach by replacing the averaging aggregator with a weighted attention mechanism and can be combined with our approach.

3 Approach

Given a knowledge graph $\mathcal{G} \subset E \times R \times E$ containing triples (h, r, t) (E and R denote finite sets of entities and relations), KGC models can perform tail prediction as follows: Given a pair of head and relation (h, r), the tail is estimated as

$$t^* = \arg\max_{t \in E} \ \phi(u_h, u_r, u_t) \tag{1}$$

where u_h, u_r, u_t are entity/relation embeddings and ϕ is a model-specific scoring function. Note that this approach – and our extension – can be applied for head prediction accordingly.

We address an open-world setting, where the triple's head is not a part of the knowledge graph, i.e.

Test Entity v_h	Relation r	Nearest Neighbors to $\Psi_r(v_h)$	Test Entity v_h	Relation r	Nearest Neighbors to $\Psi_r(v_h)$
Sheila Walsh (Scottish contemporary vocalist and songwriter)	birthPlace	1. Nigel McGuiness (British) 2. Darren Burridge (British) 3. Kim Newman (British)	Ferlyn Wong (Singaporean singer, dancer and actress)	occupation	1. Park Ji-yeon (singer) 2. Kim Sae-ron (actress) 3. Park Shin-hye (actress)
	recordLabel	1. Picture this live (album) 2. Thick as a Brick (album) 3. Whiplash Smile (album)		genre	1. Tony An (singer) 2. Exodus (album) 3. S.E.S. (band)

Table 1: Relation-specific transformations map entities to corresponding regions in embedding space: When mapping *Sheila_Walsh* with the *birthplace* relation, the resulting embedding lies in a cluster of British people. When mapping with the *recordLabel* relation, the resulting embedding lies close to pop albums (which also have a recordLabel relation).

$h \notin E$. However, h is assumed to come with a textual description. Our approach (Figure 1) transforms this text into a token sequence $\mathcal{W}_h = (w_1, w_2, ..., w_n)$, from which a sequence of embeddings $(v_{w_1}, v_{w_2}, ..., v_{w_n})$ is derived using a textual embedding model pre-trained on a large text corpus. We experimented with BERT (Devlin et al., 2018) but did not achieve major improvements over static embeddings, likely because descriptions on KGC datasets tend to be short. Instead we use Wikipedia2Vec (Yamada et al., 2016), which contains phrase embeddings for entity names like ”Sheila Walsh”. If no phrase embedding is available, we use token-wise embeddings instead. If no embedding is available for a token, we use a vector of zeros as an “unknown” token. The resulting sequence of embeddings is aggregated by average pooling to obtain a single text-based embedding vector of the head entity $v_h \in \mathbb{R}^d$.

The key step of our approach is to learn transformation functions Ψ_r that align the text-based and graph-based embedding spaces such that $\Psi_r(v_h) {\approx} u_h$. When applying this mapping, the open-world entity's text-based embedding v_h is transformed into a graph-based proxy embedding $\Psi_r(v_h)$. Triples with h are scored by applying the KGC model from Equation 1 to $\Psi_r(v_h)$:

$$t^* = \arg\max_{t \in E} \ \phi\Big(\Psi_r(v_h), u_r, u_t\Big) \tag{2}$$

Like OWE (Shah et al., 2019), we use an affine transformation $\Psi_r(v) = A_r {\cdot} v + b_r$. Our focus of this paper is to deal with relation specificity, for which we propose the following two strategies:

Relation Specific Transformation (RST) While the OWE model consists of a global transformation function (Ψ), our proposed RST approach trains a separate transformation function per relation (Ψ_r): Our hypothesis is that when predicting a tail $(h, r, ?)$, including information on the relation r may be beneficial. Have a look at Table 1, where the transformation $\Psi_{birthPlace}$ maps v_{Shiela_Walsh} to a completely different region in the graph embedding space than $\Psi_{recordLabel}$.

Therefore, we learn a separate transformation Ψ_r for each relation $r{\in}R$, containing a separate learnable matrix A_r and vector b_r. For a fair comparison with OWE, we use the ComplEx KGC model (Trouillon et al., 2016) in our experiments and use separate parameters for the real and imaginary parts.

