International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-13 Issue-5, April 2024

Missing Link Prediction in Art Knowledge Graph using Representation Learning

Swapnil S. Mahure, Anish R. Khobragade

***Abstract:** Knowledge graphs are an important evolving field in Artificial Intelligence domain which has multiple applications such as in question answering, important information retrieval, information recommendation, Natural language processing etc. Knowledge graph has one big limitation i.e. Incompleteness, it is due to because of real world data are dynamic and continues evolving. This incompleteness of Knowledge graph can be overcome or minimized by using representation learning models. There are several models which are classified on the base of translation distance, semantic information and NN (Neural Network) based. Earlier the various embedding models are test on mostly two well-known datasets WN18RR & FB15k-237. In this paper, new dataset i.e. ArtGraph has been utilised for link prediction using representation learning models to enhance the utilization of ArtGraph in various domains. Experimental results shown ConvKB performed better over the other models for link prediction task.*

***Index Terms:** KG Embeddings, Artwork, Link Prediction, Neural Network*

I. INTRODUCTION

KGs are collection of relations and entities, connected by various kinds of edges and nodes, respectively. KG embedding is a technique of mapping for translating the real world information of $\vec{\varepsilon}$ (#entities) & $\vec{r}$ (#relations) in a KG to a low-dimensional space of vector of continuous values. Knowledge Graphs contains enormous data so it also known as one of the big data application. They contains facts which has millions relations and entities [4]. Each fact of real world is represented as a triple. The triples are denoted by t=tail entity, h=head entity, r=relation between head and tail. Databases that contain triples (`head-`relation-`tail) to express the relationships between $\vec{\varepsilon}$ (#entities) in the pattern of fact (h_n, r_n, t_n), e.g., (Pune, locatedIn, Maharashtra).

Knowledge graph have some disadvantage as they are always not complete [4]. So, it is quite challenging problem to build more correct and accurate graph. To construct more complete graph we often formulate as the link prediction problem. A huge Knowledge Bases can be failing to convince valid triplate score because of its size.

Manuscript received on 18 August 2022 | Revised Manuscript received on 03 September 2022 | Manuscript Accepted on 15 April 2024 | Manuscript published on 30 April 2024.
**Correspondence Author(s)*
Swapnil S. Mahure*, College of Engineering, COEP Technological University Pune (Maharashtra), India. Email: mahuress20.mfg@coep.ac.in
Anish R. Khobragade, College of Engineering, COEP Technological University Pune (Maharashtra), India. Email: anishraj.comp@coep.ac.in

The examples of open KGs are WN18RR [2], FB15k-237 [18], YAGO [8] and DBpedia [7] are the databases of triple representation. KBs are utilised in a several applications, including question-answering [6][14][15][16][17][18], data retrieval, semantic searching, etc. As we mentioned above most of the KBs are still not complete, there is a problem of handling missing valid triplets. Much research work has been devoted to link prediction task whether it is valid or invalid. There are many embedding models which take notes from the representation on the vector form for of $\vec{\varepsilon}$ (#entities) and $\vec{r}$ (#relations). There are several types of embedding models used for link prediction. Translation model, Semantic information based model, NN (Neural Network models). Some of known translation models are TransE [5], TransH [12], TransR [10], TransM [13] etc. Let us take TransE, in which for a triplet hed, tail and relation represented in lower dimensional vectors $h_n, r_n, t_n \in R^k$ respectively, where k is a embedding dimension. In order to model relationships between entities, TransE [5] uses a transitional feature. It makes the assumption that if (h_n, r_n, t_n) is a true fact, the embedding of *h (*head entity) plus the relation embedding *r* should be near the tail entity's embedding *t*. i.e. $h_n, + r_n, \approx t_n$. Similarly TransH [12], TransR [10], TransM [13] uses the TransE [5] approach in modified way. In DISTMULT [9] and ComplEx [11] models, the score for each triple is calculated using a tri-linear dot product of factorised tensor X which represent semantic relation between pair of entities.

