Capturing Concept Similarity with Knowledge Graphs

Filip Ilievski *, Kartik Shenoy, Nicholas Klein, Hans Chalupsky and Pedro Szekely
Information Sciences Institute, University of Southern California, CA, USA
E-mail: {ilievski,kshenoy,nmklein,hans,pszekely}@isi.edu

Abstract. Robust estimation of concept similarity is crucial for a range of AI applications, like deduplication, recommendation, and entity linking. Rich and diverse knowledge in large knowledge graphs like Wikidata can be exploited for this purpose. In this paper, we study a wide range of representative similarity methods for Wikidata, organized into three categories, and leverage additional knowledge as a self-supervision signal through retrofitting. We measure the impact of retrofitting with subsets from Wikidata and ProBase, scored based on language models. Experiments on three benchmarks reveal that pairing language models with rich information performs best, whereas the impact of retrofitting is most positive on methods that originally do not consider comprehensive information. The performance of retrofitting depends on the source of knowledge and the edge weighting function. Meanwhile, creating evaluation benchmarks for contextual similarity in Wikidata remains a key challenge.

Keywords: Similarity, Wikidata, Retrofitting, Knowledge graphs, Embeddings

1. Introduction

Robust and scalable metrics of similarity between two concepts is at the core of a wide range of applications. Deduplication requires detection of entities that are similar enough and thus mutually redundant. Recommendation scenarios rely on similarity metrics in order to suggest further entities that are either similar or contextually related to a given entity. Entities mentioned in the same column of a table are typically coherent, making similarity a key ingredient in tasks like table typing and linking. To support these applications, we need methods to automatically infer whether two arbitrary concepts are identical, dissimilar, or nearly identical [1].

The rich and diverse knowledge available in modern knowledge graphs (KGs) with billions of statements, like Wikidata [2], can be used to develop robust similarity methods. Yet, surprisingly little effort has been devoted to understanding and devising metrics of similarity for KGs like Wikidata. The task of concept word similarity has been very popular [3–6]. Early work generally relies on taxonomy-based methods that leverage the distance between two words in a taxonomy hierarchy [7]. More recently, pre-trained word embeddings have been shown to natively capture word similarity at scale [8–10]. Word embeddings may benefit from retrofitting to lexical resources like WordNet [11]. It is unclear how to best estimate similarity of concepts described in KGs. Besides language models and taxonomy-based metrics, we can leverage graph embeddings, like TransE [12] and ComplEx [13], which organize nodes in a geometric space according to their structural links to other nodes. Random walk methods, such as node2vec variants [14, 15], leverage the generalizability of language modeling, applying it to graph nodes instead of words. Furthermore, the embeddings created by language models (LMs) or KGs can be retrofitted based on background knowledge, coming from the target graph or additional resources.

*Corresponding author. Email: ilievski@isi.edu.

In this paper, we study the effect of a wide range of similarity methods based on taxonomical structure, language models, and knowledge graph embeddings. While the relevance of these methods can be loosely attributed to similarity, this is the first head-to-head investigation of their application to a knowledge graph like Wikidata. We compare the methods in an unsupervised form on three benchmarks. In addition, we measure the impact of retrofitting based on various knowledge subsets from Wikidata and ProBase, and different edge weighting functions applied to these subsets (e.g., based on BERT or taxonomy metrics). This study reveals that many of the existing similarity metrics can be adapted for estimating similarity on large-scale knowledge graphs like Wikidata. It shows that retrofitting to Wikidata is typically beneficial, but not to ProBase [16]. The paper points to open questions regarding meaningful evaluation of similarity between two Wikidata concepts, and the need for novel benchmarks that can also compute entity-to-entity similarity. Novel metrics, such as dimensional similarity metrics that generalize beyond the P279 taxonomy, are also critical future work directions.

2. Similarity of concepts

Similarity is a central theoretical construct in psychology, facilitating the transfer of a situation to an original training context [17, 18]. Tversky [19] poses that the *literal similarity* between two objects A and B is proportional to the intersection of their features and inversely proportional to the features that differ ($A - B$ and $B - A$). In other words, A and B are literally similar if their set of shared features is relatively larger than the non-shared ones. Here, features include both attributes and relational predicates. This differs from *analogy*, where only relational predicates are shared (e.g., atom - solar system), and from *mere appearance*, where only the attributes are shared and not the relationships (e.g., moon - coin) [20]. Comparison with no attribute nor relational overlap is called anomaly [20].

Gentner and Markman [21] argue that similarity is like analogy, in the sense that both rely on the alignment between the two compared objects/domains. When provided a similar pair, like hotel - motel, people align them based on the shared properties (e.g, used for accommodation), and are able to easily point out differences (e.g., hotels are in cities, motels are on the highway). The authors discuss that it makes no sense to talk about differences in the absence of a meaningful alignment (e.g., kitten - magazine). Which features should be/are considered when computing similarity? According to [22], the relative importance of a feature depends on the stimulus task and the context. This flexibility of the estimation of similarity led to criticism, which argues that the set of features that are being compared is seemingly arbitrary [23]. Yet, the relatively high inter-annotator agreement indicates systematicity in the human judgments of similarity [18]. In addition, some of the variation across subject can be explained with phenomena like selective learning [24], developmental changes [25], and knowledge and expertise [26]. Similarity judgments are also known to be impacted by the context of the task [27], i.e., the features activated depend on the object we compare against; as well as the direction of the comparison: people tend to rate the similarity of North Korea to China higher than the reverse [18].

We can distinguish three measures of similarity: indirect, direct, and theoretical [18]. An example for indirect similarity comparison is asking human participants to identify potentially confusable stimuli, such as judging whether an object has been observed before. Similarity can be measured directly, by rating the similarity of stimuli on a numeric scale. Theoretical similarity is observed as a component in human cognition, for instance, when participants categorize an item by comparing its fit in various categories.

