Towards Utilizing Knowledge Graph Embedding Models for Conceptual Clustering

Mohamed H. Gad-Elrab[1,2], Vinh Thinh Ho[1], Evgeny Levinkov[2], Trung-Kien Tran[2], and Daria Stepanova[2]

[1] Max-Planck Institute for Informatics, Saarbrücken, Germany
{gadelrab,hvthinh}@mpi-inf.mpg.de
[2] Bosch Center for Artificial Intelligence, Renningen, Germany
{firstname.lastname}@de.bosch.com

Abstract. We propose a framework to utilize Knowledge Graph (KG) embedding models for conceptual clustering, i.e., the task of clustering a given set of entities in a KG based on the quality of the resulting descriptions for the clusters. Specifically, prominent regions in the embedding space are detected using Multicut clustering algorithm, and then the queries describing/covering the entities within these regions are obtained by rule learning. Finally, we evaluate these queries using different metrics. In our preliminary experiments, we compare the suitability of well-known KG embedding models for conceptual clustering. The reported results provide insights for the capability of these embeddings to capture graph topology and their applicability for data mining tasks beyond link prediction.

Motivation. Knowledge graphs (KGs) are collections of $\langle subject, predicate, object\rangle$ triples representing factual information in various domains. KGs are widely applied in semantic search, question answering and data analytics.

One of the important tasks in KG construction and curation is the *conceptual clustering* [14, 5], which concerns splitting a given set of entities into groups and finding descriptions for them in the form of conjunctive queries, e.g., $Q(X) \leftarrow created(Y, X), type(Y, productOwner), belongsTo(Y, boschGMBH)$ describing Bosch products. Conceptual clustering is useful in a number of applications. First, the clusters along with their descriptions facilitate the learning of new emerging concepts (*aka*, types) characterising prominent common entity properties. Second, the clusters can be exploited to refine existing KG concepts, which is useful in the context of ontology evolution. Finally, the intentionally defined groupings potentially optimize semantic search and knowledge discovery.

Recent advances in (deep) representation learning on KGs have proved to be effective for specialized tasks such as KG completion [16] and conjunctive query (CQ) answering [8, 13, 6]. In particular, in [13] queries are embedded as boxes/hyper-rectangles in the embedding space, where interior points of these boxes correspond to the set of query's answers.

In this preliminary study, our goal is to analyze the suitability of embedding models for conceptual clustering. More specifically, we aim at investigating whether prominent regions in the embedding space constructed by existing embeddings correspond to any conjunctive queries. Comparing embeddings w.r.t. their capability of capturing such queries opens new perspectives for their applicability for various data mining tasks such as conceptual clustering.

Framework Description. To start with, we introduce our proposed framework for utilizing KG embeddings for conceptual clustering. Given a KG and a set of target entities, we first compute KG embedding (see, e.g., [16]). Once the entities and relations are embedded into the vector space, we construct a complete graph $G = (V, E)$ over the target entities and compute pair-wise costs $\varPhi \colon E \mapsto \mathbb{R}$ between entity pairs using the cosine similarity of their embedding vectors.

To detect prominent regions in the embedding space we propose to use the *multicut* graph clustering approach [1], as it is an effective clustering method, which does not require the number of clusters as input. Surprisingly, multicut method has previously not been applied in the context of KGs to the best of our knowledge, despite it's obvious advantages compared to other algorithms, e.g., the small number of parameters to be tuned. A multicut of a graph is a subset of its edges s.t. no cycle in the graph intersects this subset exactly once. If we label edges in the multicut as 1, and all other edges as 0, the set of all valid multicuts can be formalized by the following set of linear inequalities:

$$Y_G = \left\{ y \colon E \mapsto \{0,1\} \mid \forall T \in \text{cycles}(G), \forall e \in T \colon y_e \leq \sum_{f \in T \setminus \{e\}} y_f \right\} \tag{1}$$

where y_e and y_f are labels of the edges e and f respectively obtained using the labeling function y. Considering only chordless cycles is sufficient [1], and any valid multicut $y \in Y_G$ uniquely defines a graph decomposition. Given the above definitions, we can formulate the **minimum cost multicut problem**:

$$\min_{y \in Y_G} \sum_{e \in E} (\varPhi_e + \beta)\, y_e \tag{2}$$

By solving (2) using efficient local search methods [10], we find an optimal multicut and the respective optimal graph decomposition, which allows us to detect prominent regions in the embedding space without knowing their number even for large KGs. The cutting prior value $\beta \in \mathbb{R}$ can be tuned to discover more ($\beta < 0$) or less ($\beta > 0$) clusters, than given by the pair-wise costs $\varPhi$ only.

