PADERBORN UNIVERSITY
Faculty for Computer Science, Electrical Engineering and Mathematics

Thesis Work Plan
—
Knowledge Graph Embedding Model Ensemble

by
Ana Alexandra Morim da Silva (Matr. no. 6845223)

submitted to
Prof. Dr. Axel-Cyrille Ngonga Ngomo

Paderborn, July 19, 2021

Master Thesis Work Plan by Ana Alexandra Morim da Silva (Matr. no. 6845223)

Supervisor: Prof. Dr. Axel-Cyrille Ngonga Ngomo

First examiner: Prof. Dr. Axel-Cyrille Ngonga Ngomo

Contents

1 Work plan

1.1 Introduction

Language, as the cornerstone of human communication, is composed of individually meaningful words, able of conveying one's thoughts and emotions (Rabiah, 2018). Its inherent partialness dictates the words' meaning, as the connotation is specific to a context or to priorly acquired knowledge. As language is vital to knowledge sharing, its persistence also becomes indispensable.

One of the preferred ways of language persistence is in text-form. This is noticeable in the quantity of text data available on the Web. However, this data is unstructured and therefore, the data's possible applications do not reach its full potential. One of the data-models able to represent information within a text in a structured manner are knowledge graphs (KG). A knowledge graph structures data in the form of a directed labeled graph, where the nodes and edges represent entities and relations, respectively (Ji et al., 2021).

However, as the amount of data is increasing over time, KGs are deemed as incomplete (Wang et al., 2017). Predicting missing facts in a KG is the task of link prediction. Knowledge graph embedding (KGE) models are one of the methods able of solving the link prediction problem. A KGE aims to map entities and relations of a KG in a low-dimensional vector space while capturing the KG structure (Ji et al., 2021).

There exist multiple KGE models, each aiming to capture a different set of properties from the input data. Each KGE model can be described by the representation space it acts on, e. g., point-wise, complex vector space, and its score function (Ji et al., 2021). Expectedly, different techniques cause the embeddings to embody a meaning specific to the model. While there exist plenty of KGE models, each model has its own strengths and weaknesses. Hence, an opportunity arises from the contrasting models.

This thesis acts on this opportunity. Through the combination of the different models' embeddings, the creation of an improved embedding and consequent reconstruction is facilitated. An advantage would consist of the production of a condensed, enriched and therefore meaningful embedding, that could prove useful in the same tasks as its original counterparts. As the number of dimensions of the internal vector is smaller than the input vectors', this would also translate into an improved management of computing resources in the long run. This thesis will analyze the designed system's capabilities by comparing the internal and the reconstructed vectors' performance ability in a link prediction task.

1.2 Explicit formulation of the target setting

This section starts by defining some key concepts to this thesis, followed by a general description of the work to be conducted throughout this thesis.

1.2.1 Link prediction

A knowledge graph can be defined as $\mathcal{G} = \{(h, r, t)\} \subseteq \mathcal{E} \times \mathcal{R} \times \mathcal{E}$. It is a set of triples, each with a head and tail entity $h, t \in \mathcal{E}$ connected by a relation $r \in \mathcal{R}$. $\mathcal{E}$ and $\mathcal{R}$ represent the set of entities and relations, respectively. The link prediction problem in a KG can thus be defined as the task of predicting the missing entity in triples of the form $(h, r, ?)$ or $(?, r, t)$. This can be achieved through learning a score function $\delta : \mathcal{E} \times \mathcal{R} \times \mathcal{E} \mapsto \mathbb{R}$, indicative of the triple's existence in the KG (Dettmers et al., 2018). Link prediction is a widespread evaluation task used by KGE models (Ji et al., 2021).

1.2.2 Knowledge graph embedding models

The objective of a knowledge graph embedding model is to map entities and relations to a low dimensional vector space. According to Ji et al. (2021), there exist some key properties that allow for an easier discerning of KGE models: 1) the representation space of the entities and relations, e. g., point-wise space, manifold, complex vector space and 2) a score function, δ.