Now a day Neural Network is significantly used in research work. Here ConvE and ConvKB are the two models we using in this paper. ConvKB gives better result on KG than ConvE due to their some limitations. In ConvE, the input matrix for the convolution layer is created by reshaping vh and vr and then concatenating them. To produce feature map tensors, various filters with the same 3×3 shape are applied to the input matrix. Then, using a linear transformation, these feature map tensors are vectorized and translated into a vector.

II. LINK PRECTION EVALUATION MODELS

Recently Neural Network models are widely in NLP (Natural language Processing), image processing etc. The demands of KG completion task cannot be met by conventional methods. A NN technique was also added to KG completion in order to provide better and more efficient $\vec{\varepsilon}$ (#entities) and $\vec{r}$ (#relations) embeddings. So here mainly ConvE and ConvKB used for link prediction task and compares with TranE and DistMult. ConvE [2] is the first NN based model to complete KGs using a convolutional neural network (CNN).

Embedded 2D convolution is used by ConvE to forecast missing links in KG. When it comes to takes out feature interactions between entity embeddings and relation embeddings, 2D convolution performs better than 1D convolution. Through a layer of convolution and a fully linked layer, ConvE establishes local associations in several dimensions between entities. It's important to note that ConvE concentrates on the local interactions between various dimensional entries. The global links between identical dimensional elements of an embedding triple are not observed by ConvE.

Table 1. The Score Functions of Different Categories of Models

Models	Score Function	Categories of Models
ConvKB	$\emptyset(h,r,t) = concat(\varphi([h_v,r_v,t_v] * w))W$	Neural -network
ConvE	$\emptyset(x_s\ x_o) = \varphi(vec(\varphi(\langle \bar{x}_s \vert \bar{x}_r \rangle * w))W)x_o$	Neural- network
TransE	$\vert h_v + r_v - t_v \vert$	Translation distance
DistMult	$\langle h_v \vert r_v \vert t_v \rangle$	Semantic information

In this ConvE, the head and tail entity first embedded and then reshaped. After reshaping then it concatenated to 2D convolution layer's input matrix to produce a feature tensor. Matrix W is linearly transformed, the vectorized tensor, protrusion on the k-dimensional space, and then inner product gives the embedding of tail entity as same. The score function is given as

$$\emptyset(x_s\ x_o) = \varphi\big(vec\big(\varphi(\langle \bar{x}_s \vert \bar{x}_r \rangle * w)\big)W\big)x_o$$

Where $\bar{x}_s$ is 2D reshaping head entity embeddings and $\overline{x}_r$ is a 2D reshaping relation embeddings and $x_r \in R^k$ is relation parameter depending on x_r.The score ConvE provides is determined by an embedded 2D convolution, which is its most key characteristic.

In convKB, [1] it uses 1D convolution it has no any reshaping operation like ConvE. It represent each triple's in k -dimensional embedding. The triples $\langle h_v \vert r_v \vert t_v \rangle$ represented as a three column matrix $V = [h_v, r_v, t_v] \in R^{k\times 3}$. The matrix V is then pass though the convolution layer where filter of size 1×3 where the global relationship is extracted among the same direction embedding of input triple. To create various feature maps, on each and every row these filter is going to operate on each input matrix. ω is filter operated over aeach and every row of V to genrate a feature map of

$v = [v_1, v_2, \dots\dots\dots..v_n] \in R^k$ as

$$v_i = \mu(\omega \cdot V_i + b)$$

Where μ is activation function such as ReLU and b is bias. Feature maps and a triple feature vector are attached and calculates using dot product and weight vector w. The score function is given by

$$\emptyset(h,r,t) = concat\big(\varphi([h_v, r_v, t_v] * w)\big)W$$

DisMult [9] In DisMult, embeddings can be learned by a neural network. The first layer projects the pair of input entities to low dimensional vectors and the second layer combines two input vectors to a scalar for comparison via scoring function with relation-specific parameters. DisMult uses bilinear embeddings. Bilinear embeddings contain sufficient information which makes effective rule selection without looking at entities.

TransE [5] used a distance-based approach. This model determines relationships by treating them as translations and working with the entities' and relations' low-dimensional embeddings. It is a simple model that performs significantly well with fewer parameters in the KGC task. The model can be optimized efficiently with a stochastic gradient.