In this paper, we consider the task of literal similarity between two concepts. Given two concept nodes, c_1 and c_2 in a KG G, a system is asked to provide a pairwise similarity score $sim(c_1, c_2)$. We consider similarity to be asymmetric, i.e., $sim(c_1, c_2) \neq sim(c_2, c_1)$. Following common practice in the concept and word similarity tasks, we assume that the similarity of two concepts can be measured on a continuous numeric scale. We compare the relative order of the machine similarity for a dataset against human judgments.

3. Framework for estimating similarity

The proposed framework is visually depicted in Figure 1. We use graph embedding and text embedding models, as well as ontology-based metrics, as initial similarity estimators. We also concatenate the embeddings in order to

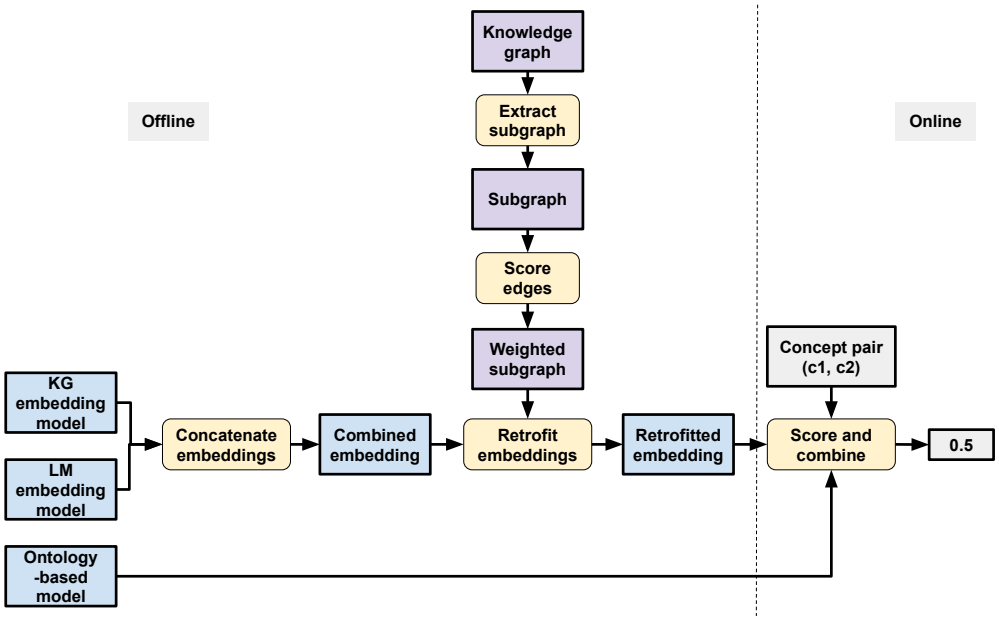

Fig. 1. Overview of our similarity estimation framework.

combine their scores. We use retrofitting to further tune the individual embedding models, through distant supervision over millions of weighted pairs extracted automatically from large-scale knowledge graphs. For a given concept pair, the similarity scores generated by the retrofitted embedding models can be combined with the scores by the ontology-based models. The individual components of our similarity framework are described in this section. The entire code of our framework and experiments can be found on GitHub: https://github.com/usc-isi-i2/wd-similarity.

3.1. Similarity models

We distinguish between similarity models based on KG embeddings, language models, and graph structure statistics. We employ representative methods from each category.

Graph embedding models We experiment with four KG embedding models, which can be divided into: translation based models (*TransE* [12] and *ComplEx* [13]) and random walk models (*DeepWalk* [28] and *S-DeepWalk* [15]).[1] For all models, we compute the cosine similarity between their embeddings for $c1$ and $c2$.

Language models We use Transformer LMs to represent the textual information node associated with a node in the graph. Similarity between two nodes is then measured through the cosine similarity between two LM embeddings. We experiment with four kinds of textual information: 1) *labels*, which consider only the English label; 2) *labels+desc*, which considers a concatenation between a node label and its description; 3) lexicalization, where we automatically generate a node description based on the properties: P31 (instance of), P279 (subclass of), P106 (occupation), P39 (position held), P1382 (partially coincident with), P373 (Commons Category), P452 (industry); and 4) abstract, which is based on the first sentences from entity abstracts in the DBpedia KG, mapped to Wikidata through their sitelinks.

Considering that KG and language model embeddings may provide complementary insights [29], we create two composite embeddings: 1) *Composite-6*, which combines all embedding models except Labels and Labels+desc, i.e., TransE, ComplEx, Deepwalk, S-Deepwalk, Abstract, Lexicalized; and 2) *Composite-2*, which combines the best KG and the best LM embeddings. We use tSNE to reduce the dimensions of the individual models to the same size, and compute the cosine similarity of the composite embeddings in the same way as with the individual models.

[1] S-DeepWalk is a variant that captures structural similarity. The input graph for S-Deepwalk is constructed by adding links between nodes that have neighbors from similar classes in the original KG, where class is defined by the instance-of (P31) property.

Ontology-aware models We use three structure-aware metrics. *Class similarity* computes the set of common IsA parents for two nodes. Each shared parent is weighted by its inverse document frequency (IDF), computed based on the number of instances that transitively belong to that parent class. *Jiang Conrath* [7] is an information theoretic node-based similarity measure that leverages the information content of the least common subsumer which is given by $jc(c1,c2) = 2 * logp(mss(c1,c2)) - (logp(c1) + logp(c2))$. Here, $mss(c1,c2)$ is the most specific subsumer, whereas $p(c)$ is the normalized probability that a particular concept c is of type C. *TopSim* computes top-similar regions for each node by enumerating nearest neighbors based on the KG ontology and from embeddings. The top-similar regions are ranked using an average of the above similarity computations. Once the top-similar regions are available, similarity is computed as a weighted average of the similarities between the two concepts and their top-5 similars.

