LM4KG: Improving Common Sense Knowledge Graphs with Language Models

Janna Omeliyanenko[1], Albin Zehe[1], Lena Hettinger[1], and Andreas Hotho[1]

Julius-Maximilians-University Würzburg, Am Hubland, 97074 Würzburg, Germany
{omeliyanenko, zehe, hettinger, hotho}@informatik.uni-wuerzburg.de

Abstract. Language Models (LMs) and Knowledge Graphs (KGs) are both active research areas in Machine Learning and Semantic Web. While LMs have brought great improvements for many downstream tasks on their own, they are often combined with KGs providing additionally aggregated, well structured knowledge. Usually, this is done by leveraging KGs to improve LMs. But what happens if we turn this around and use LMs to improve KGs?
In this paper, we propose a method enabling the use of the knowledge inherently encoded in LMs to automatically improve explicit knowledge represented in common sense KGs. Edges in these KGs represent relations between concepts, but the strength of the relations is often not clear. We propose to transform KG relations to natural language sentences, allowing us to utilize the information contained in large LMs to rate these sentences through a new perplexity-based measure, Refined Edge WEIGHTing (REWEIGHT). We test our scoring scheme REWEIGHT on the popular LM BERT to produce new weights for the edges in the well-known ConceptNet KG. By retrofitting existing word embeddings to our modified ConceptNet, we create ConceptNet NumBERTbatch embeddings and show that these outperform the original ConceptNet Numberbatch on multiple established semantic similarity datasets.

Keywords: Knowledge Graph · Language Model · Common Sense

1 Introduction

Knowledge Graphs (KG) are one of the core areas of research in the Semantic Web community [11]. Their creation and curation have long been tasks of great interest, since the resulting graphs are invaluable in a wide range of applications within the community, but also in Natural Language Processing, Information Retrieval and Machine Learning. Thus, KGs provide a natural link between the Semantic Web and Machine Learning, where they are being used to provide explicit, structured background knowledge that may not be readily available in unstructured data sources. While the use of KGs in Machine Learning applications is very common [27,28,37], in this paper we propose to go in the opposite direction: We use a well-established model from the area of Natural Language

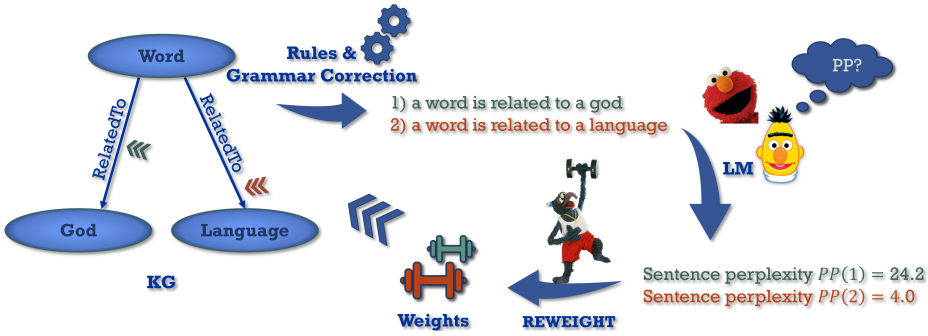

Fig. 1. Illustration of the REWEIGHT pipeline. A KG's relations are weighted through transforming them to sentences and obtaining perplexity scores with an LM.

Processing, the Language Model (LM) BERT [8] to improve the knowledge encoded in a widely used, state of the art KG, ConceptNet [28].

In order to supply a range of information that is as broad as possible, KGs are often constructed using (semi-)automatic methods, with ConceptNet combining multiple other knowledge bases (e.g., Wiktionary, DBPedia) as well as extracting additional information from plain text and games with a purpose [14,28,30]. While these sources are mostly reliable, there is no explicit information about the strength of the relations described therein. However, often it is crucial to know the strength of the relation between two words: For example, a search engine may want to perform a query expansion using a KG to look for related terms. When looking for terms related to "word", a KG like Wiktionary would return both "god" and "language". While both relations are correct, the second one is more prevalent in most situations and would usually be considered stronger, meaning that it will be the better choice for a query expansion in most contexts. However, Wiktionary contains no indication that there is a difference between both relations. ConceptNet partially deals with this issue by assigning reliability scores to different sources, but this does not help to distinguish between relations from the same source. Hence we are interested in automatically extracting this prevalence information from unstructured data and adding it as structured information to the graph by refining its edge weights.

Hypothesis In this paper, we propose Refined Edge WEIGHTing (REWEIGHT), a novel approach towards automatically acquiring the prevalence information of relations in order to weight the edges in a KG using pre-trained LMs such as BERT [8]. Recent research has shown that these models trained on enormous corpora contain a certain amount of world knowledge, in some cases even being able to perform limited question answering without ever being trained for that task or being given explicit background information [24]. Recent work on BERT specifically suggests that it may contain relevant common sense information [23,35]. Seeing LMs such as BERT as an automatic information extraction approach with access to a vast amount of data through training, we hypothesize

that, by representing a relation as a natural language sentence and asking BERT to rate how likely the sentence is to occur (i.e., calculating its perplexity), we will be able to automatically extract a weighting for common sense relations that corresponds well to a human rating.

Approach Our methodology, illustrated in Figure 1, can be summarised as follows: We use an existing KG as a starting point. From this graph, we extract all edges (corresponding to relations between two words) and construct sentences from these edges by applying a manually defined set of rules and automated grammar correction. The resulting sentences are then used as input to a LM and their perplexity is calculated. We apply a transformation to the perplexity scores to map them to the range of the edge weights in the original graph, where high edge weights correspond to strong relations. Finally, we feed the edge weights back into the KG, yielding an enriched knowledge resource that contains information about the prevalence of its relations. While we utilize BERT's common sense knowledge in our pipeline, we formulate the approach in a general way to allow application on all types of KGs with different LMs.