The constructed regions, i.e., clusters, in the vector space are then mapped to CQs by learning *Horn rules*. E.g., the rule r : $\mathit{belongsTo}(X, c) \leftarrow \mathit{created}(Y, X)$, $\mathit{type}(Y, \mathit{productOwner}), \mathit{belongsTo}(Y, \mathit{boschGMBH})$ states that the answers to the CQ $q(X) \leftarrow \mathit{created}(Y, X), \mathit{type}(Y, \mathit{productOwner}), \mathit{belongsTo}(Y, \mathit{boschGMBH})$ belong to the cluster c. For learning such rules, we adapt [7] to capture constants in rule heads. We assess the quality of the rules using the following measures.

- *Per cluster coverage (cov)* for a rule r, cluster c, and KG $\mathcal{G}$, written as $\mathit{cov}(r, c, \mathcal{G})$, is defined as the ratio of entities covered by r within the cluster c over the cardinality of c [11].

- *Exclusive coverage (exc)* estimates exclusiveness of the rule r to the corresponding cluster c compared to other clusters from a set of clusters $\mathcal{S}$. Formally,

$$exc(r,c,\mathcal{S},\mathcal{G})=\begin{cases}0,\ \text{if} \min\limits_{c'\in\mathcal{S}\setminus c}\{cov(r,c,\mathcal{G})-cov(r,c',\mathcal{G})\}\leq 0\\ cov(r,c,\mathcal{G})-\frac{\sum\limits_{c'\in\mathcal{S}\setminus c} cov(r,c',\mathcal{G})}{|\mathcal{S}\setminus c|},\ \text{otherwise.}\end{cases}$$

- *Weighted Relative Accuracy (wra)* measures unusualness of patterns (see [11]).

Experiments. We aim at: ***(Q1)*** *comparing KG embeddings w.r.t. their suitability for conceptual clustering*, ***(Q2)*** *evaluating the correlation between the predictive quality of embeddings and their performance in conceptual clustering*, and ***(Q3)*** *verifying the effectiveness of the multicut clustering algorithm in our context compared to other common clustering algorithms.*

We implemented our framework in Python, using the KG embedding models from Ampligraph [2] and Multicut algorithm [10]. We experimented with *TransE (T)* and *ComplEx (C)*, which are respectively representatives of translation-based and linear map embedding models. We trained the two models for 100 epochs with embedding dimension set to 100. Experimenting with other embedding models is left for future work.

As described above, the cosine similarity is used to compute the pairwise distances between entities. We tried several cutting prior values β for the Multicut algorithm and report here the best results. The Multicut is compared to commonly used clustering algorithms, namely, DBSCAN [4], k-means [12], and Spectral clustering [9], whose parameters are tuned and the best results are reported. For fair comparison, we pass the number of clusters produced by Multicut to k-means and Spectral clustering, as both of these algorithms require this parameter as input.

We performed experiments on widely-used relational datasets: Hepatitis, Mutagensis, WebKB, and Terrorist Attacks (see [3] for dataset statistics). In addition, we experimented with a large scale dataset *YAGO-Artwork* which contains 3.9K target entities for *books*, *songs*, and *movies*, randomly selected from YAGO KG [15]. Note that, while the conceptual clustering procedure is applied only on a subset of entities *i.e.*, target entities, the embeddings are trained on the full KGs to ensure that the semantic data for all of the entities is preserved.

We evaluated the predictive quality of the trained embeddings using standard *Mean Reciprocal Rank (MRR)* and *Hit@k* measures, and assessed the quality of the mined queries using the functions *cov*, *exc*, and *wra* introduced above.

