LiteralE: Incorporating Literals into Knowledge Graph Embeddings

A. Kristiadi[1], A. Khan[1], **D. Lukovnikov**, J. Lehmann, A. Fischer

October 28, 2019

UNIVERSITÄT BONN
Rheinische Friedrich-Wilhelms-Universität Bonn

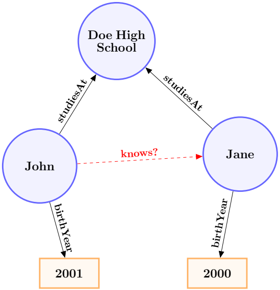
[1]Equal Contribution

Outline

UNIVERSITÄT BONN
Rheinische Friedrich-Wilhelms-Universität Bonn

Literals for link prediction

Literals encode information that cannot be represented by relations alone, and are useful for link prediction task.



UNIVERSITÄT BONN
Rheinische Friedrich-Wilhelms-Universität Bonn

Background

UNIVERSITÄT BONN
Rheinische
Friedrich-Wilhelms-
Universität Bonn

Link Prediction

- A knowledge graph $\mathcal{G}$ is a subset of $(\mathcal{E} \times \mathcal{E} \times \mathcal{R}) \cup (\mathcal{E} \times \mathcal{L} \times \mathcal{D})$ representing the facts that are assumed to hold.
  where $\mathcal{E} = \{e_1, \cdots, e_{N_e}\}$ is the set of entities,
  $\mathcal{R} = \{r_1, \cdots, r_{N_r}\}$ is the set of relations connecting two entities,
  $\mathcal{D} = \{d_1, \cdots, d_{N_d}\}$ is the set of relations connecting an entity and a literal, and
  $\mathcal{L}$ is the set of all literal values.

UNIVERSITÄT BONN
Rheinische
Friedrich-Wilhelms-
Universität Bonn

Link Prediction

- A knowledge graph $\mathcal{G}$ is a subset of $(\mathcal{E} \times \mathcal{E} \times \mathcal{R}) \cup (\mathcal{E} \times \mathcal{L} \times \mathcal{D})$ representing the facts that are assumed to hold.
  where $\mathcal{E} = \{e_1, \cdots, e_{N_e}\}$ is the set of entities,
  $\mathcal{R} = \{r_1, \cdots, r_{N_r}\}$ is the set of relations connecting two entities,
  $\mathcal{D} = \{d_1, \cdots, d_{N_d}\}$ is the set of relations connecting an entity and a literal, and
  $\mathcal{L}$ is the set of all literal values.
- The link prediction is defined as the task of deciding whether a fact $(e_i, e_j, r_k) \in \mathcal{E} \times \mathcal{E} \times \mathcal{R}$ is true or false.

UNIVERSITÄT BONN
Rheinische Friedrich-Wilhelms-Universität Bonn

Link Prediction

- A knowledge graph $\mathcal{G}$ is a subset of $(\mathcal{E} \times \mathcal{E} \times \mathcal{R}) \cup (\mathcal{E} \times \mathcal{L} \times \mathcal{D})$ representing the facts that are assumed to hold.
  where $\mathcal{E} = \{e_1, \cdots, e_{N_e}\}$ is the set of entities,
  $\mathcal{R} = \{r_1, \cdots, r_{N_r}\}$ is the set of relations connecting two entities,
  $\mathcal{D} = \{d_1, \cdots, d_{N_d}\}$ is the set of relations connecting an entity and a literal, and
  $\mathcal{L}$ is the set of all literal values.
- The link prediction is defined as the task of deciding whether a fact $(e_i, e_j, r_k) \in \mathcal{E} \times \mathcal{E} \times \mathcal{R}$ is true or false.
- Formally, each possible (e_i, e_j, r_k) is mapped to a score value $\psi(e_i, e_j, r_k)$ under certain transformation $\psi : \mathcal{E} \times \mathcal{E} \times \mathcal{R} \rightarrow \mathbb{R}$, where a higher score implies the triple is more likely to be true.

