Journal of Artificial Intelligence Research 78 (2023) 269-285 Submitted 10/2022; published 10/2023

Select and Augment: Enhanced Dense Retrieval Knowledge Graph Augmentation

Micheal Abaho MICHEAL.ABAHO@LIVERPOOL.AC.UK
University of Liverpool, United Kingdom

Yousef H. Alfaifi Y_ALFAIFI@UT.EDU.SA
Faculty of Computers and Information Technology,
University of Tabuk, Tabuk, Saudi Arabia

Abstract

Injecting textual information into knowledge graph (KG) entity representations has been a worthwhile expedition in terms of improving performance in KG oriented tasks within the NLP community. External knowledge often adopted to enhance KG embeddings ranges from semantically rich lexical dependency parsed features to a set of relevant key words to entire text descriptions supplied from an external corpus such as wikipedia and many more. Despite the gains this innovation (Text-enhanced KG embeddings) has made, the proposal in this work suggests that it can be improved even further. Instead of using a single text description (which would not sufficiently represent an entity because of the inherent lexical ambiguity of text), we propose a multi-task framework that jointly selects a set of text descriptions relevant to KG entities as well as align or augment KG embeddings with text descriptions. Different from prior work that plugs formal entity descriptions declared in knowledge bases, this framework leverages a retriever model to selectively identify richer or highly relevant text descriptions to use in augmenting entities. Furthermore, the framework treats the number of descriptions to use in augmentation process as a parameter, which allows the flexibility of enumerating across several numbers before identifying an appropriate number. Experiment results for Link Prediction demonstrate a 5.5% and 3.5% percentage increase in the Mean Reciprocal Rank (MRR) and Hits@10 scores respectively, in comparison to text-enhanced knowledge graph augmentation methods using traditional CNNs.

1. Introduction

Jointly learning relational information by using both textual mentions and knowledge graph (KG) mentions of entity pairs has improved performance in not just knowledge base (KB) completion tasks, such as link and relation prediction (Bordes et al., 2013; Gardner et al., 2014), but also in various other NLP tasks such as fact retrieval (Bordes et al., 2013) and analogical reasoning (Gentner & Maravilla, 2017; Mikolov et al., 2013). More so, these Text-enhanced knowledge graph embedding (KGE) methods have been recently used to enrich representations of entities in order to improve performance in domain-specific tasks such as Biomedical Named Entity Recognition (Yuan et al., 2021), Medical Natural Language Inference MedNLI (Michalopoulos et al., 2020), and Normalising medical concepts (Limsopatham & Collier, 2016).

Typically, the objective in adapting text to the KG is to maximise the similarity between vectors encoding KG entities, and vectors encoding text descriptions in which the entities are

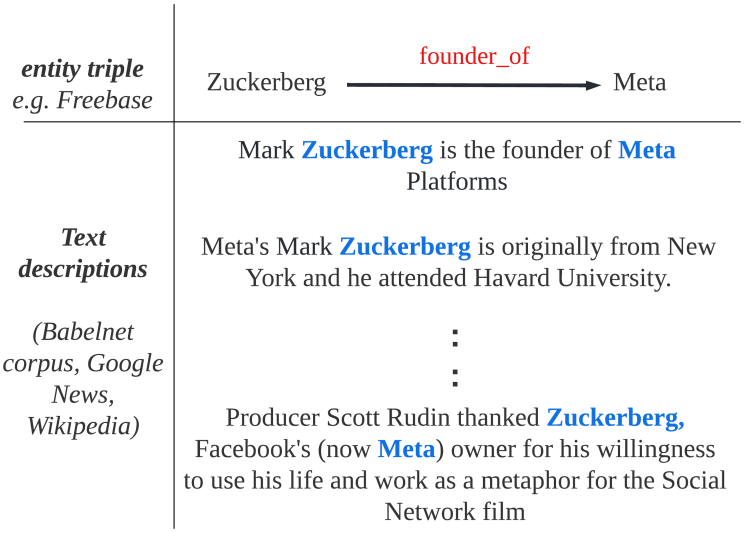

Figure 1: An example of KB triple and corresponding set of relevant text descriptions.

mentioned. For example, given a fact such as "Zuckerberg is the founder of MetaPlatforms", Zhong et al. (2015) aligns a KGE model vector for the entity Zuckerberg to a text embedding model vector of the same entity, however obtained from a text depicting this fact. These aligned representations generated for both object and subject entities would then be encoded by a model that uses a scoring function to indicate existence or non-existence of a relationship triple between the subject and object i.e. *(subject, relation, object)*.

We notice that prior efforts aligning textual descriptions to KG's often rely on a single text description of an entity which is often provided in the KB being studied or an external KB, such as Wikipedia (Toutanova et al., 2015). More so, some works constrain a relevant text description to a sequence of words within a text window of a predefined size (Xie et al., 2016; Zhong et al., 2015). The downside of these works is the assumption that a single description accompanying an entity in a KB would sufficiently and comprehensively represent the entity. This assumption is however, an over-estimated and exaggerated expectation of a single entity description. Even if it were a quality description of the entity, it would not contain all co-occurrences of the entity and its related entities. Moreover, word sense disambiguation research indicates that a word can have multiple glosses (word senses) depending on the context in which its used (Huang et al., 2019; Blevins & Zettlemoyer, 2020; Scarlini et al., 2020).