To evaluate our approach, we show that REWEIGHT is capable of improving the already well suited ConceptNet KG on the task of refining existing word embeddings for semantic relatedness. We evaluate REWEIGHT by applying the same retrofitting [10] procedure as ConceptNet Numberbatch [28], showing that the enriched graph yields embeddings with improved performance on multiple semantic relatedness datasets.

Contribution Our contribution in this paper is three-fold: 1. We propose a novel, general methodology for enriching KGs by weighting the edges in a KG according to their importance. 2. We update the weights of the common sense KG ConceptNet with the BERT LM, showing that our approach improves the semantic information encoded in the graph.[1] 3. We perform a detailed analysis, investigating different influence factors on our proposed approach.

Structure The remainder of this paper is structured as follows: Section 2 gives an introduction to common sense KGs and retrofitting, while Section 3 describes work related to our paper. In Section 4 we describe our edge weighting scheme REWEIGHT. Section 5 describes the experimental setup of our evaluation. Our results are contained in Section 6, while we carry out deeper analysis in Section 7. Section 8 concludes our work.

2 Background

In this section we describe the background setting of our paper. This includes a general overview of KGs for Common Sense Knowledge, where the most prominent representative is ConceptNet, and the Retrofitting algorithm that we use

[1] Code, updated KGs and embeddings are available under https://github.com/JohannaOm/REWEIGHT

to derive word embeddings from our modified KG. Following previous work, we will later use these embeddings to evaluate the quality of our new weights.

2.1 Common Sense Knowledge Graphs

Common sense describes the most basic knowledge and information a human has at their disposal [28].

In our experiments we focus on one of the most prevalent common sense KGs, **ConceptNet** [28]. ConceptNet aggregates its information from sources like DBPedia, Wiktionary, Open Multilingual WordNet and "games with a purpose". It is a multilingual KG specifically designed as an information source for assessing semantic relatedness between concepts, setting a special focus on natural language expressions. For our experiments, we use version 5.7, which contains 30.6 million relations between 17.8 million concepts, out of which 2.3 million relations exist between 440.000 English concepts. The graphs original weights are distributed between 0.1 and $\gamma_{\max} = 50$, with mean 1.1 and median $\tilde{\gamma} = 1$.

Next to ConceptNet, we also take a look at other common sense KGs. **WebChild 2.0** [30] is an English common sense KG focusing on activities, properties, and their semantic relations. It contains 23 million relations between 450.000 concepts. **YAGO 3.1** [25] is an ontology extracted from multilingual Wikipedia. Wile YAGO, as a general knowledge base, contains many facts about specific real world entities, we use YAGO's taxonomy subgraph that contains more abstract information that more closely represents generally applicable common sense knowledge. The YAGO Taxonomy contains 1.7 million relations between 800.000 concepts.

2.2 Evaluation of KGs

Evaluating KGs is a non-trivial problem, since there is usually no ground truth available that can directly be used as an intrinsic evaluation target. One possible way of extrinsic evaluation is using the KG to enrich existing word embeddings and assess the quality improvement in the embeddings induced through the KG. We will adopt this way of evaluation by following [29] in applying Retrofitting (cf. Section 2.3) to enrich word embeddings using either the unmodified version of a KG or the modified version after applying REWEIGHT. Retrofitting uses the weights in the KG as indication of how close two words should be, making this a suitable method of evaluating whether REWEIGHT actually improves the weights in the KG: If the quality of the embeddings improves after applying REWEIGHT, we can conclude that we have improved the weights in the KG.

2.3 Retrofitting

Retrofitting [10] uses relational information of KGs to refine existing word embeddings. The main idea is to re-calibrate the embedding vector of each word, leaving it both close to the original embedding vector and close to the embeddings of all neighboring words in the KG.

Formally, retrofitting minimizes the objective function

$$\Psi(Q) = \sum_{i \in V} \left[\alpha_i ||q_i - \hat{q}_i||^2 + \sum_{(i,j) \in E} \beta_{i,j} ||q_i - q_j||^2 \right] \quad (1)$$

where q_i and $\hat{q}_i$ mark the new and original embeddings of word i out of vocabulary V respectively, and $\beta_{i,j}$ represents the weight of the edge $(i, j) \in E$ in the KG, which we want to improve with REWEIGHT. α_i is a hyper-parameter determining how close q_i should stay to $\hat{q}_i$.

[28] use an extended version of retrofitting that is capable of processing out-of-vocabulary words. In this work we also use the same extended version.

3 Related Work

KGs and (large) LMs have been investigated extensively in recent years. One of the best understood large **language models** is BERT. It performs on par with knowledge bases extracted from text [23] and substantially outperforms pretrained word embeddings when queried for relational common sense knowledge [4]. When compared to other recently introduced LMs, BERT outperformed GPT2 and XLnet on a series of common sense tasks [38]. BERT also offers enough clues to enable common sense reasoning for visual understanding, and more so than GloVe and ELMo embeddings [35]. Hence, we conclude that BERT is a fitting model to extract common sense knowledge for KG enhancement.

In general, **knowledge resources** can be a vital addition to any NLP task. There exist various methods to create knowledge bases, e.g. from web resources like Wikipedia or through crowdsourcing as in ConceptNet. So far, relations from KGs have mostly been used to enrich LMs: either during training [32,34] or by adapting the resulting embeddings afterwards (*retrofitting*) [10,18,29]. A prominent example is ERNIE [36], which aligns KG entities from WikiData with NEs to then train contextual word embeddings similar to BERT. Experimental results show that ERNIE significantly outperforms BERT on knowledge-driven tasks such as relation classification. It is also possible to learn an improved word embedding by integrating human feedback [22] or by jointly exploiting a text corpus and a KG [2].