3.2. Self-supervision knowledge

We tune the original embeddings by self-supervision to two KGs: Wikidata and ProBase. We derive three datasets from Wikidata's subclass-of (P279) ontology: 1) *WD-child-parent* consists of 304k edges whose nodes have non-empty and non-identical labels and descriptions, and both nodes have at least 10 P279 descendants;[2] 2) *WD-sibling* creates 785k sibling edges between two nodes in WD-child-parent which share the same immediate parent and have a non-identical description; 3) *WD-all*, which is a union of WD-child-parent and WD-sibling.[3]

We define three weighing methods for the generated pairs from these two datasets: (1) *constant* weighting value of 1; (2) *class* similarity between the two nodes (using the class metric described in the last section); and (3) *cosine* similarity between the concatenated labels and descriptions of the two nodes. For WD-child-parent pairs, we compute cosine similarity of the DistilRoberta embeddings [30] of their labels and descriptions, whereas for the WD-sibling pairs, we compute cosine similarity between the DistilRoberta embeddings of their sentences. Both sentences follow the template `{Label}, {Description}, is {Parent}`. We use the absolute cosine values as similarities, formally $sim(c1,c2) = |cosine(c1,c2)|$. We focus our experiments on cosine similarity as a weighting function, because we observed empirically that it consistently performs better or comparable to the other two weighting functions.

ProBase [31] has 33.37M child-parent pairs associated with a count of occurrences (relations) in textual sources. Based on an exact matching strategy, we aligned 1.6M (4.79 %) of these pairs to Qnodes in Wikidata. If multiple Qnodes are retrieved, we choose the Qnode with lowest numeric ID. The number of relations in this subset range from 1 to 35,167. As most nodes have a small number of relations, we scale the edge weight based on the following equation: $0.5 \times (1 + \frac{log(no\_of\_relations)}{log(max(no\_of\_relations))})$.

3.3. Retrofitting

We use the *retrofitting* technique proposed by Faruqui et al. [11], which iteratively updates node embeddings in order to bring them closer in accordance to their connections in an external dataset. Given a node with an embedding q_i, and n neighbours, where the j^{th} neighbour with similarity β_j has an embedding q_j, the retrofitted embedding $\hat{q}_i$ is computed as follows: $\hat{q}_i = \frac{q_i \times n^k + \sum_{j=1}^{n} q_j \times \beta_j}{n^k + \sum_{j=1}^{n} \beta_j}$. The parameter k dictates how much the original embedding will be changed based on its neighbors. Higher k values result in higher preservation of the original embedding. We adapt retrofitting to tune embeddings from KGs and LMs to Wikidata and ProBase.[4]

[2] If more than 500 children are present for a parent, we randomly sample 500 of them.

[3] The datasets are publicly available: https://drive.google.com/drive/folders/19poqPcXbLjSl5PbYogMVUWneiRb-81XO?usp=sharing

[4] We also experimented with cross-validated supervision models like SVM. As we did not observe improvements, we leave out these results from the paper.

Table 1
Statistics of our evaluation benchmarks.

Benchmark	#pairs	#unique concepts
WD-WordSim353	334	420
WD-RG65	34	31
WD-MC30	16	23

4. Experimental setup

Benchmarks and metrics We experiment with three benchmarks: 1) *WD-WordSim353* is derived based on the popular word similarity dataset WordSim-353 [32].[5] WordSim-353 contains 353 pairs of English words (334 without duplicate pairs), along with human similarity scores. As we observed that the original scores often conflate the notions of semantic similarity (*car-bike*) with relatedness (*car-wheel*), we re-annotated the dataset with numeric scores between 1 and 4: using 1 for (near-) identity; 2 for cases where two entities are partially substitutable, one is a slight specification of the other, or they are close siblings of the same category; 3 for pairs that are related by one of the following relations: distant inheritance, location, utility/capability, part-whole, antonymy, or domain; and 4 for unrelated pairs. Five researchers participated in this annotation, after which we produced new scores by averaging the five annotations. 2) *WD-RG65* is a benchmark which is based on the DBpedia disambiguation [33] of the RG-65 benchmark [34], which consists of 34 word pairs featuring 32 unique concepts. 3) *WD-MC30* is a benchmark which is based on the DBpedia disambiguation [33] of the MC-30 benchmark [35], which consists of 16 word pairs featuring 23 unique concepts. The benchmarks WD-MC30 and WD-RG65 were derived from their DBpedia version by using sitelinks data. Statistics about the three benchmarks can be found in Table 1. We evaluate using three metrics: Kendall-Tau (KT), Spearman rank (SR), and Root Mean Square Error (RMSE). We make the resulting benchmarks available for future evaluations.[6]

Implementation details For abstract, labels, label+description, and abstract we use DistilRoberta,[7] whereas for lexicalization we use BERT-base. For the class similarity model, IsA relations are computed as a transitive closure over both the subclass-of (P279) and the instance-of (P31) relations. For Jiang Conrath, the instance counts are computed using the properties: P31 (instance of), P39 (position held), P106 (occupation) and transitive P279 (subclass of). The reason for using positions and occupations in addition to P31 is that more descriptive classes (e.g., actor) are usually not linked via P31 which generally only points to Q5 (human) in those cases. We use Wikidata's dump from February 15th, 2021, and the current version of ProBase.[8] We experiment with three values for the retrofitting variable k: 0.5, 1, and 2. We generally observe best results with $k = 2$ and 2 iterations of retrofitting, and we present results for this configuration. The dimensions of the different embeddings are as follows: Lexicalized - 1024, ComplEx - 100, TransE - 100, Abstract - 768, Labels - 768, Labels+desc - 768, Deepwalk - 200, S-Deepwalk - 200.