To address this lexical ambiguity, as well as concerns around the exaggerated assumption mentioned in previous paragraph, Kartsaklis et al. (2018) arbitrary gathers textual descriptions for an entity from numerous sources such as WordNet, Wikipedia, and FrameNet. They then use a term-based weighting mechanism (Tf-idf) for computing probabilities of an entity. Recently, Veira et al. (2019) augmented KG entity embeddings using word2vec embeddings obtained by training word2vec (Mikolov et al., 2013) on an external corpus that contained mentions of the KG entities. These efforts are worthy of recognition in terms of minimising nuances brought by lexical ambiguity, however, we recognize two challenges, 1) the arbitrary gathered formal descriptions of entities may not contain an entity's related entities i.e. no

co-occurrence of related entities, more so, the gathering process may un-wantedly become tedious for enormous KGs, 2) they utilize traditional vectorization techniques, which also have their limitations. Tf-idf ignores properties such as word-order and co-occurrence statistics, which are necessary when generating a semantic vector representation, and word2vec learns the same vector for a word irrespective of the context in which it is used.

In this work, we propose Dense Retrieval KG Augmentation (DRKA), a multi-task framework which uses dense retrieval to obtain documents or text descriptions semantically relevant to an entity pair, and subsequently augments a given KG with dense representations. This method addresses the first challenge in preceding paragraph by introducing a retriever model which automatically selects a richer set of descriptions to use for augmenting KG embeddings. To address the second challenge, DRKA takes full advantage of transformer-based SBERT (Reimers & Gurevych, 2019) to encode text descriptions. SBERT (Sentence-BERT) and other contextual language models (CLMs) have proven their superiority over traditional embedding methods in various downstream NLP tasks (Brown et al., 2020). Generally, DRKA builds on the idea of using multiple entity descriptions (Kartsaklis et al., 2018; Veira et al., 2019) to establish a relation between a pair of entities, such as in Figure 1, where the multiple sentences are used to establish the relation triple *(Zuckerberg, founder_of, Meta)*.

Similar to Document level relation extraction tasks (Zhou et al., 2021), DRKA leverages CLMs ability to capture interactions among distantly or remotely connected entities. Relying on multiple text descriptions for an entity, DRKA is able to increase the probability of co-occurrence of KG entity pairs within text, which inadvertently minimises the challenge of without mention entity pairs if a small text window is considered (Kartsaklis et al., 2018; Veira et al., 2019). DRKA learns to jointly embed KG mentions and textual mentions of an entity in the same embedding space. Our proposed method is evaluated on KG completion tasks (described under Section 4.3) such as link and relation prediction using Freebase FB15k dataset Veira et al. (2019). As later shown, there is approximately 6% and 3% percentage increase in the overall Mean Reciprocal Rank (MRR) and Hits@10 scores respectively, for the link prediction (LP) task obtained by DRKA. This is in contrast to DKRL (Xie et al., 2016), a model that uses Continuous bag of words (CBOW) and a CNN to generate description based representations of entities. The evaluation results are indicative of the effectiveness of augmenting KG embeddings with dense contextualised representations encoded from multiple text descriptions rather than a single text description.

2. Related Work

There are several works on augmenting KGs for purposes on improving performance in link prediction tasks. This section categorises related work into three areas, these include, Knowledge bases, Text-enhanced knowledge graph completion and Dense representation learning.

Knowledge Bases: KBs are often adopted in distance supervised learning (Mintz et al., 2009) because they are incomplete, implying that they do not possess all existing knowledge for the domains they represent, or at the bare minimum, that they do not explicitly state knowledge in its basic granular form (Reschke et al., 2014). There have been several efforts to alleviate the bottleneck of incompleteness such as entity linking across domains-based

graphs (Schneider et al., 2022) and integrating entities mentioned in unstructured text (Toutanova et al., 2015; Kartsaklis et al., 2018; Veira et al., 2022). Another rapidly growing paradigm that has attracted a lot of attention to KBs, is Language Models as Knowledge bases (LM-as-KBs). LM-as-KBs suggests that neural language models (LMs) can be treated as suitable alternatives or at least a proxy for KBs (Petroni et al., 2019). To achieve this, researchers design experiments in which they train LMs to learn to correctly answer prompts (Petroni et al., 2019; Heinzerling & Inui, 2021; Gao et al., 2021). Quite clearly, the subject of KBs has attracted a lot of research across domains with a keen interest in knowledge representation, management and dissemination. The focus in this work is augmenting KGE representations with external text. Moreover, we aim to enhance the augmentation process by enabling selection of richer text descriptions as opposed to simply adopting formal entity descriptions that may not sufficiently represent the entity in the context of its relationship with other entities.

Text-enhanced Knowledge Graph Completion: To improve performance in tasks such as LP, several authors have undertaken efforts to augment KG embeddings using external text. Wang et al. (2016) recently used co-occurrences between entities and words in text to enrich entity and relation representations in order to better handle 1-to-N, N-to-1, and N-to-N relations. They used point-wise and pairwise contexts using co-occurrence frequencies to build textual context embeddings, which were then used to enrich embeddings of the KG components. Kartsaklis et al. (2018) extends a KG by adding Tf-idf weighted terms from textual descriptions of entities, and later uses a multi-sense LSTM to learn multi-sense embeddings in order to achieve sensitivity towards lexically ambiguous words, i.e. words having more than one disjoint meaning (homonymy) and words with multiple different meanings (polysemy). (Riedel, Yao, McCallum, & Marlin, 2013) combines text-driven KG-based relations in the same entity-pair co-occurrence matrix, which are subsequently decomposed to obtain entity embeddings. Toutanova et al. (2015), Veira et al. (2022) use Lexicalised Dependency Paths (LDPs) obtained from sentences that co-occur in a text corpus as textual relations in a KG.

Similar to the above works, our work incorporates external text into a KG, however, it differs form them in such a way that, instead of using a single description, it relies on multiple text descriptions when augmenting a KG entity embedding. Our work further distinctively differs from other works that have used embeddings initialised by training word2vec on a corpus (Veira et al., 2019) in 2 ways, the first being, introduction of a description retrieval task in the augmentation process, thereby having a multi-task framework that learns to jointly select relevant descriptions as well as align these descriptions to KG embeddings. The second difference being the use of transformer-based SBERT, which provides high quality sentence embeddings that capture both the semantic and syntactic information of a sentence (Reimers & Gurevych, 2019).