To the best of our knowledge, we are the first to explore the usage of LMs to **enhance common sense knowledge** in a KG. We asses the validity of our method by producing word embeddings from our improved KG and evaluating them on several semantic similarity tasks, which stem from the primary ConceptNet papers [28] and [29] and hence provide direct comparison to the initial results. Additionally, we assume that semantic similarity of word representations is a good indicator of improvement, as it has been shown that it influences other tasks, e.g. named entity disambiguation [9].

A related setting for improving existing KGs is **graph completion**, or link prediction, in which the goal is to automatically create edges between existing and new nodes of a given KG [3,26]. There exist approaches which successfully

utilize BERT for the task of graph completion on ConceptNet [20] as well as WordNet and Freebase [33]. While this is in principle similar to our method, we set ourselves a different task: Where KG completion allows for enriching existing KGs with new nodes and relations, this is only effective when the base graph used for training the completion approaches already contains facts of high quality. Our aim, however, is to further improve the information contained in existing edges in the base KG.

4 Methodology

To transfer knowledge from a LM to a KG, we propose REWEIGHT, which consists of a *sentence construction* and a *weight generation* step.

4.1 Sentence Construction

Relation-to-Sentence Mapping We want to evaluate the information contained in a KG by means of an LM. Thus, we first transform every edge $e \in E$ from the graph into a natural language sentence. We manually define a set of rules, which map the relations between graph nodes to sentences. For example, a *"DefinedAs"* relation between nodes **A** and **B** in ConceptNet is transformed to the sentence "**A** *is defined as* **B**". For most relations, such a simple transcription of the relation is sufficient. For some relation types, however, we observe that the LM reacts poorly to the direct transcription. We assume that this is due to the sparsity of sentences explicitly mentioning words such as *"antonym"* in their training data. Hence, we manually create transcriptions to better reflect natural language. Similarly, we observe that sentences are rated as more likely if all concepts are preceded by the indefinite article "a". For the full mapping we employ for the ConceptNet KG, we refer to the Supplementary Material.

Sentence Correction Due to the simple transformation rules, sentences generated in the previous step may not always be grammatically correct. LMs like BERT, however, have been shown to encode both syntactic and semantic information [17]. Since we aim to use the LM for assessing the semantic content of the sentence, we would like to discard any influence of syntax. To achieve this, we employ an additional LM trained to improve the grammatical quality of sentences. We feed our rule-generated sentences into the grammar correction model, obtaining semantically equivalent sentences with improved syntax. Further details on the specific implementation used in this work is given in Section 5.

4.2 Weight Generation

After constructing a sentence for every edge $e \in E$ in the KG, we need to measure the sentences' meaningfulness. We use the perplexity of a pre-trained LM to assess whether the sentence and thus the relation contains a probable fact, assuming that a well-trained LM will assign a high perplexity to sentences describing questionable or uncommon relations.

Perplexity in bi-directional LMs Since [7] has shown that the common definition of perplexity is not applicable to bi-directional LMs such as BERT, we use their approximation to compute a score for each edge $e \in E$ in the graph: the perplexity $pp(e)$ for each sentence $\mathbf{s}_e = \langle w_{e,1}, \ldots, w_{e,n_e} \rangle$ can be approximated as

$$pp(e) = \exp\left(-\frac{1}{n_e} \sum_{j=1}^{n_e} \log p(w_{e,j} | \langle w_{e,k} : k \neq j \rangle) \right), \tag{2}$$

where $\langle w_{e,k} : k \neq j \rangle$ denotes the context of $w_{e,j}$ in sentence $\mathbf{s}_e$.

We will now introduce two approaches to transform the resulting perplexities from their original range of $[1, +\infty)$ to the range of original KG weights $[0, \gamma_{\max}]$, where $\gamma_{\max}$ is the maximum weight of the original KG and $\tilde{\gamma}$ its median.

REWEIGHT$_{light}$ For a light variant of the REWEIGHT scheme, we obtain the weight for an edge $e \in E$ in the KG through transforming the perplexities obtained by the LM into the range of the original KG edge weights through

$$\beta_{\mathrm{RWL}}(e) \coloneqq \frac{\gamma_{\max}}{pp(e)} \tag{3}$$

where $\gamma_{\max}$ denotes the maximum weight in the original KG as noted above.

REWEIGHT$_{mod}$ Furthermore, we propose an adaptive version of REWEIGHT, making use of a parameter pp_b to separate sentences into reasonable and unreasonable ones. Let $pp_{\max}$ be the maximum perplexity obtained by feeding all edges $e \in E$ through the above pipeline, $pp_{\max} = \max_{e \in E} pp(e)$, and pp_b a parameter, which can be chosen freely. Then we define our REWEIGHT$_{\mathrm{mod}}$ weights as

$$\beta_{\mathrm{RWM}}(e) \coloneqq \begin{cases} r(e), \text{ if } pp(e) < pp_b \\ u(e), \text{ otherwise} \end{cases} \tag{4}$$

with $r : [0, pp_b[\to]\tilde{\gamma}, \gamma_{\max}]$ producing new edge weights for reasonable sentences and $u : [pp_b, \infty[\to [0, \tilde{\gamma}]$ for uncommon relations. For both functions we will utilize an inverted perplexity, which limits the influence of very high values

$$pp^{\mathrm{inv}}(e) = \log_{10}\left(\frac{pp_{\max}}{pp(e)} \right). \tag{5}$$