We use the KGTK [36] toolkit to lexicalize a node, subset the graphs, and create various graph and language model-based embeddings. We use scikit-learn for supervised learning. We use KGTK's similarity API to obtain scores for the metrics Class, Jiang Conrath, and TopSim.[9]

5. Results

5.1. How well do different algorithms and combinations capture semantic similarity?

As can be seen from table 2, amongst our taxonomy-based methods, TopSim yields the highest correlation with the human-annotated similarity scores. This result indicates that taxonomical information alone does not suffice for

[5] http://www.gabrilovich.com/resources/data/wordsim353/wordsim353.html
[6] https://drive.google.com/drive/folders/1_u2MqXzBiUSMPjWH5GRmp7RP2FUYYxf1?usp=sharing
[7] https://huggingface.co/sentence-transformers/all-distilroberta-v1
[8] https://concept.research.microsoft.com/Home/Download, accessed on January 21st, 2022.
[9] https://kgtk.isi.edu/similarity_api

Table 2

Correlation scores for the raw methods and combinations that we have, for each of the benchmarks: Kendall-Tau (KT), Spearman rank (SR), and Root Mean Square Error (RMSE). Best values per column are marked in bold.

Algorithm	WordSim-353				DBPedia RG 65			DBPedia MC 30		
	Coverage	KT	SR	RMSE	KT	SR	RMSE	KT	SR	RMSE
Class	334	0.319	0.441	0.741	0	-0.031	1.054	-0.059	-0.091	1.14
Jiang Conrath	334	0.28	0.393	0.725	-0.065	-0.095	1.03	0.076	0.1	1.074
TopSim	334	0.382	0.517	0.703	0.257	0.37	0.660	0.217	0.324	0.764
Labels	334	0.041	0.059	0.732	-0.032	-0.047	0.909	0.167	0.218	0.948
Labels+desc	334	0.368	0.508	0.646	0.132	0.219	0.897	0.25	0.388	0.937
Lexicalized	334	0.374	0.512	1.031	0.408	0.581	0.734	0.45	0.597	0.799
Abstract	334	**0.523**	**0.697**	**0.523**	**0.518**	**0.662**	**0.592**	**0.567**	**0.753**	**0.568**
ComplEx	334	0.208	0.294	0.81	0.161	0.274	0.748	0.25	0.426	0.763
TransE	334	0.22	0.305	0.699	0.182	0.301	0.793	0.133	0.244	0.87
Deepwalk	327	0.281	0.392	0.731	0.238	0.322	0.741	0.291	0.422	0.683
S-Deepwalk	226	0.042	0.055	0.916	0.03	0.054	1.17	-0.143	-0.201	1.177
Composite-6	334	0.437	0.587	0.707	0.304	0.432	0.882	0.25	0.409	0.97
Composite-2	334	0.488	0.654	0.572	0.408	0.516	0.718	0.45	0.585	0.729

capturing similarity in knowledge graphs, as the hybrid method (TopSim) clearly outperforms the methods that only rely on ontological structure (class and Jiang Conrath).

The Abstract-based method performs best among all language model variants, and overall. It outperforms the other LMs because DBpedia's abstracts contain information that is more comprehensive and tailored to entity types than Wikidata labels, descriptions, or static property sets. The Lexicalization model may be improved by dynamic selection of properties, e.g., through profiling [37]. Language models perform worst when they consider the Labels alone, which can be expected because the labels contain the least information. Complementing description to the labels yields a notable improvement, whereas the Lexicalization method does slightly better than Labels+desc. These results indicate that the information fed to the language model is critical for its similarity estimation accuracy.

The graph embedding methods each focus on abstracting the rich information available in Wikidata. Among these methods, the Deepwalk embeddings perform the best. These methods are consistently outperformed by the Lexicalization and Abstract methods, suggesting that the graph embeddings' wealth of information to consider is a double-edged-sword: many properties are considered that may not be useful for determining similarity, adding distractions that can decrease performance. The Abstract method has an additional advantage over the graph embeddings in that it is less restricted in terms of the kind of information it can consider, whereas the graph embeddings focus solely on relations and can not make use of numeric- or string-valued properties.

The combination methods that we evaluated generally did not yield improved performance over the best individual method (Abstract). Because DBpedia abstracts focus on salient information that is chosen on an entity-by-entity basis, they have high utility when considering similarity and it is difficult to combine methods that consider additional information without adding noise that decreases the utility.

5.2. What is the impact of retrofitting?

Retrofitting is overall beneficial for estimating similarity (Table 3). On average across the three benchmarks, it improves the performance of nine out of the eleven methods. The highest overall improvement is observed for the S-Deepwalk method, whose Kendall-Tau score on the WD-MC30 benchmark is increased from -0.143 to 0.067. We also note a consistent improvement with the simpler methods, like Labels and Labels+desc, which can be expected given that these methods do not consider taxonomic information sufficiently before retrofitting. For example, the distance between dissimilar objects, like credit and card, is nearly the same before and after retrofitting the Labels method, whereas the distance between highly similar objects like money and cash decreases significantly (from 3.7 to 2.2, on a scale where 4 is the maximum and 1 is the minimum). The impact of retrofitting is lower on methods that consider richer information already, like Abstract and Lexicalized. This is because these methods already integrate

Table 3

Impact of retrofitting across the different benchmarks. Here we show results on retrofitting with *WD-all*, where the edges are scored with BERT-based cosine similarity. Highest Kendall-Tau (KT) values and increases per column are marked in bold.