III. ART GRAPH DATASET

An artistic knowledge graph called ArtGraph [3] is based on DBpedia and WikiArt. The WikiArt metadata was retrieved and converted into nodes and associations that generally pertain to the artworks' genre, style, location, etc. WikiArt does not offer comprehensive information about artists; rather, each artist in our KG is linked to other nodes created using RDF triples taken from DBpedia in addition to the artworks they have made. Data extraction and integration from these two sources needed a time-consuming procedure of data cleaning and normalisation, as well as personal intervention to fix a number of data errors. The artist nodes and artwork nodes that make up ArtGraph's conceptual framework Artgraph consist of *artwork* nodes and *artist* node. Each piece of artwork has links to the following nodes: tags (such as "woman," "sea," and "birds"), genre (such as "self-portrait"), style, time period, series (such as Giuseppe Arcimboldo's "The Seasons"), auction, media (such as paper or watercolour), the gallery where the piece is displayed, and the city (or nation) where it was created.

Table 2. Experimental Dataset Statistics of Art Graph

#dataset	#entities	#relation	#_train	#_test
Art Graph	*18495*	*31*	*1174014*	*172054*

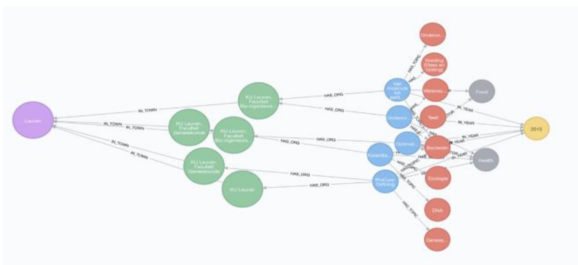

Figure 1: General Representation of Knowledge Graph

Retrieval Number: 100.1/ijitee.J926409111022
DOI: 10.35940/ijitee.J9264.13050424
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

IV. EXPERIMENTAL SETUP

A. Parameter Setting

ConvKB [1]: In this, Adam optimizer is use to train our model's parameters having different embedding size and different batch size. Learning rate is allotted as 0.001 and the activation function is used here is ReLu and the type of embedding space is used as low dimension vector space. Number of epoch is set at 50 and batch size is at 100. The training time required for ConvE model on above parameter setting is around 4-5 hrs. The size of triples is 1518149, futher it is bifurcate into train, valid and test, 1174014, 172054 and 172081 respectively.

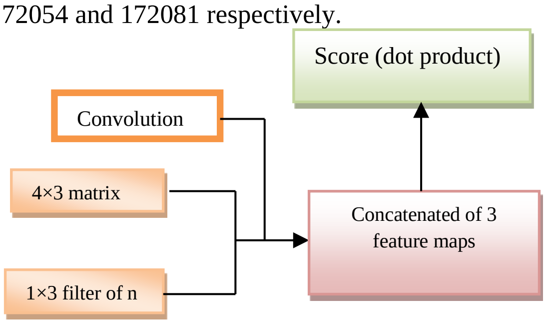

Figure 2: Process used in ConvKB [1] with the following parameters: activation function μ = ReLU, and number of filters = n.(4×3) matrix represent 4 embedding dimention and 3 denotes the $\boldsymbol{h_n}, \boldsymbol{r_n}, \boldsymbol{t_n}$.

ConvE [2]: For hyperparameters tuning in ConvE, here grid search technique is used. The embedding dropout, feature map dropout, projection layer dropout, embedding size and label smoothing hyperparameter ranges for the grid search were mentioned in base paper are same in our paper. In this model, batch size = 100 and 0.001 is the recommended learning rate is used. For optimization Adagrad is used. The entity embeddings must be made to have an L2 norm in order to regularise the ConvE model. The train test split is same as mentioned in above model.

TransE [5]: For TransE, 0.001 is the recommended learning rate is used. The optimizer used here is Stochastic Gradient Decent. The model is trained on different embedding size. The epoch size is of 50.