We feed this inverted perplexity into a min-max-scaling scheme separately for $r(e)$ and $u(e)$, to distribute scores evenly for both partitions. Thus, we set

$$u(e) \coloneqq \frac{\tilde{\gamma} \cdot pp^{\mathrm{inv}}(e)}{pp_b^{\mathrm{inv}}}, \tag{6}$$

where $pp_b^{\mathrm{inv}} \coloneqq \log_{10}(\frac{pp_{\max}}{pp_b})$. Note that while the lower bound for our new edge weights could be set to any value, we choose 0 as the retrofitting method used in our experiments treats relations with edge weight 0 as non-existent, allowing

our rating scheme to effectively remove edges with very low reasonability scores. For reasonable sentences we use a similar min-max-scaling,

$$r(e) := \tilde{\gamma} + (\gamma_{\max} - \tilde{\gamma}) \cdot \frac{pp^{\text{inv}}(e) - \max_{e \in E}(pp^{\text{inv}}(e)) + log_{10}(pp_b)}{log_{10}(pp_b)}. \quad (7)$$

The resulting weights are then used to replace the edge weights of the KG, yielding a linear mapping with control over the reasonability border pp_b.

5 Experimental Setting

We use the following setup throughout all our experiments: We apply REWEIGHT to ConceptNet to derive a new weighting for all relations between English concepts, leaving all relations that involve at least one non-English concept unchanged. We additionally follow [28] by removing uncommon concepts with less than three neighbors, and concepts that are not in any way connected to those in the vocabulary of the word embeddings used during Retrofitting.

For the sentence correction step of our approach, we use the BERT-based language correction model PIE [1], a current model performing strongly on the CoNLL-2014 shared task on grammatical error correction [21]. We additionally chose the PIE grammar checker since it is specifically tuned to improve sentences towards what BERT would consider to be syntactically correct, fitting the aim of mitigating the syntactic signal of sentences. The model takes as input sentences for correction and iteratively improves their grammar. In our experiments we use three correction iterations over each sentence, after which no further changes to the sentences were observed.

To obtain the perplexity of each sentence, we then use an openly available BERT LM adaptation[2] that calculates sentence perplexities based on the perplexity approximation for bidirectional LMs described in Section 4.2. We specifically chose a BERT model, since next to its state-of-the-art performance on many natural language tasks, BERT has also been shown to contain a certain amount of world knowledge [31]. The BERT model used in our experiments is the BERT-large (whole word masked) model.

5.1 Evaluation Task

As highlighted in Section 2.2, we extrinsically evaluate the weights determined by REWEIGHT by deriving word embeddings from our modified ConceptNet using the expanded retrofitting algorithm described in Section 2.3. With retrofitting using all weights in the KG to transform the embedding space, we use the relatedness scores between many words in this space to measure how well the information in the KG enriches the embedding space. Comparing the results to embeddings obtained through the base graph (with identical structure) then factors out the impact of the general graph structure and yields an automatic

[2] https://github.com/xu-song/bert-as-language-model

evaluation scheme that highlights the quality of the edge weights in the entire graph. In order to enable a direct comparison of our modified weights to the original ones, we use the same setup as [28], the only difference being that we apply Retrofitting to our REWEIGHTed ConceptNet instead of the original. Since the resulting embeddings are a combination of ConceptNet NumberBatch and BERT, we name them *ConceptNet NumBERTbatch*.

Where not otherwise noted, we employ $\text{REWEIGHT}_{\text{mod}}$ with the following parameters: For ConceptNet we find that the median and maximum of original weights is $\tilde{\gamma} = 1$ and $\gamma_{\max} = 50$, respectively (cf. Section 2). After inspecting perplexities of generated sentences, we set the perplexity border value to $pp_b = 100$, which will be validated later in Section 6.2. We also follow [28] in assessing the quality of the embeddings by calculating the cosine similarity of words in the embedding space and comparing the results to human intuition through Spearman correlation for several word similarity and relatedness datasets.

5.2 Evaluation Datasets

We use the following established semantic relatedness datasets for evaluation: **MEN3000** [5] consists of 3000 word pairs and their similarity scores collected through crowdsourcing. Scores of the dataset are distributed between 0-50. Additionally, this dataset contains a development- and test-split of 2000 and 1000 word pairs respectively. We report our main results on the full 3000 word pairs, while using only the development set for some additional experiments. **Rare Words** (RW) [19] contains 2034 word pairs of words with low occurrence counts in a Wikipedia text corpus. Each word pair is assigned a similarity score by ten human annotators. The pair scores are defined between 0 and 10. For ablation studies, we employ a development set of 1356 word pairs (RW_{dev}). **MTurk-771** [13] contains 771 word pairs with their relatedness scores. The dataset aims to cover different types of relatedness (e.g. synonymy, meronymy, etc.). The scores are defined between 1-5. **WS353** [12] consists of 353 word pairs with human relatedness scores distributed between 0 and 10. **Semeval17-2a** [6] consists of 500 word pairs, with scores ranging from 0 to 4. The pairs contain named entities and multi-word expressions. The dataset was designed to cover different domains (e.g. Biology, Education, etc.). **SimLex999** [16] contains 999 word pairs. Human annotators were instructed to differentiate between similarity and relatedness, rating word pairs purely on their similarity. SimLex999 has been created to evaluate how well models asses similarity of word pairs rather than relatedness.

On some of the described, widely used datasets, small sample size does not allow for showing significance when comparing to an already strong baseline. Hence, we follow [28] by calculating results on many different datasets, showing significance on the larger and the overall trend on all datasets.