UNIVERSITÄT BONN

Rheinische Friedrich-Wilhelms-Universität Bonn

Latent Feature Methods

- $\psi(e_i, e_j, r_k) \stackrel{\text{def}}{=} f(\mathbf{e}_i, \mathbf{e}_j, \mathbf{r}_k)$.

Latent Feature Methods

- $\psi(e_i, e_j, r_k) \stackrel{\text{def}}{=} f(\mathbf{e}_i, \mathbf{e}_j, \mathbf{r}_k)$.
- $f : \mathbb{R}^H \times \mathbb{R}^H \times \mathbb{R}^H \to \mathbb{R}$ computes a score of triple $(\mathbf{e}_i, \mathbf{e}_j, \mathbf{r}_k)$ that correlates with the truth value of the triple.

UNIVERSITÄT BONN

Rheinische Friedrich-Wilhelms-Universität Bonn

Latent Feature Methods

- $\psi(e_i, e_j, r_k) \stackrel{\text{def}}{=} f(\mathbf{e}_i, \mathbf{e}_j, \mathbf{r}_k)$.
- $f : \mathbb{R}^H \times \mathbb{R}^H \times \mathbb{R}^H \to \mathbb{R}$ computes a score of triple $(\mathbf{e}_i, \mathbf{e}_j, \mathbf{r}_k)$ that correlates with the truth value of the triple.
- Latent Feature Methods are a class of methods defined to learn low dimensional vector representations of entities and relations, called *embeddings* or *latent features*.

Latent Feature Methods - Models

- DistMult [YYH+15]:

$$f_{\text{DistMult}}(\mathbf{e}_i, \mathbf{e}_j, \mathbf{r}_k) = \langle \mathbf{e}_i, \mathbf{e}_j, \mathbf{r}_k \rangle = \mathbf{e}_i^\top \operatorname{diag}(\mathbf{r_k})\, \mathbf{e}_j \quad ,$$

Latent Feature Methods - Models

- DistMult [YYH+15]:

$$f_{\text{DistMult}}(\mathbf{e}_i, \mathbf{e}_j, \mathbf{r}_k) = \langle \mathbf{e}_i, \mathbf{e}_j, \mathbf{r}_k \rangle = \mathbf{e}_i^T \operatorname{diag}(\mathbf{r_k})\, \mathbf{e}_j \quad ,$$

- ComplEx [TWR+16]:

$$\begin{aligned} f_{\text{ComplEx}}(\mathbf{e}_i, \mathbf{e}_j, \mathbf{r}_k) &= \operatorname{Re}(\langle \mathbf{e}_i, \bar{\mathbf{e}}_j, \mathbf{r}_k \rangle) \\ &= \langle \operatorname{Re}(\mathbf{e}_i), \operatorname{Re}(\mathbf{e}_j), \operatorname{Re}(\mathbf{r}_k) \rangle \\ &+ \langle \operatorname{Im}(\mathbf{e}_i), \operatorname{Im}(\mathbf{e}_j), \operatorname{Re}(\mathbf{r}_k) \rangle \\ &+ \langle \operatorname{Re}(\mathbf{e}_i), \operatorname{Im}(\mathbf{e}_j), \operatorname{Im}(\mathbf{r}_k) \rangle \\ &- \langle \operatorname{Im}(\mathbf{e}_i), \operatorname{Re}(\mathbf{e}_j), \operatorname{Im}(\mathbf{r}_k) \rangle \quad , \end{aligned}$$

UNIVERSITÄT BONN

Rheinische Friedrich-Wilhelms-Universität Bonn

Latent Feature Methods - Models

- DistMult [YYH+15]:

$$f_{\text{DistMult}}(\mathbf{e}_i, \mathbf{e}_j, \mathbf{r}_k) = \langle \mathbf{e}_i, \mathbf{e}_j, \mathbf{r}_k \rangle = \mathbf{e}_i^\top \operatorname{diag}(\mathbf{r_k})\, \mathbf{e}_j \quad ,$$