Dense Representation Learning: Dense embedding models such as BERT (Devlin et al., 2018) have demonstrated an ability to effectively capture the semantics of words and sentences by encoding information about their neighbouring words and sentences, and the overall contexts in which they are mentioned. Recently, some works have shown that dense embeddings have surpassed term-based weighting schemes in tasks that require context retrieval from a piece of text in order to solve downstream tasks, such as Question Answering

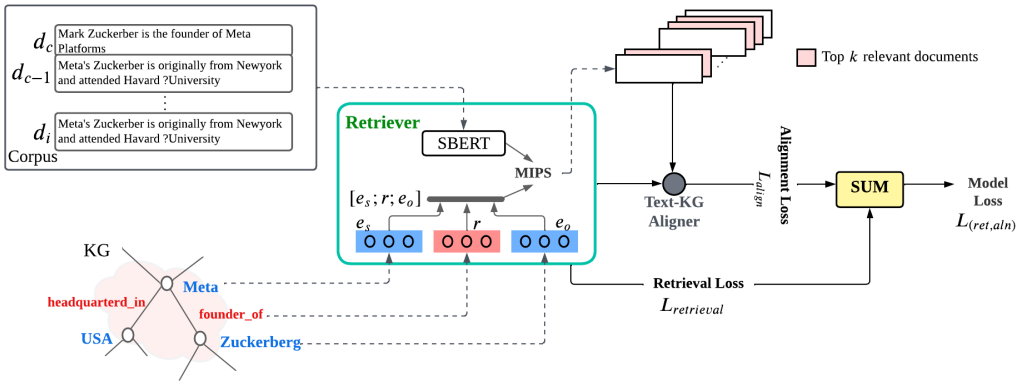

Figure 2: An overview of the proposed DRKA framework. DRKA combines a Retriever, which retrieves a set of documents (text descriptions) relevant to a relation triple and an Text-KG Aligner which fuses the identified relevant documents to KG embeddings. The last component of the framework shows a joint loss that involves summing up retrieval and alignment losses.

(QA) and Analogical Reasoning (Karpukhin et al., 2020; Izacard & Grave, 2020). Similar to how prior work has used these models for retrieval augmented generation to generate answers to questions in QA tasks (Izacard & Grave, 2020), this work explores enhancing KG augmentation using dense retrieval of entity descriptions from a corpus of text. We integrate this retrieval task as an auxiliary task to provide dense representations corresponding to multiple text descriptions that are semantically related to an entity.

3. Method

In this work, we assume access to a KG and a corpus of documents (text descriptions). Both of these are taken as input to the DRKA framework illustrated in Figure 2. A pre-trained SBERT model (Reimers & Gurevych, 2019) is used to encode text descriptions and a KG embedding model used to encode KG entities. Maximum Inner Product Search (MIPS) aided by an attention mechanism is used for finding the top k documents relevant to the query which is a concatenation of the triple elements. Concatenation is performed in order to allow knowledge transfer across elements within a triple, hence when searching for entity relevant documents, DKRA relies on knowledge of not just the entity but also its related entities. Subsequently the identified relevant documents embeddings are aligned to the KG embeddings and the model is trained by optimising the joint loss which is a summation of the losses with respect to retrieval and alignment.

3.1 Dense Retrieval KG Augmentation (DRKA)

Given a graph (for a certain KB) KG consisting of facts expressed in form of triples (e_s, r, e_o), where e_s and e_o are subject (head) and object (tail) entities respectively, and r is the relation linking the two together, we propose DRKA, a method that learns to cast entity embeddings and their text description embeddings into a common vector space, in order to accurately infer missing links or triples.

Different from prior work that arbitrary choose and select descriptions to use for the entities, this work considers learning a dense retriever to automatically select top k most relevant text descriptions for a particular entity triple. The idea is to maximise a LM's ability to capture missing facts from multiple sources as earlier indicated in the introduction. Similar to Dense Passage Retrieval (DPR) (Karpukhin et al., 2020), this retriever computes an inner product to indicate the similarity between a dense representation of a text description and the dense representation of an entity triple. To incorporate textual information, we leverage pre-trained SBERT (Reimers & Gurevych, 2019) to encode input text descriptions and learn to assign a similarity score between each sentence and a given query (triple), as described in Section 3.1.1. These scores are treated as attention scores which are used to compute an overall representation of the triple which is then aligned to the relevant text descriptions as described in Section 3.1.2.

3.1.1 Dense Description Retrieval

We initialise entities $e \in \mathcal{E}$ using TransE (Bordes et al., 2013) which follows the formulation $\|\vec{e_s} + \vec{r} - \vec{e_o}\|$, indicating that an object embedding $\vec{e_o}$ should be in close proximity to the summation of its corresponding subject and relation embeddings i.e. $\vec{e_s} + \vec{r}$. For a triple e_t, an embedding $\vec{e_t}$ is obtained through a concatenation of $[\vec{e_s}; \vec{r}; \vec{e_o}]$, which is then used as a query to search a corpus of external documents C for documents relevant or semantically similar to e_t. To answer this query, we compute an inner product between a dense vector representation $\vec{e_t}$ of the query e_t, and another projected dense vector representation $\vec{d_i'}$ of a description d_i. $\vec{d_i'}$ is computed as $\vec{d_i'} = W_c\vec{d_i} + b_c$, where $\vec{d_i} = \text{SBERT}(d_i)$, W_c is a matrix used to project the description's embeddings to the same space as the triple embedding, and b_c is a bias term. Equation (1) shows the computation for a similarity score between triple e_t and description d_i.

$$sim(e_t, d_i) = \vec{e_t} \cdot \vec{d_i'}^{\top} \tag{1}$$

where $\vec{e_t}$ is a matrix of three vectors corresponding to the triple elements i.e. $\vec{e_t} \in \mathbb{R}^{|e_t| \times l}$, $|e_t| = 3$, $\vec{d_i'} \in \mathbb{R}^{1 \times l}$, l is the embedding dimension size, and $sim(e_t, d_i)$ is a vector of similarity scores between elements of the triple e_t elements and description d_i.