Results. Table 1 shows the predictive quality of the KG embedding models, where ComplEx consistently achieves better results than TransE on all datasets, except YAGO. Training embeddings for this KG is particularly challenging for both models, and ComplEx could not converge within the training epochs due to its complexity.

In Table 2, we report the estimated quality of the discovered queries. For each dataset, we present the average *cov*, *exc*, *wra* measures, and the *number of*

Table 1: The predictive quality of KG embedding models

	TransE				ComplEx			
	MRR	*Hit@1*	*Hit@3*	*Hit@10*	*MRR*	*Hit@1*	*Hit@3*	*Hit@10*
Hepatitis	0.929	0.903	0.947	0.974	**0.946**	**0.919**	**0.970**	**0.988**
Mutagenesis	0.896	0.840	0.959	0.983	**0.953**	**0.919**	**0.988**	**0.998**
WebKB	0.210	0.158	0.223	0.293	**0.415**	**0.329**	**0.441**	**0.584**
Terrorist	0.320	0.140	0.429	0.623	**0.930**	**0.876**	**0.984**	**0.998**
YAGO-Art	**0.105**	**0.085**	**0.111**	**0.133**	0.000	0.000	0.000	0.000

Table 2: Quality of learned rules; "–" refers to the failure of finding clusters

Methods	**Hepatitis**				**Mutagenesis**				**WebKB**				**Terrorist**				**YAGO-Art**			
	cov	*exc*	*wra*	*cls*	*cov*	*exc*	*wra*	*cls*	*cov*	*exc*	*wra*	*cls*	*cov*	*exc*	*wra*	*cls*	*cov*	*exc*	*wra*	*cls*
Multicut-T	0.567	0.567	0.037	6	**0.917**	0.667	0.004	4	**0.948**	0.147	**0.009**	5	0.789	0.382	**0.048**	4	0.763	**0.695**	0.053	3
Multicut-C	**0.826**	**0.764**	**0.005**	4	0.773	0.625	**0.097**	2	0.849	**0.271**	0.002	2	**0.797**	**0.713**	**0.048**	3	0.820	0.211	0.000	6
DBSCAN-T	0.707	0.582	0.028	3	–	–	–	–	–	–	–	–	0.757	0.423	0.005	2	**0.908**	0.578	**0.072**	4
DBSCAN-C	0.732	0.616	0.036	2	**0.917**	**0.750**	0.016	5	–	–	–	–	0.745	0.564	0.005	3	–	–	–	–
K-Means-T	0.610	0.491	0.065	6	0.793	0.041	0.012	4	0.969	0.015	0.013	5	0.622	0.231	0.062	4	0.672	0.404	0.094	3
K-Means-C	0.658	0.561	0.110	4	0.941	0.882	0.220	2	0.951	0.105	0.026	2	0.774	0.516	0.097	3	0.751	0.006	0.001	6
Spectral-T	0.592	0.288	0.078	6	0.932	0.035	0.012	4	0.966	0.041	0.018	5	0.667	0.270	0.062	4	0.675	0.400	0.094	3
Spectral-C	0.775	0.616	0.055	4	1.000	0.482	0.008	2	0.919	0.427	0.003	2	0.839	0.750	0.047	3	0.810	0.005	0.001	6

discovered clusters (cls). Conceptual clustering over ComplEx results in better average *exc* and *wra*, which answers our first research question ***(Q1)***. In addition, the higher predictive quality of ComplEx in Table 1, supports the hypothesis ***(Q2)***, suggesting correlation between the predictive quality and the quality of the discovered queries. This also holds for YAGO, where TransE performs better than ComplEx in both prediction and clustering.

Regarding ***(Q3)***, multicut achieved better results compared to DBSCAN in the majority of datasets. Moreover, DBSCAN failed in several cases regardless of the used parameters. Interestingly, even compared to other clustering algorithms that require the number of clusters, Multicut performed better on several datasets. This demonstrates the suitability of this algorithm for the KG domain.

Finally, Table 3 shows example rules mined from YAGO over TransE. E.g., r_1 describes entities in c_1 as *"artifacts that have a director and an actor"*, while r_3 describes entities in c_2 as: *"artifacts created by an award winner"*.