Relation Clustering Transformation (RCT) Our second approach – Relation Clustering Transformation (RCT) – aggregates relations to clusters and then learns a separate transformation Ψ_C for each cluster C. We use an agglomerative clustering approach (pseudo-code in Figure 2), which first initializes each cluster with a single relation $r{\in}R$ and conducts η fusion steps, each joining “similar” clusters: Let $t(C)$ denote the set of tails attached to any relation r in C. Clusters C, C' are fused if the number of shared tails, divided by size of the smaller cluster, exceeds a threshold $\mathcal{S}$:

$$\frac{|t(C) \cap t(C')|}{min(|t(C)|, |t(C')|)} > \mathcal{S} \tag{3}$$

3.1 Training

Our text embeddings v and graph embeddings u are based on pretrained models, such that the only parameters Θ to be learned are the relation matrices A_r and vectors b_r. First a KGC model is trained

	DBPedia50k				FB15k-237-OWE				FB20k			
Model	H@1	H@3	H@10	MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10	MRR
Target Filtering Baseline†	4.5	9.7	23.0	11.0	6.4	14.2	23.3	12.7	17.5	32.1	41.2	27.2
DKRL†	-	-	40.0	23.0	-	-	-	-	-	-	-	-
ConMask†	47.1	64.5	**81.0**	58.4	21.5	39.9	45.8	29.9	42.3	57.3	71.7	53.3
ComplEx-OWE-200†	49.0	62.3	73.6	57.7	29.1	41.0	52.7	37.3	44.2	55.9	68.2	52.3
ComplEx-OWE-300†	51.9	65.2	76.0	60.3	31.6	43.9	56.0	40.1	44.8	57.1	69.1	53.1
WOWE (Zhou et al., 2020)	52.7	66.5	76.9	61.2	31.9	44.1	56.4	40.4	45.2	58.3	70.0	54.1
ComplEx-RCT ($\eta : 6, \mathcal{S} : 0.8$)	54.2	68.6	79.1	63.1	33.0	45.7	58.2	41.7	46.4	59.4	71.0	55.0
ComplEx-RST	**55.7**	**69.1**	80.1	**64.3**	**33.2**	**46.0**	**58.8**	**42.0**	**49.8**	**63.5**	**75.5**	**58.7**

Table 2: Comparison with other open-world KGC models on tail prediction. The Relation Specific Transformation (ComplEx-RST) performs best, particularly on the DBPedia50K and FB20K datasets with long textual descriptions († results reported by Shah et al. (2019)).

on the full graph $\mathcal{G}$, obtaining graph-based entity embeddings $u_1, ..., u_n$. We then choose the subset E^t of all entities in the graph with textual descriptions, and define $\mathcal{G}^t := \mathcal{G} \cap (E^t \times R \times E)$ as all triplets containing heads with text. We then minimize the following loss:

$$L(\Theta) = \sum_{(h,r,t) \in \mathcal{G}^t} \text{dist}\Big(\Psi_r(v_h), u_h\Big) \tag{4}$$

with *dist()* referring either to Euclidean or cosine distance between the graph- and text-based head embeddings. As we use ComplEx, where embeddings u_h contain real and imaginary parts, the above loss is summed for both parts. Since the number of entities in the datasets used is limited and overfitting is expected to be an issue, we neither fine-tune into the graph nor the text-embeddings.

4 Evaluation

We evaluate our model on FB20k (Xie et al., 2016), DBPedia50k (Shi and Weninger, 2017a) and FB15k-237-OWE (Shah et al., 2019) and compare our model with the state of the art on the task in open-world tail prediction. Results are illustrated in Table 2. Due to the lack of an open-world validation set on FB20k, we remove random 10% of the test triples and use them as a validation set. Hyperparameters were optimized using a grid search (details in the appendix). We use the same evaluation criteria as Shah et al. (2019), and evaluate our results only on ComplEx to provide a fair comparison with the OWE models. For training the closed-world KGC models, we utilize OpenKE (Han et al., 2018). Additionally, we use the target filtering approach (Shi and Weninger, 2017a) on any results reported. The Target Filtering Baseline is evaluated by assigning random scores to all targets that pass the target filtering criterion.