DistMult [9]: In DistMult, here, standard L2 regularisation is employed for the relation parameters. We put the batch size for this model at 100, and different trainings result in varied dimensionalities of the entity vector. There are 50 training epochs, and L2 regularisation is used.

B. Evaluation Metrics

Mean Rank (MR):

For each testing triplet, the mean rank—which is susceptible to outliers—is the average rank of the real truth entities.

$$MR = \frac{1}{|q|}\sum_{j=1}^{q} rank\ of(sub, pred, obj)$$

Mean Reciprocal Rank (MRR):

MRR determines the average inversely related rank of all real candidates and is a more reliable evaluation indicator.

$$MRR = \frac{1}{|q|}\sum_{j=1}^{q} \frac{1}{rank\ of(sub, pred, obj)}$$

Hit@*n* (*n* = 3, 5, 10):

Hits@n is the % of valid test triplets that place in the top n predictions is counted.

$$hit@n = \frac{1}{|q|}\sum_{j=1}^{q} 1\ if\ rank\ of(sub, pred, obj) \le n$$

V. RESULT AND DISCUSSION

Table.2 shows comparison between neural network model, translation distance model and semantic information-based models. Table.3 demonstrates that ConvKB achieves the highest Hits@10 scores and the best MR.

Table 3. Art Graph Experimental Results.

#Model	MR	MRR	#hit@n		
			3	5	10
ConvKB	3498	0.59	0.49	0.65	0.77
ConvE	3681	0.60	0.46	0.62	0.69
TranE	3173	0.41	0.41	0.47	0.51
DistMult	3788	0.41	0.41	0.55	0.64

The experimental findings of various KGE models are compared in Table 3. ConvKB depict the best MR & best hit@10 score on the ArtGrapgh Dataset. On experimental datasets, ConvKB outperforms the closely comparable model TransE. 3573 - 3173 = 400 is the gain in MR (improvement about 12.50%). (0.59 - 0.41) = 0.18 which is about 30.50 % and (0.77 - 0.51) = 0.26 which is about 33 %, in MRR and hit@10 respectively. ConvKB outperforms ConvE on the ArtGraph dataset (apart from MRR), demonstrating the value of taking transitional properties into consideration. ConvKB outperforms the closely comparable model ConvE also. **(**3681 – 3491) = 250 is the gain in MR (improvement about 7 %). (0.77 - 0.69) = 0.08 which is about 10 % in hit@10

VI. CONCLUSION AND FUTURE SCOPE

In this KGE work, the *ArtGraph* is tested for link prediction to improve its efficiency and effectiveness of knowledge graph representation learning using convolution neural network, translation and sematic model for downstream task for given query. In light of the research on this KG, the KGE models experimental findings, we can generally outline main point on how to make KGE models work better. This work demonstrates a possible scope for improving existing knowledge graph embedding's. This experiment was solely concerned with KGE link prediction; more research on knowledge graph completion issue, such as $\vec{\varepsilon}$ prediction, $\vec{\varepsilon}$ classification, and classification of triplet, is expected on this *ArtGraph*.

DECLARATION STATEMENT

Funding	No, I did not receive.
Conflicts of Interest	No conflicts of interest to the best of our knowledge.
Ethical Approval and Consent to Participate	No, the article does not require ethical approval and consent to participate with evidence.
Availability of Data and Material	Not relevant.
Authors Contributions	All authors have equal participation in this article.