Conclusion and Outlook. We have introduced a framework for utilizing KG embeddings for the task of conceptual clustering, which exploits the Multicut algorithm [1, 10] for detecting prominent regions in the embedding space and maps them to conjunctive queries over KGs using an extension of [7]. We believe that our framework and preliminary experimental results contribute to a better understanding of the strengths and weaknesses of the existing KG embeddings beyond fact prediction.

For future work, we plan to extend our framework to account for background knowledge in the form of ontologies. Additionally, we will consider comparing the suitability of other embedding models for the task of conceptual clustering, especially those trained on complex patterns [13]. Another interesting direction is to investigate utilizing the learned rules for guiding the clustering process.

Table 3: Example rules from YAGO-Art dataset learned over TransE.

Query	**cov**	**exc**	**wra**
$r_1 : \mathit{belongsTo}(X, c_1) \leftarrow \mathit{directed}(Y, X), \mathit{acted}(Z, X)$	0.768	0.749	0.153
$r_2 : \mathit{belongsTo}(X, c_1) \leftarrow \mathit{actedIn}(Y, X), \mathit{type}(Y, \mathit{film\_actor})$	0.724	0.708	0.144
$r_3 : \mathit{belongsTo}(X, c_2) \leftarrow \mathit{created}(Y, X), \mathit{hasWonPrize}(Y, Z)$	0.564	0.424	0.093
$r_4 : \mathit{belongsTo}(X, c_2) \leftarrow \mathit{created}(Y, X), \mathit{type}(Y, \mathit{writer})$	0.504	0.420	0.091

References

1. Chopra, S., Rao, M.: The partition problem. Math. Prog. **59**(1–3), 87–115 (1993)
2. Costabello, L., Pai, S., Van, C.L., McGrath, R., McCarthy, N., Tabacof, P.: AmpliGraph: a Library for Representation Learning on Knowledge Graphs (Mar 2019)
3. Dumancic, S., Blockeel, H.: An expressive dissimilarity measure for relational clustering over neighbourhood trees. MLJ (2017)
4. Ester, M., Kriegel, H.P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. pp. 226–231. AAAI Press (1996)
5. Fanizzi, N., d'Amato, C., Esposito, F.: Conceptual clustering and its application to concept drift and novelty detection. In: ESWC. pp. 318–332 (2008)
6. Friedman, T., den Broeck, G.V.: Symbolic querying of vector spaces: Probabilistic databases meets relational embeddings. CoRR **2002.10029** (2020)
7. Galárraga, L., Teflioudi, C., Hose, K., Suchanek, F.M.: Fast rule mining in ontological knowledge bases with amie++. The VLDB Journal **24**(6), 707–730 (2015)
8. Hamilton, W.L., Bajaj, P., Zitnik, M., Jurafsky, D., Leskovec, J.: Embedding logical queries on knowledge graphs. In: NeurIPS 2018. pp. 2030–2041 (2018)
9. Jianbo Shi, Malik, J.: Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **22**(8), 888–905 (2000)
10. Keuper, M., Levinkov, E., Bonneel, N., Lavoué, G., Brox, T., Andres, B.: Efficient decomposition of image and mesh graphs by lifted multicuts. In: ICCV (2015)
11. Lavrač, N., Flach, P., Zupan, B.: Rule evaluation measures: A unifying view. In: ILP (1999)
12. MacQueen, J., et al.: Some methods for classification and analysis of multivariate observations. In: Symp. on math. stat. and prob. vol. 1, pp. 281–297 (1967)
13. Ren, H., Hu, W., Leskovec, J.: Query2box: Reasoning over knowledge graphs in vector space using box embeddings. In: ICLR 2020 (2020)
14. Suárez, A.P., Mart'inez Trinidad, J.F., Carrasco-Ochoa, J.A.: A review of conceptual clustering algorithms. Artif. Intell. Rev. **52**(2), 1267–1296 (2019)
15. Suchanek, F.M., Kasneci, G., Weikum, G.: Yago: A Core of Semantic Knowledge. In: Proceedings of WWW. pp. 697–706 (2007)
16. Wang, Q., Mao, Z., Wang, B., Guo, L.: Knowledge graph embedding: A survey of approaches and applications. IEEE Trans. Knowl. Data Eng. **29**(12), 2724–2743 (2017)