3.1.2 Text-KG Aligner

For a triple e_t, we search for the k most relevant documents by computing an attention score across all documents in C. This attention is a normalised similarity score between the documents and the triple embedding $\vec{e_t}$ as shown in (2)). Since $sim(e_t, d_t)$ returns a vector of similarity scores, we compute its L2 norm prior to the normalisation.

$$\vec{A_t} = \frac{\exp(\|sim(\vec{e_t} \cdot \vec{d_i'})\|)}{\sum_{\vec{d_k} \in C} \exp(\|sim(\vec{e_t} \cdot \vec{d_i'})\|)} \tag{2}$$

Prior authors have benefited from generating overall representations of sentences or tokens by using attention scores to compute a weighted sum (Abaho et al., 2021, 2022). Inspired by

these works, we generate an overall triple representation $\vec{e_t^k}$ by computing an inner product between a matrix $\vec{D_t^k}$ of top k projected description embeddings relevant to triple e_t i.e. $\vec{D_t^k} \in \mathbb{R}^{k \times l}$ and the attention vector A_t as shown in (3),

$$\vec{e_t^k} = \vec{A_t}^\top \cdot \vec{D_t} \tag{3}$$

where $\vec{A_t} \in \mathbb{R}^{k \times 1}$ and $\vec{e_t^k} \in \mathbb{R}^{1 \times l}$. The attention vector scores indicate the strength of association or how relevant each document is to the triple in question.

3.1.3 Joint retrieval and Text-KG alignment

The proposed DRKA model is trained to jointly optimize the retrieval of relevant documents (text descriptions) as well as the alignment of the documents to a KG. During training, the trainable matrix W_c used to cast text description embeddings into the same vector space as the KG entity embeddings is updated alongside the embeddings. The model is trained to minimise the loss in (4) and illustrated in Figure 2.

$$L(ret, aln) = L_{align} + \alpha L_{retrieval} \tag{4}$$

α is a tunable parameter to adapt the auxiliary retrieval loss to the Text-KG alignment. We use a margin-based loss function to formulate our L_{align} loss function presented in (5).

$$L_{align} = - \sum_{\vec{e_t^k} \in E_t} \sum_{\vec{e_t^{k'}} \in E_t'} max(\gamma + d(\vec{e_t^k} - \vec{e_t^{k'}}), 0) \tag{5}$$

where $\gamma \geq 0$, $d(\vec{e_t^k} - \vec{e_t^{k'}})$ is a dissimilarity function in which we use an L1 norm, having performed well in prior knowledge graph representation approaches (Xie et al., 2016). $E_t^{'}$ is a set of negative training instances drawn as shown in (6).

$$\begin{aligned} E_t^{'} = \{(e_s^{'}, r, e_o) \mid e_s^{'} \in \mathcal{E}\} \cup \{(e_s, r, e_o^{'}) \mid e_o^{'} \in \mathcal{E}\} \\ \cup \{(e_s, r^{'}, e_o) \mid r^{'} \in \mathcal{R} \cup \{(e_s^{'}, r^{'}, e_o^{'}) \mid r^{'} \in \{\mathcal{E}, \mathcal{R}\}\} \end{aligned} \tag{6}$$

The retrieval loss minimised is given by (7),

$$L_{retrieval} = - \sum_{e_t \in E_t} \log \frac{\exp(\|sim(\vec{e_t} \cdot \vec{d_i})\|)}{\sum_{\vec{d_{i=1}}}^{|C|} \exp(\|sim(\vec{e_t} \cdot \vec{d_i})\|)} \tag{7}$$

4. Experiments

We evaluate the proposed DRKA method on three tasks: Link Prediction (Bordes et al., 2013), Relation Prediction (Weston et al., 2013; Yao et al., 2019) and Triplet classification (Zhong et al., 2015; Yao et al., 2019). We adopt the Freebase (FB15K) Knowledge graph, the

Dataset	Relations	Entities	Train / Val / Test Triples
FB15K	1,341	14,904	472,860 / 57,803 / 48,991
Text	3814190	14,308	244946 / 17572 / 14599

Table 1: Dataset statistics for both KB and text corpus

Babelnet corpus (Navigli & Ponzetto, 2012), Google News dataset, and Wikipedia articles (Veira et al., 2019), from which we assemble text descriptions for all the KG entities.

Unlike prior authors who often eliminate entity descriptions of short lengths (or with no description at all), we do not eliminate any description during our preliminary text-preprocessing phase. We hypothesise that discarding some descriptions on account of their short length might unwittingly eliminate relevant descriptions, instead we rely on k (the hyper-parameter), as discussed in Section 3.1.2, which if tuned well enough will enable us to obtain a sufficient number of text descriptions relevant to a triple. After pre-processing the gathered entity descriptions, the dataset is split into training, validation and test sets) and the resultant dataset statistics are presented in Table 4.

Table 4 provides a breakdown of the Train, Validation (Val) and Test splits used in our experiments and the 3.8M entity descriptions which contain 96% of the entities within the KB. As shown in the table, the text descriptions respectively cover 51.8%, 30.4% and 29.9% of the Training, Validation and Testing triples. Additionally, we have an average of 5 descriptions per entity and an average of 3 descriptions per entity pair.

Metrics: Following prior work on KG completion, we report two different metrics: the Mean Reciprocal Rank(MRR), and Hits@10. Percentages for MRR and Hits@10 are reported for the test sets across all experiments conducted.