Method	WD-WordSim353			WD-RG65			WD-MC30			Avg Δ
	Old KT	New KT	Δ	Old KT	New KT	Δ	Old KT	New KT	Δ	
Labels	0.041	0.140	**0.099**	-0.032	0.029	0.061	0.167	0.267	0.100	0.087
Labels+desc	0.368	0.448	0.080	0.132	0.154	0.021	0.25	0.283	0.033	0.045
Lexicalized	0.374	0.381	0.007	0.408	0.397	-0.011	0.450	0.400	-0.050	-0.018
Abstract	0.523	**0.567**	0.044	**0.518**	**0.497**	-0.021	**0.567**	**0.583**	0.017	0.013
TransE	0.220	0.212	-0.008	0.182	0.132	-0.05	0.133	0.167	0.033	-0.008
ComplEx	0.208	0.237	0.029	0.161	0.193	0.032	0.250	0.233	-0.017	0.015
Deepwalk	0.281	0.323	0.042	0.238	0.212	-0.026	0.291	0.291	0	0.005
S-Deepwalk	0.042	0.099	0.058	0.030	0.124	**0.093**	-0.143	0.067	**0.210**	**0.120**
Composite-6	0.437	0.489	0.052	0.304	0.275	-0.029	0.250	0.317	0.067	0.030
Composite-2	**0.483**	0.533	0.050	0.393	0.411	0.018	0.483	0.517	0.033	0.034

Table 4

Impact of different retrofitting knowledge variants on the WD-WordSim353 dataset. Highest Kendall-Tau (KT) increases per column are marked in bold.

Method	/	WD-all		WD-child-parent		WD-siblings		Probase	
	KT	KT	Δ	KT	Δ	KT	Δ	KT	Δ
Labels	0.041	0.140	**0.099**	0.156	**0.115**	0.07	0.029	0.037	**-0.004**
Labels+desc	0.368	0.448	0.080	0.457	0.089	0.401	0.033	0.318	-0.050
Lexicalized	0.374	0.381	0.007	0.372	-0.002	0.373	-0.001	0.273	-0.101
Abstract	**0.523**	**0.567**	0.044	**0.569**	0.046	**0.521**	-0.002	0.462	-0.061
TransE	0.220	0.212	-0.008	0.210	-0.010	0.212	-0.008	0.092	-0.128
ComplEx	0.208	0.237	0.029	0.248	0.040	0.219	0.011	0.123	-0.085
Deepwalk	0.281	0.323	0.042	0.323	0.042	0.296	0.015	0.220	-0.061
S-Deepwalk	0.042	0.099	0.057	0.125	0.083	0.04	-0.002	-0.091	-0.133
Composite-6	0.437	0.489	0.052	0.488	0.051	0.471	**0.034**	0.369	-0.068
Composite-2	0.483	0.533	0.050	0.529	0.046	0.500	0.017	**0.464**	-0.019

taxonomic information, and retrofitting might bring concepts that are nearly identical or merely related too close in the embedding space. For instance, retrofitting decreases the distance between seafood and lobster from 2.8 to 1.3. Still the impact of retrofitting on Abstract is positive on two out of three benchmarks, leading to new top result on the benchmarks WD-WordSim353 and WD-MC30.

We perform further analysis on the WD-WordSim353 benchmark where we split the entire set of pairs into four quartiles, where Q1 contains the most similar pairs and Q4 the least similar ones. We observe that the method performance before retrofitting is best on the identity and dissimilarity quartiles (Q1 and Q4). On these, the top-performing Abstract method has an F1 score of 0.573 and 0.578, respectively. The performance of the methods is much lower on the intermediate quartiles that signify relatedness: the performance of Abstract is 0.083 for Q2 and 0.371 for Q3. Retrofitting merely reinforces this effect: it increases the performance of the Absract method to 0.622 and 0.610 F1-score on the quartiles Q1 and Q4, whereas its impact is minimal on the quartiles Q2 and Q3. These findings indicate that similarity between highly similar and dissimilar concepts is well-understood and captured by current methods, whereas the intermediate spectrum of near-identity and relatedness requires further study and focused evaluation.

5.3. What knowledge is most beneficial for retrofitting?

We analyze the impact of different retrofitting knowledge sources in Table 4. Among the Wikidata variants, we observe that retrofitting with child-parent data performs comparable to using both child-parent and sibling data

across the methods. This result indicates that WD-sibling data is less useful for retrofitting of models. Retrofitting with ProBase's IsA relations appears to yield consistently negative results across all methods. This could be due to the quality of the underlying data, our choice to use the relation counts as similarity estimates, or the imperfect mapping of ProBase nodes to Wikidata. Comparing the results across the different methods, we again observe that the simpler methods and the composite methods benefit most from retrofitting, whereas the more elaborate methods benefit from retrofitting much less.

6. Related work

Natural Language Processing and Knowledge Graphs Natural Language Processing (NLP) research has studied the extent to which two concepts are similar or related. Here, similarity likens the notion of literal similarity in psycholinguistics, while relatedness is a broader notion that indicates that two concepts tend to appear in the same topical context [4, 38]. Literally similar concepts are found nearby in an *is-a* hierarchy, whereas related concepts connect through another relation, e.g., causality or part-whole [4]. In practice, the similarity between two concepts is evaluated by comparing human score to algorithmic score. Attaching meaning to the absolute scores is difficult, thus it is common to consider the scores across pairs relative to each other, and compare the pair similarity ordering between the algorithm and humans.

Algorithmic measures that capture the similarity between two concepts can be classified broadly into corpus-based and knowledge-based metrics [5]. Corpus-based semantic metrics are based on text analysis and typically rely on the distributional hypothesis that word meanings can be inferred based on their co-occurrence in language [39]. Language structure comes from two aspects: paradigmatic and syntagmatic. According to the paradigmatic view, linguistic symbols are regarded as paradigms which are members of a specific group (e.g., nouns), whereas the syntagmatic relationships are formed between surface symbols (e.g., words) to form a syntagm and define the meaning of a sentence. While set-based [40] and probabilistic [41] methods exist, most distributional metrics are geometric, including Latent Semantic Analysis [42], Explicit Semantic Analysis [43], and Hyperspace Analogue to Language [44]. State-of-the-art distributional measures of similarity are based on large-scale language models, such as Word2Vec [8], GloVe [45], BERT [9], and RoBERTa [10].