5.3 Baselines

We evaluate our approach against two baselines. As a first baseline, we join several pretrained word embeddings (word2vec, GloVe, FastText) through trun-

Table 1. Spearman correlation of embeddings generated through retrofitting with different KGs on multiple word similarity datasets. Significant difference to NB_{orig}through Fischer's z-transformation with $^{\dagger}p < 0.01$, $^{\S}p < 0.05$.

Group	Embedding	MEN 3000	RW	MTurk	WS353	SemEval	SimLex	Average
Baseline	Joint	0.852	0.565	0.782	0.803	0.645	0.519	0.694
	NB_{orig}	0.872	0.630	0.822	0.833	0.779	**0.633**	0.762
Ours	$NBERT_{light}$	0.877	†**0.663**	0.827	0.840	0.783	**0.633**	**0.770**
	NBERT	†**0.881**	§0.651	**0.828**	**0.845**	0.780	0.618	0.767
Other	$NBERT_{base}$	0.873	0.644	0.822	0.833	**0.784**	0.625	0.764
LMs	w/o grammar	§0.879	§0.650	**0.828**	0.843	0.774	0.624	0.766

cated SVD [28], which achieves stronger performance than the base embeddings individually. The second baseline is provided by the ConceptNet NumberBatch embeddings [28], which are constructed from ConceptNet in the same procedure we use for our NumBERTbatch embeddings, joining several pretrained word embeddings (word2vec, GloVe, FastText) and Retrofitting them to ConceptNet.

6 Results

In this section, we report our main experimental results in comparison to the two baselines, as well as an ablation study evaluating different variations of our proposed measure REWEIGHT. Table 1 contains all main results from this section, which we will address in the course of the section.

6.1 NumBERTbatch Embeddings

We compare the NumBERTbatch embeddings resulting from our REWEIGHTed KG to the performance of the original NumberBatch and the joint word embeddings without retrofitting. We additionally evaluate $REWEIGHT_{light}$, generating $NumBERTbatch_{light}$ embeddings. The results for these settings are shown in the first two blocks of Table 1. With both weighting schemes, we obtain consistent improvements over the already strong original NumberBatch on multiple datasets, especially showing significant improvements on the large MEN3000 and Rare Words (RW) datasets. This suggests that our method is capable of improving the knowledge aggregated in a KG. It is interesting to note that, while both schemes improve the overall performance over the baselines, they seem to present different focuses, with one improving more strongly on MEN3000 and the other on Rare Words. Another interesting observation is that on SimLex, a dataset tailored to semantic similarity (as opposed to relatedness), NumBERTbatch performs worse than the original. This is not unexpected, since we do not enforce a distinction between relatedness and similarity. It would be an interesting task for future work to evaluate whether the performance on SimLex can be improved by focusing on relations describing similarity, such as "SimilarTo".

6.2 Ablation Study

In order to gain further insights into the performance of our approach, we conduct a deeper investigation, analyzing the influence of different hyper-parameter choices and model variations on the performance of our method.

Varying the Perplexity Border pp_b First, we investigate the influence of the perplexity border pp_b for $\text{REWEIGHT}_{\text{mod}}$ (the maximum perplexity of a sentence that is considered to be "reasonable"), varying pp_b in the range from 50 to 1000. We find that most values for pp_b do not have a large influence on the results and refer to the supplementary material for details. Choosing higher values of pp_b leads to slight loss of performance, while still consistently remaining above the original graph. This matches our intuition, since for very high values of pp_b even sentences that the LM deems improbable receive somewhat high scores. Thus the separation between more and less reasonable sentences is weakened.

Clipping Outliers As a next step, we test the impact that possible outliers (i.e., sentences with extremely high perplexity) may have on our weighting scheme. For this, we define an upper bound pp_c for the perplexity of generated sentences, setting $pp_i = \min(pp_i, pp_c)$ for all sentences. Results of applying $\text{REWEIGHT}_{\text{mod}}$ to ConceptNet with different upper bounds show no statistically significant changes compared to not using any upper bound. For details, we again refer to the supplementary material.

Trimming Extreme Weights To investigate how much information is contained within the edges that received particularly low (high) weights during our reweighting, we experiment with setting all weights below (above) a given threshold to 0, thus removing the information of these edges during retrofitting. We expect removing edges with low weights to only have a small influence on the results (since these are not particularly important), while removing highly weighted edges having a more serious impact. The results of the experiment support our hypothesis: Removing edges with high weights has much more impact on the overall performance than removing edges with small weights. Details are provided in the supplementary material.

Changing the LM In order to investigate the influence of the LM used during the REWEIGHT process, we experiment with using the smaller *BERT-base* model instead of *BERT-large*. With the *BERT-base* model containing $110M$ parameters, significantly fewer than the $340M$ parameters of *BERT-large*, we expect it to encode less knowledge, leading to a lower performance when used with $\text{REWEIGHT}_{\text{mod}}$. The results in Table 1 under $NBERT_{base}$ show a considerable loss of performance with the use of the smaller LM, with performance on most datasets being only slightly above NB_{orig}, which uses the original ConceptNet.

Removing Grammar Correction As a final experiment, we want to show that the grammar correction step is necessary for our model. We therefore apply the

Table 2. Spearman correlation of embeddings generated through retrofitting. Different KGs used for retrofitting, as well as KGs with shuffled and rescaled edge weights.