4.1 Baselines

Besides traditional KGE models i.e. TransE (Bordes et al., 2013), DistMult (Yang et al., 2014), CompIEx (Trouillon et al., 2016) and RotatE (Sun et al., 2019), we use DKRL (Xie et al., 2016), a text-enhanced KGE that generates entity representations by adding structured based representations, obtained from TransE, to description-based representations obtained using either CBOW or CNN Encoder. The CNN Encoder takes word2vec word embeddings as input. Additionally, we consider DRKA(DPR) which decouples the retriever from the alignment/fusion (illustrated as Text-KG aligner) in Figure 2, in which case we formulate

Parameter	**Tuned-range**	**Optimal**
KG Embedding dimension	[50,100,200,300]	200
γ	[0.5,1.0,1.5,2.0]	1.0
Optimizer	[SGD,Adam]	Adam
Epochs	[20, 50, 70, 100, 120]	70
Learning rate	[5e-4, 1e-4, 5e-3, 1e-3, 5e-2, 1e-2]	1e-3

Table 2: Parameter settings for DRKA

the triple of elements as a sentence of (e_s, r, e_o) concatenated e.g. "*Zuckerberg founder of Meta*" (i.e. to serve as a query), and train DPR to select descriptions semantically relevant to this sentence and separately train DRKA with just L_{align} in Equation 5. For this work, we compare variants withf the KGE's for both DKRL's architecture and DRKA's architecture.

4.2 Training

We initialise four different KGE learning models (as Section 4.1 indicates), and use them to extract KG embeddings, and initialise SBERT for text embeddings. The number of negative samples per triple is set to 100 and k is set to 5. We tune all hyper-parameters using the validation data, and obtain optimal values as follows: learning rate - 1e-3, batch size - 8, KG embedding size - 200. Further details on tuning bounds are provided in Table 4.

4.3 Evaluation Results

Setup: We perform two sets of experiments for the different evaluation tasks. Initially, we use the KGE models as stand alone methods, and we later test KGE embeddings that are augmented with text description embeddings. We consider scenarios with triples whose entities co-occur within the text descriptions (With mentions), triples whose entities do not co-occur within the text descriptions (Without mentions) as well as all of the triples together (Overall).

Link Prediction (LP): LP is a popular KG completion task which attempts to evaluate how well KGE, text-enhanced KGE and Pre-trained LMs predict either a missing subject

		Overall		With mentions		Without mentions	
		MRR ↑	Hits@10 ↑	MRR ↑	Hits@10 ↑	MRR ↑	Hits@10 ↑
KG Only	TransE	36.8	52.4	34.5	<u>54.2</u>	<u>38.2</u>	<u>58.1</u>
	DistMult	36.3	51.8	34.1	53.3	36.6	55.7
	CompIE	<u>37.1</u>	<u>52.8</u>	<u>34.6</u>	52.7	37.4	55.9
	RotatE	**38.8**	**53.1**	**35.9**	**54.6**	**38.6**	**59.7**
KG + Text	DKRL(CNN) + TransE	38.9	54.1	38.7	**54.5**	<u>40.8</u>	58.6
	DKRL(CNN) + DistMult	37.8	52.9	37.7	52.4	39.8	58.4
	DKRL(CNN) + CompIE	38.1	54.2	39.5	53.8	41.4	58.9
	DKRL(CNN) + RotatE	40.6	54.8	41.3	57.4	39.8	60.1
	DRKA(DPR) + TransE$_{k=5}$	38.2	51.4	37.3	53.9	38.8	58.1
	DRKA(DPR) + DistMult$_{k=5}$	35.7	49.0	36.6	52.1	37.0	56.5
	DRKA + TransE$_{k=5}$	<u>41.2</u>	**55.7**	39.7	<u>54.1</u>	<u>41.8</u>	<u>61.1</u>
	DRKA + DistMult$_{k=5}$	40.3	<u>54.8</u>	39.3	54.1	39.2	59.4
	DRKA + CompIE$_{k=5}$	40.3	54.5	<u>40.2</u>	53.7	41.2	59.7
	DRKA + RotatE$_{k=5}$	**42.7**	**55.7**	**42.5**	**58.9**	**43.1**	**63.8**

Table 3: Link prediction results on the test split set on FB15K. The upper section includes results obtained in a KG standalone setup, where KGE models are used to learn from KG's alone; the lower section results are obtained when the KGE models are augmented using textual descriptions, as covered in Section 2 (Text-enhanced Knowledge Graph Completion). DRKA(DPR) Best and second-best results are formatted respectively as bold and underlined text, across each column for the KG and KG+Text setup.

entity from a given triple $(?, r, e_o)$, or a missing object entity from a triple $(e_s, r, ?)$. Table 3 shows performance results for the various models across the two different setups. We observe RotatE outperforming the other KGE across all 6 experiments in the KG only setup. However, we notice both sets of models perform slightly better with triples whose elements are not mentioned in text (without mentions), compared to those with textual mentions.

We notice that augmenting the KGE with text descriptions (KG + text setup) significantly improves the performance in all three scenarios, with and without mentions as well as overall. DRKA+RotatE produces the best results in majority of the experiments (5/6 to be precise) followed by DRKA+TransE. We observe the best performing model DRKA + RotatE$_{k=5}$ outperform the baseline architecture DKRL(CNN) + RotatE i.e. the average percentage increase in MRR and Hits@10 across the three setups is 5.5% and 3% respectively. We attribute this performance to the fact that DRKA injects richer, semantically relevant contextualised representations (of the text descriptions) obtained by a transformer-based (SBERT) retriever model. On the otherhand, DKRL(CNN)+TransE injects word2vec representations initialised using word2vec. While word2vec and other traditional word embedding models are context-insensitive (i.e. a word embedding is fixed irrespective of the context in which its mentioned), contextualised embedding models such as BERT dynamically produce word embeddings, in other words, different contexts trigger different embeddings for the same word. This ultimately enhances the learning of different meanings and senses (Loureiro et al., 2021) in language modelling tasks, such as those covered in this work. We additionally observe DRKA(DPR) models perform poorly in comparison to the other models, and we attribute this to error propagation as a result of decoupling the retrieving from the alignment process i.e. errors that originate from the retrieving process done by DPR automatically affect the alignment process. Furthermore, the query formulation process is