The score function is usually a distance or a semantic-based function. As the name implies, distance-based score functions measure the distance between the head and tail entity in accordance to the adopted representation space. On the other hand, semantic-based functions rely on semantic matching.

Below are some KGE models, representative of the chosen representation space and type of score function.

ManifoldE (Xiao et al., 2016a) is a manifold-based embedding model. Given a head entity and a relation, the tail entity is projected to a high-dimension manifold, e. g., spheres or hyperplanes. Let D_r be a relation's manifold parameter and $\mathcal{M}$ the manifold function $\mathcal{M} : \mathcal{E} \times \mathcal{R} \times \mathcal{E} \mapsto \mathbb{R}$.

$$\delta_{ManifoldE} = ||\mathcal{M}(h, r, t) - D_r^2||^2 \tag{1.1}$$

TranSparse (Ji et al., 2016) is a real-valued translation-based model, i. e., a relation r is regarded as a translation from a head h to a tail entity t. The model utilizes two sparse transfer matrices for each relation, $M_r^h(\theta_r^h)$ and $M_r^t(\theta_r^t)$ with a sparse degree θ_r, depending on whether it will be applied to a head or a tail entity.

$$\delta_{TranSparse} = -||M_r^h(\theta_r^h)h + r - M_r^t(\theta_r^t)t||_{1/2}^2 \tag{1.2}$$

TransG (Xiao et al., 2016b) is also a real-valued translation-based model. The model proposes that a single relation vector might not be enough, as relations may have multiple meanings depending on the entities. With the intent of solving this issue, TransG uses a Bayesian non-parametric infinite mixture embedding model. Each relation is represented by a mixture of component vectors. Let the mean vectors $u_h, u_t \sim \mathcal{N}(0,1)$, the variance be denoted as var, the mean vector $u_r = u_t - u_h$, the semantic component $\pi_r \sim CRP(\beta)$[1] parameterized by β and I_r is the number of semantic components of each relation.

$$\delta_{TransG} = \sum_{i}^{I_r} \pi_r^i exp\left(-\frac{||u_h + u_r^i - u_t||_2^2}{var_h + var_t}\right) \tag{1.3}$$

TorusE (Ebisu and Ichise, 2018) is a translational embedding model where entities and relations are projected from vector space to tori.

$$\delta_{TorusE} = min_{(x,y)\in([h]+[r])\times[t]}||x-y||_i \tag{1.4}$$

ConvE (Dettmers et al., 2018) is a real-valued embedding model that uses two-dimensional convolutions over the entity and relations embeddings. Let w be the convolutional filters, vec the vectorization operation, σ the sigmoid activation function, N the two-dimensional representation of the embedding and W the linear transformation parameters.

$$\delta_{ConvE} = \sigma(vec(\sigma([N_h; N_r] * w))W)t \tag{1.5}$$

DensE (Lu and Hu, 2020) is a translation-based model, based on scaling and rotations of vectors in a three-dimensional euclidean space. Relations are modelled as rotations from the head to the tail entity, followed by a scaling factor. Hence, entities and relations are represented by quaternions. Let $\mathcal{O}$ be an operator applied on an entity such that $e(t)_i = \mathcal{O}(e(r)_i)e(h)_i$.

$$\delta_{DensE} = -\frac{1}{2}\left(|\mathcal{O}(r)h - t| + |\mathcal{O}(r^{-1}t) - h|\right) \tag{1.6}$$

AttH (Chami et al., 2020) is a hyperbolic-based embedding model with a Poincaré ball model. Relations are deemed as rotations in $\mathbb{B}$, supplemented with a reflection operation applied over the head entity. The two operations are combined through hyperbolic attention, Att. Let $d_{\mathbb{B}}$ denote the hyperbolic distance, $\oplus$ the Möbius addition and b the entity biases.

$$\delta_{AttH} = -d_{\mathbb{B}}\left(Att(q^H_{Rot}, q^H_{Ref}; a_r) \oplus r^H, t^H\right)^2 + b_h + b_t \tag{1.7}$$