- ComplEx [TWR+16]:

$$\begin{aligned} f_{\text{ComplEx}}(\mathbf{e}_i, \mathbf{e}_j, \mathbf{r}_k) &= \operatorname{Re}(\langle \mathbf{e}_i, \bar{\mathbf{e}}_j, \mathbf{r}_k \rangle) \\ &= \langle \operatorname{Re}(\mathbf{e}_i), \operatorname{Re}(\mathbf{e}_j), \operatorname{Re}(\mathbf{r}_k) \rangle \\ &+ \langle \operatorname{Im}(\mathbf{e}_i), \operatorname{Im}(\mathbf{e}_j), \operatorname{Re}(\mathbf{r}_k) \rangle \\ &+ \langle \operatorname{Re}(\mathbf{e}_i), \operatorname{Im}(\mathbf{e}_j), \operatorname{Im}(\mathbf{r}_k) \rangle \\ &- \langle \operatorname{Im}(\mathbf{e}_i), \operatorname{Re}(\mathbf{e}_j), \operatorname{Im}(\mathbf{r}_k) \rangle \quad , \end{aligned}$$

- ConvE [DPPR18]:

$$f_{\text{ConvE}}(\mathbf{e}_i, \mathbf{e}_j, \mathbf{r}_k) = h(\operatorname{vec}(h([\mathbf{e}_i, \mathbf{r}_k] * \omega))\mathbf{W})\, \mathbf{e}_j \quad .$$

LiteralE

UNIVERSITÄT BONN
Rheinische
Friedrich-Wilhelms-
Universität Bonn

Our Contribution

- We propose LiteralE, a universal approach to enrich latent feature methods with literal information via a learnable parametric function.

UNIVERSITÄT BONN
Rheinische Friedrich-Wilhelms-Universität Bonn

[2] A literal-extended version of YAGO3-10 is provided by Pezeshkpou *et al.* [PICS17]

Our Contribution

- We propose LiteralE, a universal approach to enrich latent feature methods with literal information via a learnable parametric function.
- We evaluate LiteralE on standard link prediction datasets: FB15k, FB15k-237 and YAGO3-10. We provide literal-extended version of FB15k and FB15k-237 datasets.[2]

UNIVERSITÄT BONN
Rheinische Friedrich-Wilhelms-Universität Bonn

[2] A literal-extended version of YAGO3-10 is provided by Pezeshkpou *et al.* [PICS17]

Our Contribution

- We propose LiteralE, a universal approach to enrich latent feature methods with literal information via a learnable parametric function.
- We evaluate LiteralE on standard link prediction datasets: FB15k, FB15k-237 and YAGO3-10. We provide literal-extended version of FB15k and FB15k-237 datasets.[2]
- We empirically show that exploiting the information provided by literals significantly increases the link prediction performance of existing latent feature methods as well as the quality of their embeddings.

[2] A literal-extended version of YAGO3-10 is provided by Pezeshkpou *et al.* [PICS17]

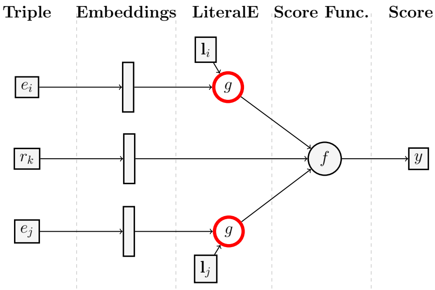
LiteralE - entity and literal vectors

- $\mathbf{e_i} \in \mathbb{R}^H$: "vanilla" embedding for entity i — trainable parameters
- $\mathbf{l_i} \in \mathbb{R}^{N_d}$: the "literal" vector for the entity i — entity attribute values (not trainable)
  - N_d: number of literal relations in KG
  - zeros for non-specified literals
  - example:
    - four data relations in total in KG: (1) height, (2) birth year, (3) surface area, (4) number of floors
    - $\mathbf{l}$ for a person: $[1.80, 1990, 0, 0]^T$
    - $\mathbf{l}$ for NZ: $[0, 0, 268000, 0]^T$

UNIVERSITÄT BONN
Rheinische Friedrich-Wilhelms-Universität Bonn

LiteralE

LiteralE uses a parameterized function $g : \mathbb{R}^H \times \mathbb{R}^{N_d} \rightarrow \mathbb{R}^H$ that learns to project entity embeddings and literal vectors for better entity representations.