		Overall		Without mentions	
		MR ↓	Hits@1 ↑	MR ↓	Hits@1 ↑
KG Only	TransE	4.2	86.7	4.1	87.3
	DistMult	4.7	85.4	4.8	85.0
	CompIEx	<u>4.0</u>	<u>86.9</u>	<u>4.0</u>	<u>88.1</u>
	RotatE	**3.5**	**89.0**	**3.1**	**89.5**
KG + Text	DKRL(CNN) + TransE	2.9	88.1	2.7	89.4
	DKRL(CNN) + DistMult	3.1	87.6	3.0	88.1
	DRKA(DPR) + TransE$_{k=5}$	3.2	84.2	3.1	85.8
	DRKA(DPR) + TransE$_{k=5}$	3.7	82.7	3.3	84.1
	DRKA + TransE$_{k=5}$	<u>1.9</u>	91.5	<u>1.7</u>	<u>92.2</u>
	DRKA + DistMult$_{k=5}$	2.2	90.3	1.9	90.8
	DRKA + CompIEx$_{k=5}$	1.8	<u>91.8</u>	<u>1.7</u>	91.6
	DRKA + RotatE$_{k=5}$	**1.1**	**93.4**	**1.1**	**93.7**

Table 4: Relation prediction results (Mean Rank (MR) and Hits@1) for the KG Only and KG + Text setup on FB15K dataset. The lower(↓) the MR score, the better and the higher (↑) the Hits@1, the better.

	e_s, r, e_o	$e_s^{'}, r, e_o$	$e_s, r, e_o^{'}$	$e_s^{'}, r^{'}, e_o^{'}$
TransE	91.2	53.8	55.4	81.1
DistMult	89.1	49.8	55.5	80.6
DRKA + TransE	**95.7**	60.4	63.2	**84.5**
DRKA + DistMult	94.2	**61.5**	**63.3**	83.2
TransE*	91.2	58.2	60.2	83.6
DistMult*	89.1	54.3	59.1	82.7
DRKA + TransE*	**95.7**	66.8	65.6	**88.6**
DRKA + DistMult*	94.2	**70.2**	**72.7**	88.5

Table 5: Triplet classification accuracy (%) over various types of triples. * indicates that the corrupted entities are drawn from the text corpus, rather than from the KB from which the KG is constructed. Only TransE and DistMult are tested for these experiments.

simply a concatenation of the triple elements, rather than an actual question with elaborative context about the triple elements.

Relation Prediction (RP): Similar to LP, RP aims to predict a missing element of a triple, however RP specifically looks to predict a missing relation from $(e_s, ?, e_o)$. Table 4 shows RotatE and DRKA + RotatE dominating the performance in the KG stand alone setup and KG + Text setup respectively. These results prove that supplementing the structured KGE with text can leas to significant performance gains in gains in not just LP, but RP too.

Triplet Classification (TP): Similar to prior work, we define TP as a binary classification task which classifies an entity triple e_t as a valid or invalid triple. We adopt the evaluation protocol used by Wang et al. (2014) when generating negative samples i.e. we construct a false triple by corrupting a valid KG triple. For $(e_s, r, e_o) \in KG$, where $\{e_s, e_r\} \in \mathcal{E}$ we 1) replace e_s with a random entity $e_s^{'}$ 2) replace e_o with a random entity $e_o^{'}$ and 3) Replace both subject and object entities with random entities, where $\{e_s^{'}, e_o^{'}\} \in \mathcal{E}$. We further repeat steps 1 to 3 , yet this time sampling the corrupt entities from the text corpus.

Table 5 shows that the models struggle less in predicting validity of valid triples, as seen in column two e_s, r, e_o i.e. the models perform best in contrast to all the other triple types investigated.

It is noticeable that corrupting the subject or head entity e_s (in column 3) causes a bigger drop in performance compared to when the object or tail entity e_o is corrupted (in column 4). The models perform relatively well when tasked with classifying triples with invalid entities and relations (in column 5), despite performing worse with valid triple types. We also observe that sampling negative or corrupt entities from the corpus does not lead to the same performance deterioration as it does when they are sampled from the KG. This is attributed to fact that the retriever model selects a core set of related text descriptions that are relevant to a triple and hence enhancing the models ability to detect presence or absence of a triple within the text.

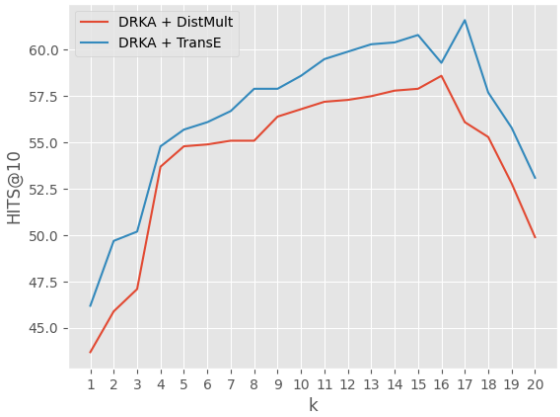

Figure 3: Adjusting k to determine how many text descriptions would be relevant in augmenting an entity representation for the Link Prediction task.

4.4 Ablation and Analysis

To further understand the performance of our proposed multi-task framework, we conduct a set of two other investigations. The first involves adjusting parameter k which specifies the number of text descriptions to use in augmentation, and the second involves eliminating the retriever component and simply use a randomly selected set of relevant text descriptions. These two investigations are detailed in the following subsections.