LowFER (Amin et al., 2020) is a real-valued embedding model. LowFER proposes a multi-modal factorized bilinear pooling mechanism that improves the bilinear product computation between the relation and the head entity. Let $S^k_{i,j}$ denote a binary matrix

[1] Chinese Restaurant Process

with $S^k_{i,j} = 1, \forall j \in [(i-1)k+1, ik]$, U and V the factorization of a bilinear map $W \in \mathbb{R}^{a\times b}$ with $U \in \mathbb{R}^{a\times k}$ and $V \in \mathbb{R}^{b\times k}$, $diag$ the corresponding symmetric diagonal matrix and k the factorization rank.

$$\delta_{LowFER} = \left(S^k diag(U^T h)V^T r\right)^T t \tag{1.8}$$

ConEx (Demir and Ngomo, 2021) is a complex-valued embedding model that leverages two-dimensional convolutions. The convolutional layer is followed by an affine transformation in $\mathbb{C}$. Let Z and c define the affine transformation, Re the real part of a complex number, ReLU the rectified linear unit function and w the convolution kernels.

$$\delta_{ConEx} = Re(\langle conv(h, r), h, r, \bar{t}\rangle) \tag{1.9}$$

with $conv(h, r)$ defined as

$$conv(h, r) = ReLU\left(vec(ReLU([h, r] * w)) \cdot Z + c\right) \tag{1.10}$$

1.2.3 Unsupervised learning

Machine learning techniques typically generate inferences from a dataset whilst minimizing an error function or optimizing a score function. In unsupervised learning, these functions do not depend on a given ground-truth value (Sindhu Meena and Suriya, 2020), but rely on mimicry or pattern recognition instead.

An unsupervised learning technique, relevant to this thesis, is the autoencoder. Autoencoders (Rumelhart et al., 1986) aim to encode the input to a meaningful representation and to reconstruct their own input. Thus, an autoencoder requires an encoding function $e : \mathbb{I}^n \mapsto \mathbb{O}^m$, a decoding function $d : \mathbb{O}^m \mapsto \mathbb{I}^n$ and a reconstruction loss function $\mathcal{L}$, that measures the similarity between the reconstructed and the input data v. Let $\mathbb{I}$ and $\mathbb{O}$ denote sets. The encoder and decoder are typically a neural network and trained on the minimization of the reconstruction loss, as seen in Equation 1.11.

$$e, d = \arg\min_{e,d} \mathcal{L}(d \circ e(v), v) \tag{1.11}$$

1.2.4 General description of the approach

The objective of this thesis aligns with that of an autoencoder. Leveraging an autoencoder's architecture, this thesis' goal becomes two-fold: 1) encode k n-dimensional embeddings into an m-dimensional vector w_m such that $e(v^n_k) = w^m$ with $m << nk$, and 2) decode w_m back into k n-dimensional embeddings such that $d(w_m) \sim v^k_n$, as depicted in Figure 1.1.

The evaluation will be performed on the three sets of embeddings involved, i. e., the original embeddings, the internal vector and the reconstructed embeddings. The performance

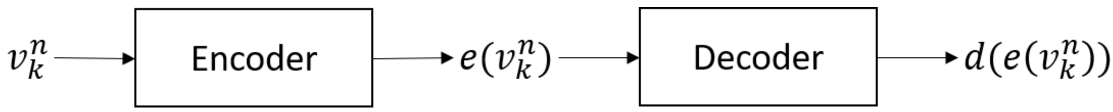

Figure 1.1: General idea of the desired solution, v_k^n represents the k n-dimensional input embedding models.

of the internal vector and the reconstructed embeddings will be compared to that of the original embeddings in a link prediction task. This will be extended not only to the designed system, but possibly also to vector concatenation, to further assess the competency of the designed system against a baseline.

To facilitate the result interpretation, the datasets used during the evaluation should be datasets typically used in KGE-models-related work, e. g., WN18RR (Dettmers et al., 2018), FB15k-237 (Toutanova and Chen, 2015) and NELL (Carlson et al., 2010). The KGE models mentioned in 1.2.2 constitute some of the representative models and could be used in this thesis.

