Inducing Interpretability in Knowledge Graph Embeddings

Chandrahas[1] **Tathagata Sengupta**[2†*] **Cibi Pragadeesh**[3*†] **Partha Talukdar**[1]
[1]Indian Institute of Science, Bangalore, [2]Adobe, Bangalore,
[3]University of California, Los Angeles
chandrahas@iisc.ac.in, tathagatasengupta3@gmail.com
cibi.pragadeesh@gmail.com, ppt@iisc.ac.in

Abstract

We study the problem of inducing interpretability in Knowledge Graph (KG) embeddings. Learning KG embeddings has been an active area of research in the past few years, resulting in many different models. However, most of these methods do not address the interpretability (semantics) of individual dimensions of the learned embeddings. In this work, we study this problem and propose a method for inducing interpretability in KG embeddings using entity co-occurrence statistics. The proposed method significantly improves the interpretability, while maintaining comparable performance in other KG tasks.

1 Introduction

Knowledge Graphs such as Freebase (Bollacker et al., 2008) and NELL (Mitchell et al., 2015) have become important resources for supporting many AI applications like web search, Q&A, etc. They store a collection of facts in the form of a graph. The nodes in the graph represent real world entities such as *Roger Federer*, *Tennis*, *United States* etc while the edges represent relationships between them.

These KGs have grown huge, but they are still not complete (Toutanova et al., 2015). Hence the task of inferring new facts becomes important. KG embeddings have been a popular approach for this task as they can perform the inference task efficiently. This task has achieved significant attention in the literature and many methods have been proposed, such as, (Bordes et al., 2013; Riedel et al., 2013; Yang et al., 2014; Toutanova et al., 2015; Trouillon et al., 2016; Schlichtkrull et al., 2017; Dettmers et al., 2018; Balazevic et al., 2019), etc. These methods learn representations for entities and relations as vectors in a vector space, capturing global information about the KG. The task of KG inference is then defined as operations over these vectors. Some of these methods like (Riedel et al., 2013) and (Toutanova et al., 2015) are capable of exploiting additional text data apart from the KG, resulting in better representations.

Although these methods have shown good performance in the end task, they do not address the interpretability, i.e., understanding semantics of individual dimensions of the KG embedding. Such representations enable a better understanding of the model and can be helpful for explaining a model's decision on an end application.

In this work, we focus on incorporating interpretability in KG embeddings. Specifically, we aim to learn interpretable embeddings for KG entities by incorporating additional entity co-occurrence statistics from text data. This work is motivated by (Lau et al., 2014) who presented automated methods for evaluating topics learned via topic modelling methods. We adapt these methods for KG embedding models and propose a method to directly maximize them while learning KG embedding. As demonstrated by the experiments, we find that such modeling significantly improves interpretability, supporting our choice of using topic coherence for embedding dimensions. To the best of our knowledge, this work presents the first regularization term which induces interpretability in KG embeddings.

2 Related Work

Several methods have been proposed for learning KG embeddings. They differ on the modeling of entities and relations, usage of text data and interpretability of the learned embeddings. We summarize some of these methods in following sections.

[*] Contributed equally to the work.

[†] This research was conducted during the authors' internships at Indian Institute of Science, Bangalore.

Proceedings of the 17th International Conference on Natural Language Processing, pages 70–75
Patna, India, December 18 - 21, 2020. ©2020 NLP Association of India (NLPAI)

2.1 KG Embedding models

Most of the KG embedding models represent entities and relations as vectors in $\mathbb{R}^{d_e}$ and $\mathbb{R}^{d_r}$ respectively (usually, d_e=d_r). A score function uses these vectors to calculate the correctness of a given triple. Based on the score function, these methods can be categorized as additive models (Bordes et al., 2013; Lin et al., 2015; Xiao et al., 2015; Xie et al., 2017), multiplicative models (Nickel et al., 2011; Yang et al., 2014; Trouillon et al., 2016; Balazevic et al., 2019) and nueral models (Dong et al., 2014; Dettmers et al., 2018). There are other methods which are able to incorporate text data while learning KG embeddings. For example, the method proposed in (Riedel et al., 2013) assumes a combined universal schema of relations from KG as well as text. This method is further improved in (Toutanova et al., 2015) using textual relation encoder allowing parameter sharing among similar textual relations. However, none of these methods address the interpretability of the embeddings.