Replace the score function $f_X(\mathbf{e}_i, \mathbf{e}_j, \mathbf{r}_k)$ with

$$f_X(g(\mathbf{e}_i, \mathbf{l}_i), g(\mathbf{e}_j, \mathbf{l}_j), \mathbf{r}_k)$$

LiteralE

- (Linear version of g):

$$g_{\text{lin}}(\mathbf{e}_i, \mathbf{l}_i) = \mathbf{W}^\top [\mathbf{e}_i, \mathbf{l}_i] \ ,$$

UNIVERSITÄT BONN
Rheinische Friedrich-Wilhelms-Universität Bonn

LiteralE

- (Linear version of g):

$$g_{\text{lin}}(\mathbf{e}_i, \mathbf{l}_i) = \mathbf{W}^\mathsf{T}[\mathbf{e}_i, \mathbf{l}_i] \ ,$$

- We define g for numerical literals as:

$$\begin{aligned} g &: \mathbb{R}^H \times \mathbb{R}^{N_d} \to \mathbb{R}^H \\ \mathbf{e}, \mathbf{l} &\mapsto \mathbf{z} \odot \mathbf{h} + (1 - \mathbf{z}) \odot \mathbf{e} \,, \end{aligned} \tag{1}$$

$$\begin{aligned} \mathbf{z} &= \sigma(\mathbf{W}_{ze}^\mathsf{T} \mathbf{e} + \mathbf{W}_{zl}^\mathsf{T} \mathbf{l} + \mathbf{b}) \\ \mathbf{h} &= h(\mathbf{W}_h^\mathsf{T} [\mathbf{e}, \mathbf{l}]) \,. \end{aligned} \tag{2}$$

UNIVERSITÄT BONN
Rheinische Friedrich-Wilhelms-Universität Bonn

LiteralE

- Extend to text literals. We use entity descriptions from freebase.
- Encode text description of an entity as latent representation $t \in \mathbb{R}^{N_t}$.[3]
- Extend previous equations:

$$
\begin{aligned}
g &: \mathbb{R}^H \times \mathbb{R}^{N_d} \times \mathbb{R}^{N_t} \rightarrow \mathbb{R}^H \\
&\mathbf{e}, \mathbf{l}, \mathbf{t} \mapsto \mathbf{z} \odot \mathbf{h} + (1 - \mathbf{z}) \odot \mathbf{e}\,,
\end{aligned} \tag{3}
$$

$$
\begin{aligned}
\mathbf{z} &= \sigma(\mathbf{W}_{ze}^{\mathrm{T}} \mathbf{e} + \mathbf{W}_{zl}^{\mathrm{T}} \mathbf{l} + \mathbf{W}_{zt}^{\mathrm{T}} \mathbf{t} + \mathbf{b}) \\
\mathbf{h} &= h(\mathbf{W}_h^{\mathrm{T}} [\mathbf{e}, \mathbf{l}, \mathbf{t}])\,.
\end{aligned} \tag{4}
$$

UNIVERSITÄT BONN Rheinische Friedrich-Wilhelms-Universität Bonn

[3] We use GloVe word embeddings provided by Spacy.

Experimental Setup & Results

Dataset

We use three widely used datasets for evaluating link prediction performance: FB15k, FB15k-237, and YAGO3-10.

Dataset

We use three widely used datasets for evaluating link prediction performance: FB15k, FB15k-237, and YAGO3-10.