1.3 Task description

The enumerated tasks below (Task 1 – 6) correspond to the schedule and description of the tasks, required for the timely completion of this thesis.

Task 1	**State of the art review**
Time Required	3 Weeks
Description	During the state of the art review, the existing solutions will be researched.

Task 2	**Solution design**
Time Required	3 Weeks
Description	During the solution design stage, the approach will be developed in more detail based on the state of the art review.

Task 3	**KGE generation**
Time Required	4 Weeks
Description	As the KGE models are independent from the designed system in the thesis, the embeddings generation will run in parallel to other tasks as soon as the models decision has been made.

Task 4	Solution implementation
Time Required	8 Weeks
Description	The previously designed solution will be implemented in this stage.

Task 5	Experiments
Time Required	4 Weeks
Description	The experiments will be conducted in this stage.

Task 6	Thesis writing
Time Required	8 Weeks
Description	The thesis will be written in three intervals: the first will be dedicated to the background and related work, the second to the approach and implementation and the third to the results and conclusion chapters. The last interval will also be used to wrap up any pending writing tasks.

1.4 Work Schedule

In Figure 1.2, the timeline of the proposed tasks described in Section 1.3 is depicted. The work schedule is reliant on a five months period. There will be three main stages. The initial phase will serve as a preparatory stage where the state-of-the-art approaches will be researched and the proposed solution details will be discussed and agreed on. The introductory phase is then followed by the implementation stage, where the proposed solution's main work will be carried out. The experimental phase will conclude the thesis' work. In this stage, the designed implementation will be tested, the results consolidated and conclusions therefrom drawn. The thesis writing will be conducted in parallel to other tasks and as a finalizing stage.

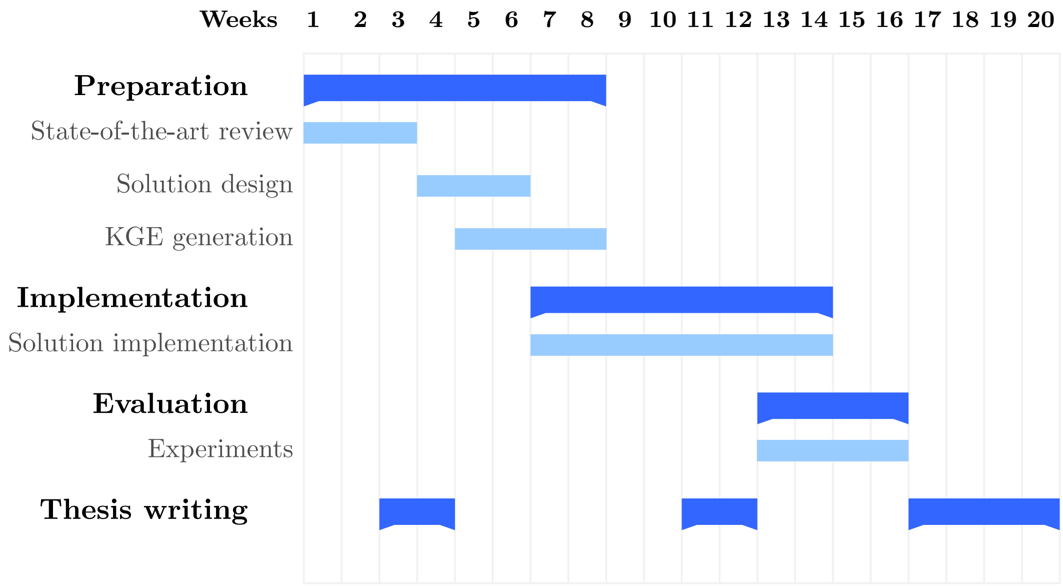

Figure 1.2: Gantt chart of the proposed tasks in Section 1.3.