2.2 Interpretability of Embeddings

While the KG embedding models perform well in many tasks, the semantics of learned representations are not directly clear. This problem for word embeddings has been addressed in (Murphy et al., 2012; Faruqui et al., 2015; Subramanian et al., 2018) where they apply a set of constraints inducing interpretability. A similar task of learning semantic features for entities and relations is KG was addressed in (Xiao et al., 2016). However, their approach is not applicable for the much popular KG embedding methods. The model proposed in (Xie et al., 2017) can generate interpretable embeddings for relations, but not entities. Another approach, as proposed in (Gusmao et al., 2018), is to generate weighted Horn rules as explanations for link prediction. We refer the reader to Section 4 of (Bianchi et al., 2020) for further reading in this direction.

Our method differs from the previous works in the following aspects. Firstly, we focus on learning interpretable embeddings for KG entities rather than relations. Second, we incorporate side information about entities instead of constraints for inducing interpretability. Third, we use vector space modeling rather than probabilistic modelling (as in (Xiao et al., 2016)) allowing the proposed method to be applicable to many existing KG embedding models.

3 Proposed Method

The proposed method is motivated by a measure of coherence in topic modelling literature (Lau et al., 2014). This measure allows an automated evaluation of the quality of topics learned by topic modeling methods by using additional Point-wise Mutual Information (PMI) for word pairs. It was also shown to have high correlation with human evaluation of topics.

Based on this measure of coherence, we propose a regularization term. This term can be used with existing KG embedding methods for inducing interpretability. It is described in the following sections.

3.1 Coherence

In topic models, coherence of a topic can be determined by semantic relatedness among top entities within the topic. This idea can also be used in vector space models by treating dimensions of the vector space as topics. With this assumption, we can use a measure of coherence defined in following section for evaluating interpretability of the embeddings.

3.1.1 $Coherence@k$

Coherence for top k entities along dimension l is defined as follows.

$$Coherence@k^{(l)} = \sum_{i=2}^{k} \sum_{j=1}^{i-1} p_{ij} \qquad (1)$$

where p_{ij} is PMI score between entities e_i and e_j extracted from text data. It is given as follows

$$p_{ij} = \log \left(\frac{Pr(e_i, e_j)}{Pr(e_i) \times Pr(e_j)} \right). \qquad (2)$$

Here, $Pr(e_i, e_j)$ represents the joint probability of co-occurrence of entities e_i and e_j, while $Pr(e_i)$ and $Pr(e_j)$ represent the corresponding marginal probabilities, pre-computed using an auxiliary corpus.

$Coherence@k$ has been shown to have high correlation with human interpretability of topics learned via various topic modeling methods(Lau et al., 2014). Hence, we can expect interpretable embeddings by maximizing it.

$Coherence@k$ for the entity embedding matrix θ_e is defined as the average over all dimensions.

$$Coherence@k = \frac{1}{d} \sum_{l=1}^{d} Coherence@k^{(l)}. \qquad (3)$$

3.1.2 Inducing coherence while learning embeddings

We want to learn an embedding matrix θ_e which has high coherence (i.e., which maximizes $Coherence@k$). Since θ_e changes during training, the set of top k entities along each dimension varies over iterations. Hence, directly maximizing $Coherence@k$ may not be feasible.

An alternative approach could be to promote higher values for entity pairs having high PMI score p_{ij}. This will result in an embedding matrix θ_e with a high value of $Coherence@k$ since high PMI entity pairs are more likely to be among top k entities.

This idea can be captured by following coherence term

$$\mathcal{C}(\theta_e, P) = \sum_{i=2}^{n} \sum_{j=1}^{i-1} \|v(e_i)^\intercal v(e_j) - p_{ij}\|^2 \quad (4)$$

where P is entity-pair PMI matrix and $v(e)$ denote vector for entity e. This term can be used in the objective function defined in (7).

3.2 Entity Model (Model-E)

We use the Entity Model proposed in (Riedel et al., 2013) for learning KG embeddings. However, it should be noted that the proposed regularizer can be used along with any KG embedding model which represents entities as vectors. Also, as pointed in (Kadlec et al., 2017; Ruffinelli et al., 2020; Jain et al., 2020), various KG embedding models achieve similar performances when trained properly. Therefore, we select Model-E which is simple yet effective. This model assumes a vector $v(e)$ for each entity and two vectors $v_s(r)$ and $v_o(r)$ for each relation of the KG. The score for the triple (e_s, r, e_o) is given by,

$$f(e_s, r, e_o) = v(e_s)^\intercal v_s(r) + v(e_o)^\intercal v_o(r). \quad (5)$$