Group	Embedding	MEN 3000	RW	MTurk	WS353	SemEval	SimLex	Average
Scaling	lim=10	0.877	0.612	0.824	0.839	0.779	0.617	0.758
	lim=15	0.875	0.617	0.823	0.839	0.780	0.622	0.759
Shuffling	R-NB$_{orig}$	0.871	0.641	0.821	0.832	0.778	0.625	0.761
	R-EN NB$_{orig}$	0.870	0.643	0.821	0.831	0.780	0.631	0.763
	R-NBERT	0.867	0.628	0.814	0.829	0.768	0.608	0.752
	R-EN NBERT	0.871	0.617	0.816	0.833	0.774	0.605	0.753
Other KGs	WebChild	0.850	0.507	0.781	0.803	0.674	0.529	0.691
	WCBERT	0.847	0.514	0.770	0.805	0.678	0.510	0.687
	Yago	0.835	0.391	0.739	0.792	0.670	0.550	0.663
	YagoBERT	0.829	0.393	0.734	0.783	0.665	0.542	0.658

REWEIGHT$_{\text{mod}}$ process without the PIE grammar checker. Results in the final column of Table 1 show a slight decrease in performance across all datasets. This suggests that the grammar correction step can indeed help to reduce the influence of *syntactical* signals on the performance, therefore increasing the weight of the *semantic* signals that we want to use for our REWEIGHTing process.

7 Analysis

In this section, we provide an extensive analysis of how our method influences the KG's weights. To this end, we verify that the improvements are not due to lucky rescaling or reshuffling of the original weights and provide insight into the weight changes from the original ConceptNet to our REWEIGHTed version.

7.1 Assessing added Information

This section aims at showing that the improvements from our REWEIGHTed graph are not only due to changing the underlying distribution of the weights in the graph, but that the LM actually adds useful information. To this end, we conduct two experiments: rescaling the weights of the original CN and reshuffling our modified weights.

Rescaling the Original Weights To make sure that our method's improvements are not simply due to amplifying the weights in the original graph, we experiment with manual rescaling. Specifically, the weights of all edges between English concepts are scaled linearly between 0 and 50 through min-max scaling as follows:

$$\gamma \rightarrow \begin{cases} [0, 45] & \text{if } \gamma \leq lim \\ (45, 50] & \text{else} \end{cases} \tag{8}$$

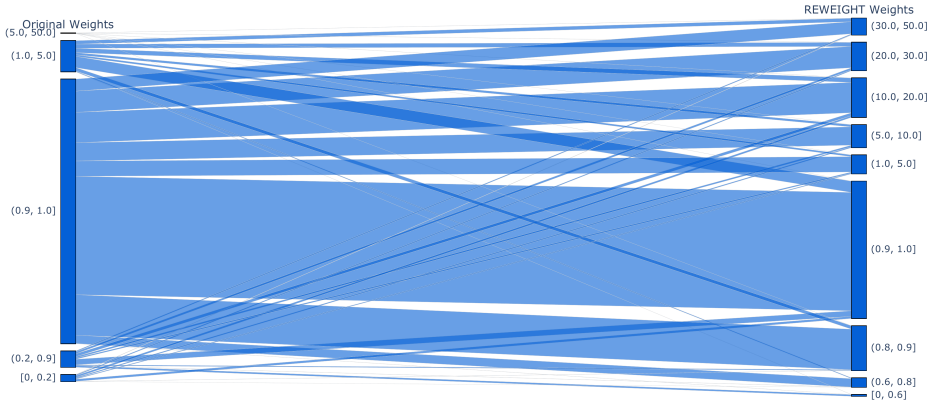

Fig. 2. ConceptNet weights before and after REWEIGHTing

where we try different values for lim in order to specifically highlight high scoring edges in the base KG. Results for the values $lim \in \{10, 15\}$ are reported in Table 2. While an increase in performance in comparison to the base graph is observable in most datasets for $lim = 10$, the results do not meet our weighted NumBERTbatch embeddings. Reasons for the improvement are further discussed when we investigate the changes made to the KG by our approach in Section 7.2.

Reshuffling the Modified Weights This experiment serves as proof that our method does not simply change the distribution of the weights in a favorable way, without actually adding information from the LM to the graph. We take our improved KG and randomly shuffle the weights of either (a) all relations or (b) only English relations. If our method just luckily changed the distribution, this reshuffling would still lead to better results than the original KG weights. The resulting correlation coefficients after retrofitting can be observed in block *Shuffling* in Table 2. It can be seen that the randomized distribution of weights leads to lower performance on all datasets. Shuffling only the English part of the graph appears to retain a small amount of information from the remaining languages, yielding a slightly higher performance than shuffling the full graph. The strong decreases in performance indicate that the information contained in the edge weights of the graph are important for the task of semantic relatedness.

7.2 Changes to the KG

We further investigate the changes that REWEIGHT made to the KG. Figure 2 shows the transition of edge weights during the application of $\text{REWEIGHT}_{\text{mod}}$. We observe that $\text{REWEIGHT}_{\text{mod}}$ redistributes its weights more broadly over the value range, with considerably more high and low weights in comparison to the base graph. This might explain why additionally increasing the weight of important edges in the base graph leads to the improvements in correlation observed in Section 7.1. The major changes our approach appears to make to

the weight distribution of the KG are increasing the weight of many relations in the interval $(0.9, 1]$, which shows an ability to highlight specific reasonable relations. REWEIGHT$_{\text{mod}}$ also slightly decreases the weight of many edges in the range $(1, 5]$, bringing e.g. "**mathematical** *SimilarTo* **unquestionable**" from 2.0 to 0.88. Additionally, for the edges that were rated very low in the original ConceptNet, we observe many slight weight increases, as well as many strong increases, with e.g. "**mathematics** *RelatedTo* **geometry**" being changed from very low (0.1) to very high (34.8) weights by REWEIGHT$_{\text{mod}}$.