1.5 Provisional Thesis Structure

The thesis will comprise the following chapters:

1. Introduction: The topic is described taking into consideration the general and particular context in which it is inserted in, as well as the topic's motivation.
2. Related work: The existing techniques and approaches will be described in this section.
3. Background: The foundational knowledge required to understand this thesis will be described here.
4. Approach: The designed system will be described here.
5. Implementation: The implementation details will be described here.
6. Evaluation: The experiments setup, requirements and results will be described here.
7. Summary: The work will be summarised in the final chapter, highlighting the key findings during the evaluation stage. Future work will also be mentioned here.

Bibliography

Saadullah Amin, Stalin Varanasi, Katherine Ann Dunfield, and Günter Neumann. Lowfer: Low-rank bilinear pooling for link prediction. In *International Conference on Machine Learning*, pages 257–268. PMLR, 2020.

Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka, and Tom M. Mitchell. Toward an architecture for never-ending language learning. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, AAAI'10, page 1306–1313. AAAI Press, 2010.

Ines Chami, Adva Wolf, Da-Cheng Juan, Frederic Sala, Sujith Ravi, and Christopher Ré. Low-dimensional hyperbolic knowledge graph embeddings. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6901–6914, 2020.

Caglar Demir and Axel-Cyrille Ngonga Ngomo. Convolutional complex knowledge graph embeddings. In *Eighteenth Extended Semantic Web Conference - Research Track*, 2021. URL https://openreview.net/forum?id=6T45-4TFqaX.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. Convolutional 2d knowledge graph embeddings. In *AAAI*, pages 1811–1818, 2018. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17366.

Takuma Ebisu and Ryutaro Ichise. Toruse: Knowledge graph embedding on a lie group. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, 2018.

Guoliang Ji, Kang Liu, Shizhu He, and Jun Zhao. Knowledge graph completion with adaptive sparse transfer matrix. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, page 985–991. AAAI Press, 2016.

Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and Philip Yu. A survey on knowledge graphs: Representation, acquisition, and applications. volume PP, 04 2021. doi: 10.1109/TNNLS.2021.3070843.

Haonan Lu and Hailin Hu. Dense: An enhanced non-abelian group representation for knowledge graph embedding. volume abs/2008.04548, 2020. URL https://arxiv.org/abs/2008.04548.

Sitti Rabiah. Language as a tool for communication and cultural reality discloser. INA-Rxiv, Nov 2018. doi: 10.31227/osf.io/nw94m. URL osf.io/preprints/inarxiv/nw94m.

David E. Rumelhart, James L. McClelland, and CORPORATE PDP Research Group, editors. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations.* MIT Press, Cambridge, MA, USA, 1986. ISBN 026268053X.

K. Sindhu Meena and S. Suriya. A survey on supervised and unsupervised learning techniques. In L. Ashok Kumar, L. S. Jayashree, and R. Manimegalai, editors, *Proceedings of International Conference on Artificial Intelligence, Smart Grid and Smart City Applications*, pages 627–644, Cham, 2020. Springer International Publishing. doi: 10.1007/978-3-030-24051-6_58.

Kristina Toutanova and Danqi Chen. Observed versus latent features for knowledge base and text inference. In *Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality*, pages 57–66, Beijing, China, July 2015. Association for Computational Linguistics. doi: 10.18653/v1/W15-4007. URL https://aclanthology.org/W15-4007.

Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. Knowledge graph embedding: A survey of approaches and applications. volume 29, pages 2724–2743, 2017. doi: 10.1109/TKDE.2017.2754499.

Han Xiao, Minlie Huang, and Xiaoyan Zhu. From one point to a manifold: Knowledge graph embedding for precise link prediction. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, page 1315–1321. AAAI Press, 2016a. ISBN 9781577357704.

Han Xiao, Minlie Huang, and Xiaoyan Zhu. TransG : A generative model for knowledge graph embedding. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2316–2325, Berlin, Germany, August 2016b. Association for Computational Linguistics. doi: 10.18653/v1/P16-1219. URL https://aclanthology.org/P16-1219.