Training these vectors requires incorrect triples. So, we use the closed world assumption. For each triple $t \in \mathcal{T}$, we create two negative triples t_o^- and t_s^- by corrupting the object and subject of the triples respectively such that the corrupted triples do not appear in training, test or validation data. The loss for a triple pair is defined as $loss(t, t^-) = -\log(\sigma(f(t) - f(t^-)))$. Then, the aggregate loss function is defined as

$$L(\theta_e, \theta_r, \mathcal{T}) = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \left(loss(t, t_o^-) + loss(t, t_s^-)\right). \quad (6)$$

3.3 Objective

The overall loss function can be written as follows

$$L(\theta_e, \theta_r, \mathcal{T}) + \lambda_c \mathcal{C}(\theta_e, P) + \lambda_r \mathcal{R}(\theta_e, \theta_r) \quad (7)$$

where $\mathcal{R}(\theta_e, \theta_r) = \frac{1}{2}\left(\|\theta_e\|^2 + \|\theta_r\|^2\right)$ is the $L2$ regularization term and λ_c and λ_r are hyper-parameters controlling the trade-off among different terms in the objective function.

4 Experiments and Results

4.1 Datasets

We use the FB15k-237 (Toutanova and Chen, 2015) dataset, a factual KG, for experiments. It contains 14541 entities and 237 relations. The triples are split into training, validation and test set having 272115, 17535 and 20466 triples respectively. For extracting entity co-occurrences, we use the textual relations used in (Toutanova et al., 2015). It contains around 3.7 millions textual triples, which we use for calculating PMI for entity pairs.

4.2 Experimental Setup

We use the method proposed in (Riedel et al., 2013) as the baseline. Please refer to Section 3.2 for more details. For evaluating the learned embeddings, we test them on different tasks. All the hyper-parameters are tuned using performance (MRR) on validation data. We use 100 dimensions after cross validating among 50, 100 and 200 dimensions. For regularization, we use $\lambda_r = 0.01$ (from $10, 1, 0.1, 0.01$) and $\lambda_c = 0.01$ (from $10, 1, 0.1, 0.01$) for $L2$ and coherence regularization respectively. We use multiple random initializations sampled from a Gaussian distribution. For optimization, we use gradient descent and stop optimization when gradient becomes 0 upto 3 decimal places. The final performance measures are reported for test data.

4.3 Results

In following sections, we compare the performance of the proposed method with the baseline method in different tasks. Please refer to Table 1 for results.

Method	Link Prediction		
	MRR↑	MR↓	Hits@10(%)↑
Baseline	$\mathbf{31.6 \pm 0.08}$	121.9 ± 1.80	$\mathbf{48.3 \pm 0.39}$
Proposed	30.4 ± 0.08	$\mathbf{111.9 \pm 1.12}$	46.8 ± 0.08
	Triple Classification		
	AUC(%)↑	Accuracy(%)↑	
Baseline	72.9 ± 0.16	63.2 ± 0.50	
Proposed	$\mathbf{73.2 \pm 0.28}$	$\mathbf{67.6 \pm 0.17}$	
	Interpretability		
	AutoWI@5(%)↑	Coherence@5↑	Manual WI(%)↑
Baseline	6 ± 4.14	-47.4 ± 4.68	12
Proposed	$\mathbf{66 \pm 5.89}$	$\mathbf{-12.5 \pm 4.48}$	**84**

Table 1: Results of various tasks on FB15k-237 dataset. Here ↑ indicates higher values are better while ↓ indicates lower values are better. The proposed method significantly improves interpretability while maintaining comparable performance on KG tasks (4.3).

Top 5
Baseline
-**Jurist**, **Pipe organ**, USA, **Lions Gate Entertainment**, UK
-Guitar, **71st Academy Awards**, **Jurist**, Piano, Bass guitar
-Actor, **Official Website**, Screenwriter, Film Producer, **USA**
-**Jurist**, USA, **Marriage**, **Male**, UK
-**Pipe organ**, **Official Website**, Actor, Film Producer, Screenwriter
Proposed Method
-Juris Doctor, Business Administration, Biology, Psychology, BS
-Bachelor of Arts, PhD, Bachelor's degree, BS, MS
-European Union, Europe, Netherlands, Portugal, **Government**
-UK, Hollywood, **DVD**, London, Europe
-Hollywood, Academy Awards, **USA**, DVD, **Los Angeles**