4.4.1 Adjusting k

To determine the optimal number of relevant descriptions as well as compare using a single text description in augmenting KG embeddings, we tune the k value (for the LP task) within the range $\{1, 20\}$ where 1 and 20 are the lower and upper bounds, respectively. The lower bound ultimately represents augmentation premised on a single description whereas, all $k \geq 1$ settings represent augmentation premised on multiple text descriptions. Figure 3 illustrates the significant improvement in performance when the number of relevant text descriptions (k) to use in the augmentation process is increased from 1 on wards. The figure shows a sharp rise in Hits@10 performance through from the first 5 k values, which is followed by a gradual increase all the way to $k = 16$. The performance is seen to decline in the last 5 k values. We attribute this to a possible overlap in descriptions for triples, which then confuses the model at inference. This decline is an indicator that, despite the gains made by using more descriptions, a high number of descriptions $k > 16$ may instead hurt the performance of the text-enhanced KG embedding model in tasks such as LP.

4.4.2 DRKA - Retriever

We obviate the role of the retriever and instead arbitrary select a set of entity descriptions existing in the Babelnet corpus (Navigli & Ponzetto, 2012), Google News dataset, and Wikipedia articles. The selected or targeted descriptions are still relevant to the entities, however, they are obtained in a distantly supervised manner, where we simply search and

select a description provided for a given entity. For a given triple (e_s, r, e_o), two pairs of relevant descriptions are randomly selected for the subject e_s and object e_o respectively. We chose the same number of descriptions (2) per entity to avoid skewing the semantics captured by embedding towards one entity and instead allow a proportionate distribution across both featuring entities.

These descriptions are aligned to the triple embedding $\vec{e_t}$ (obtained by concatenating triple elements) following the same procedure (under subsection 3.1.2) of computing an attention scores of these 4 documents to the triple and then generating an overall representation as an inner product between the attention scores and the description embeddings projected into the KG vector space. This setup however eliminates the retriever loss, thereby only minimising the alignment loss L_{align}.

	Link Prediction		Relation Prediction	
	MRR ↑	Hits@10 ↑	MR ↓	Hits@1 ↑
DRKA + TransE$_{k=4}$	**40.7**	**54.8**	<u>1.9</u>	**90.7**
DRKA + DistMult$_{k=4}$	40.1	<u>53.7</u>	2.3	<u>90.3</u>
DRKA - retriever + TransE$_{k=4}$	39.8	51.5	<u>1.9</u>	89.7
DRKA - retriever + DistMult$_{k=4}$	37.9	51.2	2.0	87.4
DRKA - retriever + TransE$_{k=6}$	<u>40.9</u>	52.1	**1.3**	89.3
DRKA - retriever + DistMult$_{k=6}$	39.1	51.9	1.8	88.4

Table 6: Link and Relation prediction results of DRKA with and withorugh the retriever component (- retriever) for two different values of k, ie. $k = 4$ and $k = 6$. The best and second best scores are in bold and underlined respectively.

Table 6 shows results of the multi-task DRKA frame work trained without the retriever component and instead a set of descriptions is selected from the corpus randomly but still relevant to the entities in a given triple as described in the preceding paragraphs. For consistency, we re-run the full DRKA framework with k set to 4 because its the same number considered in the DRKA - retriever setup. As shown in the table, there is a drop in performance when the retriever is deducted, more so, a significant drop in Hits@10. This drop is further evidence of the significance of a model trained to explicitly select a set of descriptions that are semantically relevant to the triple of entities.

To probe this impact further, we additionally test using $k = 6$, selecting 3 descriptions per entity. We hypothesize that an increase in the number of descriptions might subtly eliminate the need of a retriever model. On the contrary, we realise that the performance still drops however, to a degree lesser than it does in the $k = 4$ setting. The DRKA - retriever + TransE$_{k=6}$ achieves the best Mean rank score in the RP task. These changes indicate that select an appropriate number to use in augmenting is so critical and can have a good or detrimental impact on the performance of the model.

5. Conclusion

This paper has explored dense representation learning as a conduit for achieving KG augmentation. It proposes a retriever based augmentation model, called DRKA, that jointly

learns KG embeddings and contextualised embeddings of text produced by a dense representation model, SBERT. The initial set of evaluation experiments performed showed that augmenting KG embeddings with dense representations of text descriptions using DRKA improves performance in KG completion tasks such as Link Prediction, Relation Prediction and Triplet Classification. We are aware of the significant impact that text-enhanced KGE have had in KG completion, however, this paper has shown that increasing the number of text descriptions to use in augmenting KG embeddings can lead to further gains in performance of these models. Having said that, we observed that constantly increasing the number of descriptions to incorporate into the KG embedding may at a point, begins hurting the performance of the model. For further analysis, we investigate the impact of the retriever component within DRKA i.e. train and evaluate DRKA minus the retriever. We discover that deduction of the retriever hurts the performance of the model. Overall, this paper has empirically demonstrated that enhancing KGE models with semantically rich dense representations of text can benefit KG completion. Its proposed framework can be adapted to domain specific KG tasks and can eliminate the need to supply a set entity descriptions manually collated from multiple sources i.e the model is able to automate retrieval of relevant text descriptions to use for augmenting.

References

Abaho, M., Bollegala, D., Williamson, P., & Dodd, S. (2021). Detect and classify–joint span detection and classification for health outcomes. arXiv preprint arXiv:2104.07789.

Abaho, M., Bollegala, D., Williamson, P., & Dodd, S. (2022). Position-based prompting for health outcome generation. arXiv preprint arXiv:2204.03489.

Blevins, T., & Zettlemoyer, L. (2020). Moving down the long tail of word sense disambiguation with gloss-informed biencoders. arXiv preprint arXiv:2005.02590.

Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., & Yakhnenko, O. (2013). Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, *26*.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., & Amodei, D. (2020). Language models are few-shot learners. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., & Lin, H. (Eds.), *Advances in Neural Information Processing Systems*, Vol. 33, pp. 1877–1901.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Gao, T., Fisch, A., & Chen, D. (2021). Making pre-trained language models better few-shot learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 3816–3830, Online. Association for Computational Linguistics.