REFERENCES

1. Nguyen, Dai Quoc, et al. "A novel embedding model for knowledge base completion based on convolutional neural network." *arXiv preprint arXiv:1712.02121* (2017). https://doi.org/10.18653/v1/N18-2053
2. Dettmers, Tim, et al. "Convolutional 2d knowledge graph embeddings." *Proceedings of the AAAI conference on artificial intelligence*. Vol. 32. No. 1. 2018. https://doi.org/10.1609/aaai.v32i1.11573
3. Castellano, Giovanna, Giovanni Sansaro, and Gennaro Vessio. "Integrating contextual knowledge to visual features for fine art classification." *arXiv preprint arXiv:2105.15028* (2021).
4. Wang, Meihong, Linling Qiu, and Xiaoli Wang. "A survey on knowledge graph embeddings for link prediction." *Symmetry* 13.3 (2021): 485. https://doi.org/10.3390/sym13030485
5. Bordes, Antoine, et al. "Translating embeddings for modeling multi-relational data." *Advances in neural information processing systems* 26 (2013).
6. Wang, R.; Wang, M.; Liu, J.; Chen, W.; Cochez, M.; Decker, S. Leveraging Knowledge Graph Embeddings for Natural Language Question Answering. In Proceedings of the DASFAA 2019, Chiang Mai, Thailand, 22–25 April 2019; pp. 659–675. https://doi.org/10.1007/978-3-030-18576-3_39
7. Lehmann, J.; Isele, R.; Jakob, M.; Jentzsch, A.; Kontokostas, D. Mendes, P. N.; Hellmann, S.; Morsey, M.; Kleef, P. V.; Auer, S.; et al. DBpedia—A Large-Scale, Multilingual Knowledge base Extracted from Wikipedia; Semantic Web, Springer, 2015; Volume 6, pp. 167–195. https://doi.org/10.3233/SW-140134
8. Mahdisoltani, Farzaneh, Joanna Biega, and Fabian Suchanek. "Yago3: A knowledge base from multilingual wikipedias." *7th biennial conference on innovative data systems research*. CIDR Conference, 2014.
9. Yang, Bishan, et al. "Embedding entities and relations for learning and inference in knowledge bases." *arXiv preprint arXiv:1412.6575* (2014).
10. Lin, Y.; Liu, Z.; Sun, M.; Liu, Y.; Zhu, X. Learning Entity and Relation Embeddings for Knowledge Graph Completion; AAAI Press: Palo Alto, CA, USA, 2015; pp. 2181–2187. https://doi.org/10.1609/aaai.v29i1.9491
11. Trouillon, T.; Welbl, J.; Riedel, S.; Gaussier, É.; Bouchard, G. Complex Embeddings for Simple Link Prediction; ICML: New York City,NY, USA, 2016; pp. 2071–2080.
12. Wang, Zhen, et al. "Knowledge graph embedding by translating on hyperplanes." *Proceedings of the AAAI conference on artificial intelligence*. Vol. 28. No. 1. 2014. https://doi.org/10.1609/aaai.v28i1.8870
13. Fan, Miao, et al. "Transition-based knowledge graph embedding with relational mapping properties." *Proceedings of the 28th Pacific Asia conference on language, information and computing*. 2014.
14. Kanaparthi, V. (2022). Examining Natural Language Processing Techniques in the Education and Healthcare Fields. In International Journal of Engineering and Advanced Technology (Vol. 12, Issue 2, pp. 8–18). https://doi.org/10.35940/ijeat.b3861.1212222
15. Arya, V., Khan, R., & Aggarwal, Prof. M. (2022). A Chatbot Application by using Natural Language Processing and Artificial Intelligence Markup Language. In International Journal of Soft Computing and Engineering (Vol. 12, Issue 3, pp. 1–7). https://doi.org/10.35940/ijsce.c3566.0712322
16. J, S., & Swamy, S. (2020). Modelling Simple and Efficient Data Transformation Scheme for Improving Natural Language Processing. In International Journal of Innovative Technology and Exploring Engineering (Vol. 9, Issue 3, pp. 1479–1485). https://doi.org/10.35940/ijitee.c8185.019320
17. Reddy, D. V., Padmaja, Dr. M., Kumar, K. M., Kiran, K. S., & Pramod, P. (2024). Chatbot Based Online Shopping Web Application. In Indian Journal of Data Communication and Networking (Vol. 3, Issue 4, pp. 7–14). https://doi.org/10.54105/ijdcn.b9782.03040623
18. Sharma, Dr. K., Garg, N., Pandey, A., Yadav, D., & Nikhil. (2021). Plagiarism Detection Technique using www and Wordnet. In Indian Journal of Artificial Intelligence and Neural Networking (Vol. 1, Issue 3, pp. 1–6). https://doi.org/10.54105/ijainn.b1015.061321