Table 2: Top 5 entities for randomly selected dimensions. As we see, the proposed method produces more coherent entities compared to the baseline. Incoherent entities are marked in bold face. [1]

4.3.1 Interpretability

For evaluating the interpretability, we use $Coherence@k$ (3), automated and manual word intrusion tests. In word intrusion test (Chang et al., 2009), top $k(=5)$ entities along a dimension are mixed with the bottom most entity (the intruder) in that dimension and shuffled. Then multiple (3 in our case) human annotators are asked to find out the intruder. We use majority voting to finalize one intruder. Amazon Mechanical Turk was used for crowdsourcing the annotation task and we used 25 randomly selected dimensions for evaluation. Thus, each of the three annotators evaluates 25 examples. For automated word intrusion (Lau et al., 2014), we calculate following score for all $k+1$ entities

$$\text{AutoWI}(e_i) = \sum_{j=1, j \neq i}^{k+1} p_{ij} \tag{8}$$

where p_{ij} are the PMI scores. The entity having least score is identified as the intruder. We report the fraction of dimensions for which we were able to identify the intruder correctly.

As we can see in Table 1, the proposed method achieves better values for $Coherence@5$ as a direct consequence of the regularization term, thereby maximizing coherence between appropriate entities. Performance on the word intrusion task also improves drastically as the intruder along each dimension is a lot easier to identify owing to the fact that the top entities for each dimension group together more conspicuously.

4.3.2 Link Prediction

In this experiment, we test the model's ability to predict the best object entity for a given subject entity and relation. For each of the triples, we fix the subject and the relation and rank all entities (within same category as true object entity) based on their score according to (5). We report Mean Rank (MR) and Mean Reciprocal rank (MRR) of the true object entity and Hits@10 (the number of times true object entity is ranked in top 10) as percentage. A good model should have higher values for MRR and Hits@10, and lower value for MR.

The coherence regularization term's objective, being tangential to that of the original loss function, is not expected to affect the link prediction task's performance. However, the results show a trivial drop of 1.2 in MRR. Upon further inspection, we found that the coherence term gives credibility to certain triples otherwise deemed incorrect by the closed world assumption. These triples appear in the text corpus and contain entity pairs with high PMI values.

4.3.3 Triple Classification

In this experiment, we test the model on classifying correct and incorrect triples. For finding incorrect triples, we corrupt the object entity with a randomly selected entity within the same category. For classification, we use validation data to find the best threshold for each relation by training an SVM classifier and later use this threshold for classifying test triples. We report the mean accuracy and mean AUC over all relations.

We observe that the proposed method achieves slightly better performance for triple classification

[1] We have used abbreviations for BS (Bachelor of Science), MS (Master of Science), UK (United Kingdom) and USA (United States of America). They appear as full form in the data.

improving the accuracy by 4.4. The PMI information adds more evidence to the correct triples which are related in text data, generating a better threshold that more accurately distinguishes correct and incorrect triples.

4.4 Qualitative Analysis of Results

Since our aim is to induce interpretability in representations, in this section, we evaluate the embeddings learned by the baseline as well as the proposed method. For both methods, we select some dimensions randomly and present top 5 entities along those dimensions. As we can see from the results in Table 2, the proposed method produces more coherent entities than the baseline method.

5 Conclusion and Future Works

In this work, we proposed a method for inducing interpretability in KG embeddings using a coherence regularization term. We evaluated the proposed and the baseline method on the interpretability of the learned embeddings. We also evaluated the methods on different KG tasks and compared their performance. We found that the proposed method achieves better interpretability while maintaining comparable performance on KG tasks. As next steps, we plan to evaluate and compare the generalizability of the proposed method across various KG embedding models. Understanding the mapping between dimensions and latent categories could be another direction for future works.

Acknowledgments

We thank the anonymous reviewers for their constructive comments. This work is supported by the Ministry of Human Resources Development (Government of India).

References

Ivana Balazevic, Carl Allen, and Timothy Hospedales. 2019. TuckER: Tensor factorization for knowledge graph completion. In *EMNLP-IJCNLP*, pages 5184–5193, Hong Kong, China. ACL.

Federico Bianchi, Gaetano Rossiello, Luca Costabello, Matteo Palmonari, and Pasquale Minervini. 2020. Knowledge graph embeddings and explainable ai. *arXiv preprint arXiv:2004.14843*.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. AcM.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in neural information processing systems*, pages 2787–2795.

Jonathan Chang, Jordan L Boyd-Graber, Sean Gerrish, Chong Wang, and David M Blei. 2009. Reading tea leaves: How humans interpret topic models. In *Nips*, volume 31, pages 1–9.