On the other hand, we observe difficulties of the approach on relations that include highly specific concepts such as "anthrax". Since these concepts do not appear in the vocabulary of the BERT LM, they are assessed on character- and substring-level which causes higher perplexity scores than known concepts. Due to this, highly specific relations such as "**anthrax** *IsA* **disease**" are changed from high (2.8) to low scores (0.9) in spite of containing reasonable information. This suggests possible further improvements of the approach through assessing out-of-vocabulary concepts separately, which we leave as future work.

7.3 Choice of KG

REWEIGHT can be applied to improve the weights in any KG. Our previous experiments have focused on ConceptNet, one of the most prevalent common sense KGs being employed on a variety of application scenarios [27,28,37]. In this section, we evaluate the suitability of REWEIGHT to derive new weights for two other well-known KGs, YAGO and WebChild. As a preprocessing step, we aggregate all scores for relations that occur several times between the same concepts, creating a unique relation between the concepts with summed score. We then use the KGs with original weights for retrofitting, reporting our results in Table 2. We find that retrofitting with either WebChild or YAGO does not achieve an improvement over the original joint embeddings (*Joint* in Table 1). We evaluate both KGs further, but find that neither weighting their edges with REWEIGHT, nor any other modifications we tried (i.e., manually scaling edge weights, removing entire subgraphs, and removing uncommon concepts) manage to improve on our baselines.

We therefore conclude that the application of retrofitting to WebChild and YAGO does improve word embeddings on semantic relatedness. While this may be caused by the Retrofitting task itself, we also make the following observations concerning the structure of the graphs: WebChild strongly represents structured knowledge about activities (e.g. *drive a car*) and object properties (e.g. *hasSize*), while relations between concepts are only represented through part-whole relations (e.g. *isMemberOf*, *partOf*) and comparison relations (e.g. *largerThan*). The YAGO Taxonomy builds hierarchical information of *isA* relations between concepts. Although these relations contain important knowledge for word relatedness, the relations in both KGs are focused on hierarchical connections between concepts, which appear to carry less information for the semantic relatedness datasets compared to the rich relations in ConceptNet. Since our method only improves the weights of the edges and is not capable of changing the structure of

the graph, it may thus be unsuitable to improve the performance of WebChild and YAGO for our semantic similarity tasks.

8 Conclusion

In this paper, we have proposed REWEIGHT, a pipeline for enriching structured common sense KGs with information contained in LMs through converting KG relations to natural language sentences and rating their reasonability. For this, we introduced a mapping of KG edges to natural sentences, and assessed the semantic reasonability of the sentences by calculating their perplexity with an LM. We then introduced a scheme for transforming the resulting perplexities to edge weights in the range of the original KG weights, yielding an enriched KG containing additional information through knowledge from an LM.

We applied REWEIGHT on the relatedness-oriented common sense KG ConceptNet, investigating whether the world knowledge contained in the BERT LM can be used to improve the information contained in the KG for the task of semantic relatedness. To evaluate the performance of the enriched KG, we employed the retrofitting setting of [28], using the KG as additional information to improve existing word embeddings and evaluating the resulting embeddings on multiple semantic relatedness datasets.

Our results show that the BERT LM can be used to further improve the already strongly performing ConceptNet NumberBatch across all evaluated relatedness datasets. In an extended investigation we found that BERT managed to assess the semantic reasonability of ConceptNet relations well, giving high weights to edges with essential information for use in improving existing word embeddings.

Overall, our results uncover promising opportunities for improving existing KGs with unstructured information contained in LMs. Through representing edges in KGs as natural sentences, many established techniques in Natural Language Processing (NLP) may be used to automatically improve the information contained in KGs. Additionally, it may be possible to add information from specialized LMs into a KG, which in turn can be used as a source of background knowledge for domain dependent tasks [15]. One further opportunity for future work may be the careful construction of sentences from edges, aiming to eliminate any biases the employed NLP approaches may have towards sentence construction, i.e. through employing different and varying sentence templates.

References

1. Awasthi, A., Sarawagi, S., Goyal, R., Ghosh, S., Piratla, V.: Parallel iterative edit models for local sequence transduction. In: 2019 EMNLP-IJCNLP (2019)
2. Bollegala, D., Alsuhaibani, M., Maehara, T., Kawarabayashi, K.i.: Joint word representation learning using a corpus and a semantic lexicon. In: Thirtieth AAAI Conference on Artificial Intelligence (2016)