Knowledge-based metrics rely on analysis of ontological structures, which formally indicate how objects relate to each to other. Metrics based on graph structure estimate the similarity as a function of the degree of interconnection between concepts, captured through shortest paths [46], ontology depth [47, 48], concept specificity [49, 50], and concept density [51]. The main idea of the feature-based metrics is to represent concepts as sets of features and compute similarity by comparing these sets [52, 53]. Information theoretical methods are typically based on the Information Content (IC) of the concepts or their common ancestors [7, 47, 54], where IC is typically a proportion of all instances that belong to a concept. Hybrid measures combine aspects from multiple metric categories, e.g., depth and density with information content [7]. In recent years, various graph embeddings have been proposed [12, 13], which can be used to estimate concept similarity based on cosine distance in vector space.

Complementary to prior work, we provide a framework to investigate the role of large KGs when estimating similarity between two concepts.

Similarity of Linked Open Data Similarity allows for direct comparison between two entity representations, thus facilitating the matching of ontological schema between multiple knowledge sources [55]. In [56], the authors propose an LOD-based similarity measure based on the combination of ontological, classification, and property dimensions of knowledge. This metric has been used on a set of RDF graphs, centered around DBpedia, and evaluated on three word similarity benchmarks mapped to DBpedia. REW OrD [57] is a method to estimate relatedness between two entities, based on predicate frequency - inverse triple frequency (PF/ITF). It has been applied on DBpedia and Linked-MDB. The information content metric, called Partitioned Information Content Semantic Similarity (PICSS) [58], aims to give more importance to significant relations. Caballero and Hogan [59] propose four metrics for global node similarity in Wikidata. They also release two Wikidata benchmarks for entity similarity: movies and music albums.

Most specific similarity query (MSSQ) [60] is a similarity algorithm that can estimate similarity of two entities in a single RDF graph. The evaluation in this work shows that an approximation of MSSQ scales well, without

an indication of the extend to which the computed similarities correspond to human judgments. In [61] the author proposed a Linked Data Semantic Distance (LDSD) which relies on direct and indirect relationships between two DBpedia resources. The distance measure was employed in a music recommendation system [62]. More recently, Wang et al. [63] propose a matrix factorization method that enhances recommendation with LOD-based semantic similarity measure. Long-tail entities are explicitly considered by he collaborative filtering method in [64]. In [65], similarity is used to judge whether two representations refer to the same long-tail entity. In order to account for the large sparsity of long-tail entity knowledge, the representations include probabilistic information, based on models over instance-level knowledge in Wikidata. While prior work focuses on similarity between KG entities, we study the effect of adapting various methods that capture conceptual similarity to large KGs like Wikidata.

7. Conclusions

This paper studied representative methods for estimating similarity of concept nodes in a knowledge graph, based on language models, knowledge graph embeddings, and ontological information. The experiments revealed that pairing language models with contextualized information found in abstracts led to optimal performance. Retrofitting with taxonomic information from Wikidata generally improved performance across methods, with the simpler methods benefiting more from retrofitting. Retrofitting with the ProBase KG yielded consistently negative results, indicating that the impact of retrofitting directly depends on the quality of the underlying data. Further analysis demonstrated that both vanilla models and retrofitted models perform best on identical and dissimilar pairs. Future work should investigate contextual similarity between concepts, which would characterize partial identity and relatedness of concept pairs.

References

[1] M. Recasens, E. Hovy and M.A. Martí, Identity, non-identity, and near-identity: Addressing the complexity of coreference, *Lingua* **121**(6) (2011), 1138–1152.

[2] D. Vrandečić and M. Krötzsch, Wikidata: a free collaborative knowledgebase, *Communications of the ACM* **57**(10) (2014), 78–85.

[3] D.L.T. Rohde, L.M. Gonnerman and D.C. Plaut, An improved model of semantic similarity based on lexical co-occurence, *Communications of the Acm* **8** (2006), 627–633.

[4] T. Pedersen, S.V. Pakhomov, S. Patwardhan and C.G. Chute, Measures of semantic similarity and relatedness in the biomedical domain, *Journal of biomedical informatics* **40**(3) (2007), 288–299.

[5] S. Harispe, S. Ranwez, S. Janaqi and J. Montmain, Semantic Similarity from Natural Language and Ontology Analysis, *Synthesis Lectures on Human Language Technologies* **8**(1) (2015), 1–254, arXiv: 1704.05295. doi:10.2200/S00639ED1V01Y201504HLT027. http://arxiv.org/abs/1704.05295.

[6] P.D. Turney, Similarity of Semantic Relations, *Computational Linguistics* **32**(3) (2006), 379–416. doi:10.1162/coli.2006.32.3.379.

[7] J.J. Jiang and D. Conrath, Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy, in: *ROCLING/IJCLCLP*, 1997.

[8] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado and J. Dean, Distributed representations of words and phrases and their compositionality, in: *Advances in neural information processing systems*, 2013, pp. 3111–3119.

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

[10] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer and V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, *arXiv preprint arXiv:1907.11692* (2019).

[11] M. Faruqui, J. Dodge, S.K. Jauhar, C. Dyer, E. Hovy and N.A. Smith, Retrofitting Word Vectors to Semantic Lexicons, 2015.

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

[13] T. Trouillon, J. Welbl, S. Riedel, É. Gaussier and G. Bouchard, Complex embeddings for simple link prediction, in: *International conference on machine learning*, PMLR, 2016, pp. 2071–2080.

[14] A. Grover and J. Leskovec, node2vec: Scalable feature learning for networks, in: *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 2016, pp. 855–864.