Tim Dettmers, Minervini Pasquale, Stenetorp Pontus, and Sebastian Riedel. 2018. Convolutional 2d knowledge graph embeddings. In *Proceedings of the 32th AAAI Conference on Artificial Intelligence*, pages 1811–1818.

Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. 2014. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 601–610. ACM.

Manaal Faruqui, Yulia Tsvetkov, Dani Yogatama, Chris Dyer, and Noah A. Smith. 2015. Sparse overcomplete word vector representations. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1491–1500, Beijing, China. Association for Computational Linguistics.

Arthur Colombini Gusmao, Alvaro Henrique Chaim Correia, Glauber De Bona, and Fabio Gagliardi Cozman. 2018. Interpreting embedding models of knowledge bases: a pedagogical approach.

Prachi Jain, Sushant Rathi, Soumen Chakrabarti, et al. 2020. Knowledge base completion: Baseline strikes back (again). *arXiv preprint arXiv:2005.00804*.

Rudolf Kadlec, Ondrej Bajgar, and Jan Kleindienst. 2017. Knowledge base completion: Baselines strike back. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 69–74, Vancouver, Canada. Association for Computational Linguistics.

Jey Han Lau, David Newman, and Timothy Baldwin. 2014. Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. In *EACL*, pages 530–539.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *AAAI*, pages 2181–2187.

T. Mitchell, W. Cohen, E. Hruschka, P. Talukdar, J. Betteridge, A. Carlson, B. Dalvi, M. Gardner, B. Kisiel, J. Krishnamurthy, N. Lao, K. Mazaitis, T. Mohamed,

N. Nakashole, E. Platanios, A. Ritter, M. Samadi, B. Settles, R. Wang, D. Wijaya, A. Gupta, X. Chen, A. Saparov, M. Greaves, and J. Welling. 2015. Never-ending learning. In *Proceedings of AAAI*.

Brian Murphy, Partha Pratim Talukdar, and Tom Mitchell. 2012. Learning effective and interpretable semantic models using non-negative sparse embedding. In *International Conference on Computational Linguistics (COLING 2012), Mumbai, India*. http://aclweb.org/anthology/C/C12/C12-1118.pdf.

Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2011. A three-way model for collective learning on multi-relational data. In *ICML*, volume 11, pages 809–816.

Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M Marlin. 2013. Relation extraction with matrix factorization and universal schemas. *NAACL HLT 2013*, pages 74–84.

Daniel Ruffinelli, Samuel Broscheit, and Rainer Gemulla. 2020. You can teach an old dog new tricks! on training knowledge graph embeddings. In *International Conference on Learning Representations*.

M. Schlichtkrull, T. N. Kipf, P. Bloem, R. van den Berg, I. Titov, and M. Welling. 2017. Modeling Relational Data with Graph Convolutional Networks. *ArXiv e-prints*.

Anant Subramanian, Danish Pruthi, Harsh Jhamtani, Taylor Berg-Kirkpatrick, and Eduard Hovy. 2018. Spine: Sparse interpretable neural embeddings. *The Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18)*.

Kristina Toutanova and Danqi Chen. 2015. Observed versus latent features for knowledge base and text inference. In *3rd Workshop on Continuous Vector Space Models and Their Compositionality*. ACL – Association for Computational Linguistics.

Kristina Toutanova, Danqi Chen, Patrick Pantel, Hoifung Poon, Pallavi Choudhury, and Michael Gamon. 2015. Representing text for joint embedding of text and knowledge bases. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1499–1509, Lisbon, Portugal. Association for Computational Linguistics.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex embeddings for simple link prediction. In *ICML*.

Han Xiao, Minlie Huang, Yu Hao, and Xiaoyan Zhu. 2015. Transg: A generative mixture model for knowledge graph embedding. *arXiv preprint arXiv:1509.05488*.

Han Xiao, Minlie Huang, and Xiaoyan Zhu. 2016. Knowledge semantic representation: A generative model for interpretable knowledge graph embedding. *arXiv preprint arXiv:1608.07685*.

Qizhe Xie, Xuezhe Ma, Zihang Dai, and Eduard Hovy. 2017. An interpretable knowledge transfer model for knowledge base completion. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 950–962, Vancouver, Canada. Association for Computational Linguistics.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2014. Embedding entities and relations for learning and inference in knowledge bases. *arXiv preprint arXiv:1412.6575*.