Results We observe that the ComplEx-RST outperforms all other approaches – including OWE – by a margin on all metrics except Hits@10 on DBPedia50k. ComplEx-RCT (relation clusters) performs competitively with the ComplEx-RST (one mapping per relation), while the number of transformation functions is reduced from 351 to 279 in case of DBPedia50k, from 235 to 114 in case of FB15k-237-OWE and from 1341 to 522 in case of FB20k. The values of η and $\mathcal{S}$ were optimized to achieve the greatest reduction in number of clusters at a negligible negative impact on accuracy. We also note that the improvement achieved by utilizing the relation information is higher in DBPedia50k and FB20k, both of which use very long descriptions compared to FB15k-237-OWE. We believe that this is because the longer descriptions often have more pieces of information relevant to the relation, which the relation-specific transformations are able to extract and utilize.

Finally, in Figure 2 (right) we investigate which relations benefit strongest from relation-specific mappings: Each point represents a relation in FB15k-237-OWE. We observe that points to the left (rare relations) tend to benefit stronger from learning a transformation of their own (RST). Those scarce relations seem to be underrepresented in the training data and – accordingly – in the global mapping.

```
Input: Training set $G$, similarity factor
    $\mathcal{S} \in \mathbb{R}$, iterations $\eta \in \mathbb{Z}$
Output: Clusters $C = \{c_1, c_2, \ldots, c_m\}$
for $i \leftarrow 1 \ldots |R|$ do
    $c_i \leftarrow \{r_i\}$
Procedure make_clusters ($C, \eta$)
    if $\eta = 0$ then
        return $C$
    for $i \leftarrow 1 \ldots |C|$ do
        $t(c_i) \leftarrow$ get_tails ($c_i, G$)
        for $j = i+1 \ldots |C|$ do
            $t(c_j) \leftarrow$ get_tails ($c_j, G$)
            if $\frac{|t(c_i) \cap t(c_j)|}{min(|t(c_i)|, |t(c_j)|)} > \mathcal{S}$ then
                $c_i \leftarrow c_i \cup c_j$
                delete $c_j$ from $C$
                break
    make_clusters ($C, \eta - 1$)
end
make_clusters ($\{c_1 \ldots c_{|R|}\}, \eta$)
```

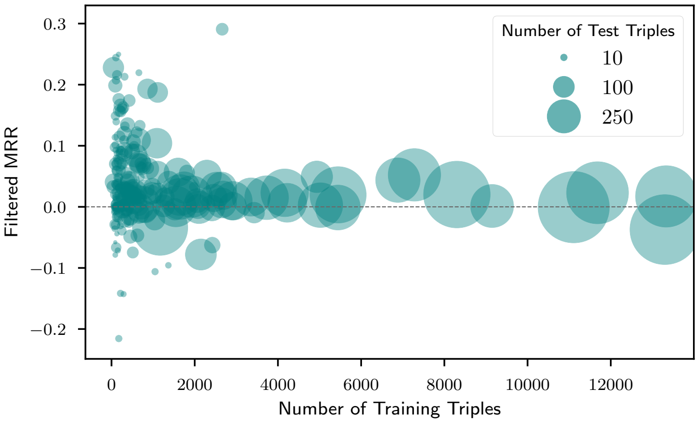

Figure 2: The algorithm (left) outlines our implementation of the RCT approach in pseudo code. The plot (right) visualizes the improvement in MRR of ComplEx-RST compared to ComplEx-OWE on the y-axis. Each point is a relation in FB15k-237-OWE. The size of the point represents the number of test triples containing the relation. The x-axis shows the number of training triples containing the relation. We see that this improvement tends to be higher for scarce relations (left on the x-axis).

5 Conclusion

We have proposed a simple approach to incorporate relation-specific information into open-world knowledge graph completion. Our approach achieves state-of-the-art results on common open-source benchmarks and offers strong improvements over relation-agnostic state-of-the-art methods. An interesting direction for future work will be to adapt our model to longer textual inputs, e.g. by using attention to enable the model to select relevant passages (similar to ConMask (Shi and Weninger, 2017a)).

Acknowledgements

This work was funded by German Federal Ministry of Education and Research (Program FHprofUnt, Project DeepCA (13FH011PX6)).

References

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating Embeddings for Modeling Multi-relational Data. In *Adv. in Neural Information Processing Systems*, pages 2787–2795.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. 2014. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, page 601–610, New York, NY, USA. Association for Computing Machinery.