[15] X. Zhang, Q. Yang, J. Ding and Z. Wang, Entity profiling in knowledge graphs, *IEEE Access* **8** (2020), 27257–27266.

[16] W. Wu, H. Li, H. Wang and K.Q. Zhu, Probase: A probabilistic taxonomy for text understanding, in: *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, 2012, pp. 481–492.

[17] C.E. Osgood, The similarity paradox in human learning: A resolution., *Psychological review* **56**(3) (1949), 132.

[18] D.L. Medin, R.L. Goldstone and D. Gentner, Respects for similarity., *Psychological review* **100**(2) (1993), 254.

[19] A. Tversky, Features of similarity., *Psychological review* **84**(4) (1977), 327.
[20] D. Gentner, Structure-mapping: A theoretical framework for analogy, *Cognitive science* **7**(2) (1983), 155–170.
[21] D. Gentner and A.B. Markman, Structure mapping in analogy and similarity., *American psychologist* **52**(1) (1997), 45.
[22] G.L. Murphy and D.L. Medin, The role of theories in conceptual coherence., *Psychological review* **92**(3) (1985), 289.
[23] N. Goodman, Seven strictures on similarity (1972).
[24] L.B. Smith and D. Heise, Perceptual similarity and conceptual structure, in: *Advances in psychology*, Vol. 93, Elsevier, 1992, pp. 233–272.
[25] D. Gentner, Metaphor as structure mapping: The relational shift, *Child development* (1988), 47–59.
[26] M.T. Chi, P.J. Feltovich and R. Glaser, Categorization and representation of physics problems by experts and novices, *Cognitive science* **5**(2) (1981), 121–152.
[27] E.M. Roth and E.J. Shoben, The effect of context on the structure of categories, *Cognitive psychology* **15**(3) (1983), 346–378.
[28] B. Perozzi, R. Al-Rfou and S. Skiena, DeepWalk: Online learning of social representations, *arXiv:1403.6652* (2014).
[29] F. Ilievski, P. Szekely, G. Satyukov and A. Singh, User-friendly Comparison of Similarity Algorithms on Wikidata, *arXiv preprint arXiv:2108.05410* (2021).
[30] N. Reimers and I. Gurevych, Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks, in: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, 2019. https://arxiv.org/abs/1908.10084.
[31] J. Cheng, Z. Wang, J.-R. Wen, J. Yan and Z. Chen, Contextual Text Understanding in Distributional Semantic Space, in: *ACM International Conference on Information and Knowledge Management (CIKM)*, Acm international conference on information and knowledge management (cikm) edn, ACM - Association for Computing Machinery, 2015. https://www.microsoft.com/en-us/research/publication/contextual-text-understanding-in-distributional-semantic-space/.
[32] L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman and E. Ruppin, Placing Search in Context: The Concept Revisited, *ACM Transactions on Information Systems* (2002), 20(1):116–131.
[33] N. Cheniki, A. Belkhir, Y. Sam and N. Messai, Lods: A linked open data based similarity measure, in: *2016 IEEE 25th international conference on enabling technologies: infrastructure for collaborative enterprises (WETICE)*, IEEE, 2016, pp. 229–234.
[34] H. Rubenstein and J.B. Goodenough, Contextual correlates of synonymy, *Communications of the ACM* **8**(10) (1965), 627–633.
[35] G.A. Miller and W.G. Charles, Contextual correlates of semantic similarity, *Language and cognitive processes* **6**(1) (1991), 1–28.
[36] F. Ilievski, D. Garijo, H. Chalupsky, N.T. Divvala, Y. Yao, C. Rogers, R. Li, J. Liu, A. Singh, D. Schwabe and P. Szekely, KGTK: a toolkit for large knowledge graph manipulation and analysis, in: *International Semantic Web Conference*, Springer, Cham, 2020, pp. 278–293.
[37] N. Klein, F. Ilievski and P. Szekely, Generating Explainable Abstractions for Wikidata Entities, in: *Proceedings of the 11th on Knowledge Capture Conference*, 2021, pp. 89–96.
[38] A. Budanitsky and G. Hirst, Evaluating wordnet-based measures of lexical semantic relatedness, *Computational linguistics* **32**(1) (2006), 13–47.
[39] M. Baroni and A. Lenci, Distributional memory: A general framework for corpus-based semantics, *Computational Linguistics* **36**(4) (2010), 673–721.
[40] D. Bollegala, Y. Matsuo and M. Ishizuka, Measuring semantic similarity between words using web search engines., *www* **7**(2007) (2007), 757–766.
[41] K. Church and P. Hanks, Word association norms, mutual information, and lexicography, *Computational linguistics* **16**(1) (1990), 22–29.
[42] T.K. Landauer, P.W. Foltz and D. Laham, An introduction to latent semantic analysis, *Discourse processes* **25**(2–3) (1998), 259–284.
[43] E. Gabrilovich, S. Markovitch et al., Computing semantic relatedness using Wikipedia-based explicit semantic analysis., in: *IJcAI*, Vol. 7, 2007, pp. 1606–1611.
[44] K. Lund and C. Burgess, Producing high-dimensional semantic spaces from lexical co-occurrence, *Behavior research methods, instruments, & computers* **28**(2) (1996), 203–208.
[45] J. Pennington, R. Socher and C.D. Manning, Glove: Global vectors for word representation, in: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.
[46] R. Rada, H. Mili, E. Bicknell and M. Blettner, Development and application of a metric on semantic nets, *IEEE transactions on systems, man, and cybernetics* **19**(1) (1989), 17–30.
[47] P. Resnik, Using information content to evaluate semantic similarity in a taxonomy, *arXiv preprint cmp-lg/9511007* (1995).
[48] C. Leacock and M. Chodorow, Combining local context and WordNet similarity for word sense identification, *WordNet: An electronic lexical database* **49**(2) (1998), 265–283.
[49] Z. Wu and M. Palmer, Verb semantics and lexical selection, *arXiv preprint cmp-lg/9406033* (1994).
[50] V. Pekar and S. Staab, Taxonomy learning-factoring the structure of a taxonomy into a semantic classification decision, in: *COLING 2002: The 19th International Conference on Computational Linguistics*, 2002.
[51] H. Al-Mubaid and H.A. Nguyen, A cluster-based approach for semantic similarity in the biomedical domain, in: *2006 international conference of the IEEE engineering in medicine and biology society*, IEEE, 2006, pp. 2713–2717.
[52] A. Maedche and S. Staab, Ontology learning for the semantic web, *IEEE Intelligent systems* **16**(2) (2001), 72–79.
[53] C. d'Amato, S. Staab and N. Fanizzi, On the influence of description logics ontologies on conceptual similarity, in: *International Conference on Knowledge Engineering and Knowledge Management*, Springer, 2008, pp. 48–63.
[54] G. Pirró and N. Seco, Design, implementation and evaluation of a new semantic similarity metric combining features and intrinsic information content, in: *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, Springer, 2008, pp. 1271–1288.