3. Bosselut, A., Rashkin, H., Sap, M., Malaviya, C., Çelikyilmaz, A., Choi, Y.: COMET: Commonsense transformers for automatic knowledge graph construction. In: ACL (2019)
4. Bouraoui, Z., Camacho-Collados, J., Schockaert, S.: Inducing relational knowledge from bert. In: AAAI 2019 (2019)
5. Bruni, E., Tran, N.K., Baroni, M.: Multimodal distributional semantics. In: Journal of artificial intelligence research. vol. 49, pp. 1–47 (2014)
6. Camacho-Collados, J., Pilehvar, M.T., Collier, N., Navigli, R.: Semeval-2017 task 2: Multilingual and cross-lingual semantic word similarity. In: 11th International Workshop on Semantic Evaluation (SemEval-2017). pp. 15–26 (2017)
7. Chen, X., Liu, X., Ragni, A., Wang, Y., Gales, M.J.: Future word contexts in neural network language models. In: 2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU). pp. 97–103. IEEE (2017)
8. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: 2019 Conf. of the North American Chapter of the Association for Computational Linguistics (2018)
9. Eshel, Y., Cohen, N., Radinsky, K., Markovitch, S., Yamada, I., Levy, O.: Named entity disambiguation for noisy text. In: Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017). pp. 58–68 (2017)
10. Faruqui, M., Dodge, J., Jauhar, S.K., Dyer, C., Hovy, E., Smith, N.A.: Retrofitting word vectors to semantic lexicons. In: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 1606–1615 (2015)
11. Fensel, D., Şimşek, U., Angele, K., Huaman, E., Kärle, E., Panasiuk, O., Toma, I., Umbrich, J., Wahler, A.: Knowledge graphs: Methodology, tools and selected use cases. Springer Nature (2020)
12. Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., Ruppin, E.: Placing search in context: The concept revisited. In: Proceedings of the 10th international conference on World Wide Web. pp. 406–414 (2001)
13. Halawi, G., Dror, G., Gabrilovich, E., Koren, Y.: Large-scale learning of word relatedness with constraints. In: Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 1406–1414 (2012)
14. Havasi, C., Speer, R., Alonso, J.: Conceptnet 3: a flexible, multilingual semantic network for common sense knowledge. In: Recent advances in natural language processing. pp. 27–29. Citeseer (2007)
15. Hettinger, L., Dallmann, A., Zehe, A., Niebler, T., Hotho, A.: Claire at semeval-2018 task 7: Classification of relations using embeddings. In: 12th International Workshop on Semantic Evaluation (2018)
16. Hill, F., Reichart, R., Korhonen, A.: Simlex-999: Evaluating semantic models with (genuine) similarity estimation. Computational Linguistics **41**(4), 665–695 (2015)
17. Jawahar, G., Sagot, B., Seddah, D.: What does bert learn about the structure of language? In: Association for Computational Linguistics (2019)
18. Lengerich, B., Maas, A., Potts, C.: Retrofitting distributional embeddings to knowledge graphs with functional relations. In: Proceedings of the 27th International Conference on Computational Linguistics. pp. 2423–2436 (2018)
19. Luong, M.T., Socher, R., Manning, C.D.: Better word representations with recursive neural networks for morphology. In: Proceedings of the Seventeenth Conference on Computational Natural Language Learning. pp. 104–113 (2013)
20. Malaviya, C., Bhagavatula, C., Bosselut, A., Choi, Y.: Commonsense knowledge base completion with structural and semantic context. In: AAAI (2020)

21. Ng, H.T., Wu, S.M., Briscoe, T., Hadiwinoto, C., Susanto, R.H., Bryant, C.: The conll-2014 shared task on grammatical error correction. In: 18th Conference on Computational Natural Language Learning: Shared Task. pp. 1–14 (2014)
22. Niebler, T., Becker, M., Pölitz, C., Hotho, A.: Learning semantic relatedness from human feedback using relative relatedness learning. In: 16th International Semantic Web Conference (ISWC) (2017)
23. Petroni, F., Rocktäschel, T., Riedel, S., Lewis, P., Bakhtin, A., Wu, Y., Miller, A.: Language models as knowledge bases? In: 2019 Conf on Empirical Methods in Natural Language Processing (EMNLP). pp. 2463–2473 (01 2019)
24. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I.: Language models are unsupervised multitask learners. OpenAI Blog **1**(8), 9 (2019)
25. Rebele, T., Suchanek, F., Hoffart, J., Biega, J., Kuzey, E., Weikum, G.: Yago: A multilingual knowledge base from wikipedia, wordnet, and geonames. In: International Semantic Web Conference. pp. 177–185. Springer (2016)
26. Sadeghi, A., Graux, D., Shariat Yazdi, H., Lehmann, J.: Mde: multiple distance embeddings for link prediction in knowledge graphs. In: ECAI (2020)
27. Sharifirad, S., Jafarpour, B., Matwin, S.: Boosting text classification performance on sexist tweets by text augmentation and text generation using a combination of knowledge graphs. In: 2nd workshop on abusive language online (ALW2) (2018)
28. Speer, R., Chin, J., Havasi, C.: Conceptnet 5.5: An open multilingual graph of general knowledge. In: Thirty-First AAAI Conference on Artificial Intelligence (2017)
29. Speer, R., Lowry-Duda, J.: Conceptnet at semeval-2017 task 2: Extending word embeddings with multilingual relational knowledge. In: Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017). pp. 85–89 (2017)
30. Tandon, N., De Melo, G., Weikum, G.: Webchild 2.0: Fine-grained commonsense knowledge distillation. In: ACL 2017, System Demonstrations. pp. 115–120 (2017)
31. Xiong, W., Du, J., Wang, W.Y., Stoyanov, V.: Pretrained encyclopedia: Weakly supervised knowledge-pretrained language model. In: International Conference on Learning Representations (2020)
32. Xu, C., Bai, Y., Bian, J., Gao, B., Wang, G., Liu, X., Liu, T.Y.: Rc-net: A general framework for incorporating knowledge into word representations. In: 23rd ACM int conf on information and knowledge management. pp. 1219–1228 (2014)
33. Yao, L., Mao, C., Luo, Y.: Kg-bert: Bert for knowledge graph completion. In: arXiv preprint arXiv:1909.03193 (2019)
34. Yu, M., Dredze, M.: Improving lexical embeddings with semantic knowledge. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). pp. 545–550 (2014)
35. Zellers, R., Bisk, Y., Farhadi, A., Choi, Y.: From recognition to cognition: Visual commonsense reasoning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6720–6731 (2019)
36. Zhang, Z., Han, X., Liu, Z., Jiang, X., Sun, M., Liu, Q.: ERNIE: Enhanced language representation with informative entities. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 1441–1451 (2019)
37. Zhong, W., Tang, D., Duan, N., Zhou, M., Wang, J., Yin, J.: Improving question answering by commonsense-based pre-training. In: Natural Language Processing and Chinese Computing (2019)
38. Zhou, X., Zhang, Y., Cui, L., Huang, D.: Evaluating commonsense in pre-trained language models. AAAI (2020)