David Ferrucci, Eric Brown, Jennifer Chu-Carroll, James Fan, David Gondek, Aditya A. Kalyanpur, Adam Lally, J. William Murdock, Eric Nyberg, John Prager, Nico Schlaefer, and Chris Welty. 2010. Building Watson: An Overview of the DeepQA Project. *AI Magazine*, 31(3):59–79.

Genet Asefa Gesese, Russa Biswas, Mehwish Alam, and Harald Sack. 2019. A survey on knowledge graph embeddings with literals: Which model links better literal-ly?

Yoav Goldberg. 2016. A Primer on Neural Network Models for Natural Language Processing. *J. Artif. Int. Res.*, 57(1):345–420, September.

Xu Han, Shulin Cao, Lv Xin, Yankai Lin, Zhiyuan Liu, Maosong Sun, and Juanzi Li. 2018. Openke: An open toolkit for knowledge embedding. In *Proceedings of EMNLP*.

Agustinus Kristiadi, Mohammad Asif Khan, Denis Lukovnikov, Jens Lehmann, and Asja Fischer. 2019. Incorporating literals into knowledge graph embeddings. In *The Semantic Web - ISWC 2019 - 18th International Semantic Web Conference, Auckland, New Zealand, October 26-30, 2019, Proceedings, Part I*, pages 347–363.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *Twenty-ninth AAAI conference on artificial intelligence*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. *CoRR*, abs/1310.4546.

Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. 2016. A Review of Relational Machine Learning for Knowledge Graphs. *Proceedings of the IEEE*, 104(1):11–33.

Heiko Paulheim. 2017. Knowledge Graph Refinement: A Survey of Approaches and Evaluation Methods. *Semantic Web*, 8(3):489–508.

Haseeb Shah, Johannes Villmow, Adrian Ulges, Ulrich Schwanecke, and Faisal Shafait. 2019. An open-world extension to knowledge graph completion models. In *The Thirty-Third AAAI Conference on Artificial Intelligence*, pages 3044–3051.

Baoxu Shi and Tim Weninger. 2017a. Open-World Knowledge Graph Completion. *CoRR*, abs/1711.03438.

Baoxu Shi and Tim Weninger. 2017b. ProjE: Embedding Projection for Knowledge Graph Completion. In *Proc. AAAI*, pages 1236–1242.

Richard Socher, Danqi Chen, Christopher D. Manning, and Andrew Y. Ng. 2013. Reasoning with neural tensor networks for knowledge base completion. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'13, pages 926–934, USA. Curran Associates Inc.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019. Rotate: Knowledge graph embedding by relational rotation in complex space. *arXiv preprint arXiv:1902.10197*.

Kristina Toutanova and Danqi Chen. 2015. Observed Versus Latent Features for Knowledge Base and Text Inference. In *3rd Workshop on Continuous Vector Space Models and Their Compositionality*, July.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex Embeddings for Simple Link Prediction. In *Int. Conference on Machine Learning*, pages 2071–2080.

Zhigang Wang and Juanzi Li. 2016. Text-enhanced Representation Learning for Knowledge Graph. In *Proc. International Joint Conference on Artificial Intelligence*, pages 1293–1299.

Ruobing Xie, Zhiyuan Liu, Jia Jia, Huanbo Luan, and Maosong Sun. 2016. Representation Learning of Knowledge Graphs with Entity Descriptions. In *Proc. AAAI*, pages 2659–2665.

Jiacheng Xu, Xipeng Qiu, Kan Chen, and Xuanjing Huang. 2017. Knowledge Graph Representation with Jointly Structural and Textual Encoding. In *Proc. Int. Joint Conference on Artificial Intelligence*, pages 1318–1324.

Ikuya Yamada, Hiroyuki Shindo, Hideaki Takeda, and Yoshiyasu Takefuji. 2016. Joint Learning of the Embedding of Words and Entities for Named Entity Disambiguation. In *Proc. SIGNLL Conference on Computational Natural Language Learning*, pages 250–259, Berlin, Germany, August.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2014. Embedding Entities and Relations for Learning and Inference in Knowledge Bases. *CoRR*, abs/1412.6575.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. Kg-bert: Bert for knowledge graph completion.

Yueyang Zhou, Shumin Shi, and Heyan Huang. 2020. Weighted aggregator for the open-world knowledge graph completion. In *Data Science*, pages 283–291, Singapore. Springer Singapore.