[55] A. Petrova, E.V. Kostylev, B.C. Grau and I. Horrocks, Towards Explainable Entity Matching via Comparison Queries., in: *OM@ ISWC*, 2019, pp. 197–198.
[56] N. Cheniki, A. Belkhir, Y. Sam and N. Messai, LODS: A Linked Open Data Based Similarity Measure, in: *2016 IEEE 25th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*, 2016, pp. 229–234. doi:10.1109/WETICE.2016.58.
[57] G. Pirró, Reword: Semantic relatedness in the web of data, in: *Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.
[58] R. Meymandpour and J.G. Davis, Enhancing Recommender Systems Using Linked Open Data-Based Semantic Analysis of Items., in: *AWC*, 2015, pp. 11–17.
[59] M. Caballero and A. Hogan, Global Vertex Similarity for Large-Scale Knowledge Graphs., in: *Wikidata@ ISWC*, 2020.
[60] A. Petrova, E.V. Kostylev, B.C. Grau and I. Horrocks, Query-based entity comparison in knowledge graphs revisited, in: *International Semantic Web Conference*, Springer, 2019, pp. 558–575.
[61] A. Passant, Measuring semantic distance on linking data and using it for resources recommendations, in: *2010 AAAI Spring Symposium Series*, 2010.
[62] A. Passant, dbrec—music recommendations using DBpedia, in: *International Semantic Web Conference*, Springer, 2010, pp. 209–224.
[63] R. Wang, H.K. Cheng, Y. Jiang and J. Lou, A novel matrix factorization model for recommendation with LOD-based semantic similarity measure, *Expert Systems with Applications* **123** (2019), 70–81.
[64] S. Natarajan, S. Vairavasundaram, S. Natarajan and A.H. Gandomi, Resolving data sparsity and cold start problem in collaborative filtering recommender system using linked open data, *Expert Systems with Applications* **149** (2020), 113248.
[65] F. Ilievski, E. Hovy, P. Vossen, S. Schlobach and Q. Xie, The role of knowledge in determining identity of long-tail entities, *Journal of Web Semantics* **61** (2020), 100565.
[66] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado and J. Dean, Distributed Representations of Words and Phrases and their Compositionality, in: *Advances in Neural Information Processing Systems*, Vol. 26, C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani and K.Q. Weinberger, eds, Curran Associates, Inc., 2013. https://proceedings.neurips.cc/paper/2013/file/9aa42b31882ec039965f3c4923ce901b-Paper.pdf.
[67] T.P. Tanon, G. Weikum and F. Suchanek, Yago 4: A reason-able knowledge base, in: *European Semantic Web Conference*, Springer, 2020, pp. 583–596.
[68] K. AlGhamdi, M. Shi and E. Simperl, Learning to Recommend Items to Wikidata Editors, *arXiv preprint arXiv:2107.06423* (2021).
[69] F. Ilievski, P. Szekely and D. Schwabe, Commonsense knowledge in wikidata, in: *ISWC Wikidata workshop*, 2020.
[70] R. Speer, J. Chin and C. Havasi, Conceptnet 5.5: An open multilingual graph of general knowledge, in: *Thirty-first AAAI conference on artificial intelligence*, 2017.
[71] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer and V. Stoyanov, RoBERTa: A Robustly Optimized BERT Pretraining Approach, *CoRR* **abs/1907.11692** (2019). http://arxiv.org/abs/1907.11692.
[72] F. Ilievski, *Identity of Long-tail Entities in Text*, Vol. 43, IOS Press, 2019.
[73] M. Recasens, E. Hovy and M.A. Martí, A Typology of Near-Identity Relations for Coreference (NIDENT), in: *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, European Language Resources Association (ELRA), Valletta, Malta, 2010. http://www.lrec-conf.org/proceedings/lrec2010/pdf/160_Paper.pdf.
[74] P.D. Turney and M.L. Littman, Learning analogies and semantic relations, *arXiv preprint cs/0307055* (2003).
[75] E.H. Huang, R. Socher, C.D. Manning and A.Y. Ng, Improving word representations via global context and multiple word prototypes, in: *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2012, pp. 873–882.
[76] B.T. McInnes and T. Pedersen, Evaluating measures of semantic similarity and relatedness to disambiguate terms in biomedical text, *Journal of biomedical informatics* **46**(6) (2013), 1116–1124.
[77] E. Hovy, T. Mitamura, F. Verdejo, J. Araki and A. Philpot, Events are not simple: Identity, non-identity, and quasi-identity, in: *Workshop on events: Definition, detection, coreference, and representation*, 2013, pp. 21–28.