Gardner, M., Talukdar, P., Krishnamurthy, J., & Mitchell, T. (2014). Incorporating vector space similarity in random walk inference over knowledge bases. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 397–406.

Gentner, D., & Maravilla, F. (2017). Analogical reasoning. In *The Routledge International Handbook of Thinking and Reasoning*, pp. 186–203. PEETERS BONDGENOTENLAAN 153, B-3000 LEUVEN, BELGIUM.

Heinzerling, B., & Inui, K. (2021). Language models as knowledge bases: On entity representations, storage capacity, and paraphrased queries. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 1772–1791, Online. Association for Computational Linguistics.

Huang, L., Sun, C., Qiu, X., & Huang, X. (2019). Glossbert: Bert for word sense disambiguation with gloss knowledge. arXiv preprint arXiv:1908.07245.

Izacard, G., & Grave, E. (2020). Leveraging passage retrieval with generative models for open domain question answering. arXiv preprint arXiv:2007.01282.

Karpukhin, V., Oğuz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., & Yih, W.-t. (2020). Dense passage retrieval for open-domain question answering. arXiv preprint arXiv:2004.04906.

Kartsaklis, D., Pilehvar, M. T., & Collier, N. (2018). Mapping text to knowledge graph entities using multi-sense lstms. arXiv preprint arXiv:1808.07724.

Limsopatham, N., & Collier, N. (2016). Normalising medical concepts in social media texts by learning semantic representation. In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 1: long papers)*, pp. 1014–1023.

Loureiro, D., Rezaee, K., Pilehvar, M. T., & Camacho-Collados, J. (2021). Analysis and evaluation of language models for word sense disambiguation. *Computational Linguistics*, *47*(2), 387–443.

Michalopoulos, G., Wang, Y., Kaka, H., Chen, H., & Wong, A. (2020). Umlsbert: Clinical domain knowledge augmentation of contextual embeddings using the unified medical language system metathesaurus. arXiv preprint arXiv:2010.10391.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, *26*.

Mintz, M., Bills, S., Snow, R., & Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pp. 1003–1011.

Navigli, R., & Ponzetto, S. P. (2012). Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial intelligence*, *193*, 217–250.

Petroni, F., Rocktaschel, T., Riedel, S., Lewis, P., Bakhtin, A., Wu, Y., & Miller, A. (2019). Language models as knowledge bases?. In *Proceedings of the 2019 Conference on*

Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Reimers, N., & Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. arXiv preprint arXiv:1908.10084.

Reschke, K., Jankowiak, M., Surdeanu, M., Manning, C. D., & Jurafsky, D. (2014). Event extraction using distant supervision. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pp. 4527–4531.

Riedel, S., Yao, L., McCallum, A., & Marlin, B. M. (2013). Relation extraction with matrix factorization and universal schemas. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 74–84.

Scarlini, B., Pasini, T., & Navigli, R. (2020). With more contexts comes better performance: Contextualized sense embeddings for all-round word sense disambiguation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 3528–3539.

Schneider, P., Schopf, T., Vladika, J., Galkin, M., Simperl, E., & Matthes, F. (2022). A decade of knowledge graphs in natural language processing: A survey. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 601–614, Online only.

Sun, Z., Deng, Z.-H., Nie, J.-Y., & Tang, J. (2019). Rotate: Knowledge graph embedding by relational rotation in complex space. In *International Conference on Learning Representations*.

Toutanova, K., Chen, D., Pantel, P., Poon, H., Choudhury, P., & Gamon, M. (2015). Representing text for joint embedding of text and knowledge bases. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pp. 1499–1509.

Trouillon, T., Welbl, J., Riedel, S., Gaussier, É., & Bouchard, G. (2016). Complex embeddings for simple link prediction. In *International conference on machine learning*, pp. 2071–2080. PMLR.

Veira, N., Keng, B., Padmanabhan, K., & Veneris, A. G. (2019). Unsupervised embedding enhancements of knowledge graphs using textual associations.. In *IJCAI*, pp. 5218–5225.

Veira, N., Keng, B., Padmanabhan, K., & Veneris, A. G. (2022). Learning to borrow.. In *IJCAI*, pp. 5218–5225.

Wang, Z., Zhang, J., Feng, J., & Chen, Z. (2014). Knowledge graph and text jointly embedding. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1591–1601.

Wang, Z., Li, J., Liu, Z., & Tang, J. (2016). Text-enhanced representation learning for knowledge graph. In *Proceedings of International joint conference on artificial intelligent (IJCAI)*, pp. 4–17.

Weston, J., Bordes, A., Yakhnenko, O., & Usunier, N. (2013). Connecting language and knowledge bases with embedding models for relation extraction. arXiv preprint arXiv:1307.7973.

Xie, R., Liu, Z., Jia, J., Luan, H., & Sun, M. (2016). Representation learning of knowledge graphs with entity descriptions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 30.

Yang, B., Yih, W.-t., He, X., Gao, J., & Deng, L. (2014). Embedding entities and relations for learning and inference in knowledge bases. arXiv preprint arXiv:1412.6575.

Yao, L., Mao, C., & Luo, Y. (2019). Kg-bert: Bert for knowledge graph completion. arXiv preprint arXiv:1909.03193.

Yuan, Z., Liu, Y., Tan, C., Huang, S., & Huang, F. (2021). Improving biomedical pretrained language models with knowledge. arXiv preprint arXiv:2104.10344.

Zhong, H., Zhang, J., Wang, Z., Wan, H., & Chen, Z. (2015). Aligning knowledge and text embeddings by entity descriptions. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pp. 267–272.

Zhou, W., Huang, K., Ma, T., & Huang, J. (2021). Document-level relation extraction with adaptive thresholding and localized context pooling. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35, pp. 14612–14620.