Dataset	**FB15k**	**FB15k-237**	**YAGO3-10**
# Entities (N_e)	14,951	14,541	123,182
# Relations (N_r)	1,345	237	37
# Data rel. (N_d)	121	121	5
# Literals ($\|\mathcal{L}\|$)	18,741	18,741	111,406
# Relational triples	592,213	310,116	1,089,040
# Literal triples	70,257	70,257	111,406

UNIVERSITÄT BONN
Rheinische
Friedrich-Wilhelms-
Universität Bonn

Training

- DistMult, ComplEx and ConvE as base models for LiteralE experiments.

Training

- DistMult, ComplEx and ConvE as base models for LiteralE experiments.
- Loss: BCE between the probability vector $\mathbf{p} \in [0,1]^{N_e}$ and the ground truth labels $\mathbf{y} \in \{0,1\}^{N_e}$:
    - $y_j = 1 \iff$ triple (e_i, e_j, r_k) exists in the KG
    - N_e is the number of entities
    - $p_j = \sigma(f_X(\cdot))$

$$L(\mathbf{p}, \mathbf{y}) = -\frac{1}{N_e} \sum_{j=1}^{N_e} (y_j \log(p_j) + (1 - y_j) \log(1 - p_j)) \ ,$$

UNIVERSITÄT BONN
Rheinische Friedrich-Wilhelms-Universität Bonn

Training

- DistMult, ComplEx and ConvE as base models for LiteralE experiments.
- Loss: BCE between the probability vector $\mathbf{p} \in [0,1]^{N_e}$ and the ground truth labels $\mathbf{y} \in \{0,1\}^{N_e}$:
  - $y_j = 1 \iff$ triple (e_i, e_j, r_k) exists in the KG
  - N_e is the number of entities
  - $p_j = \sigma(f_X(\cdot))$

$$L(\mathbf{p}, \mathbf{y}) = -\frac{1}{N_e} \sum_{j=1}^{N_e} (y_j \log(p_j) + (1 - y_j) \log(1 - p_j)) \ ,$$

- Adam [KB15] to optimize this loss function.

UNIVERSITÄT BONN
Rheinische
Friedrich-Wilhelms-
Universität Bonn

Evaluation Measures

To evaluate the model, we rank all triples with respect to their scores and use the following standard evaluation measures:

1. Mean Rank (MR)
2. Mean Reciprocal Rank (MRR)
3. Hits@1, Hits@3, and Hits@10

Results

FB15k					
Models	**MR**	**MRR**	**Hits@1**	**Hits@3**	**Hits@10**
DistMult	108	0.671	0.589	0.723	0.818
ComplEx	127	0.695	0.618	0.744	0.833
ConvE	49	0.692	0.596	0.760	0.853
KBLN [GDN17]	129	0.739	0.668	**0.788**	0.859
MTKGNN [TTPH17]	87	0.669	0.586	0.722	0.82
DistMult-LiteralE	68	0.676	0.589	0.733	0.825
ComplEx-LiteralE	80	**0.746**	**0.686**	0.782	0.853
ConvE-LiteralE	**43**	0.733	0.656	0.785	**0.863**

UNIVERSITÄT BONN
Rheinische Friedrich-Wilhelms-Universität Bonn

Results

FB15k-237					
Models	**MR**	**MRR**	**Hits@1**	**Hits@3**	**Hits@10**
DistMult	633	0.282	0.203	0.309	0.438
ComplEx	652	0.290	0.212	0.317	0.445
ConvE	297	0.313	0.228	0.344	0.479
KBLN [GDN17]	358	0.301	0.215	0.333	0.468
MTKGNN [TTPH17]	532	0.285	0.204	0.312	0.445
DistMult-LiteralE	280	**0.317**	**0.232**	**0.348**	**0.483**
ComplEx-LiteralE	357	0.305	0.222	0.336	0.466
ConvE-LiteralE	**255**	0.303	0.219	0.33	0.471

Results

YAGO3-10					
Models	**MR**	**MRR**	**Hits@1**	**Hits@3**	**Hits@10**
DistMult	2943	0.466	0.377	0.514	0.653
ComplEx	3768	0.493	0.411	0.536	0.649
ConvE	2141	0.505	0.422	0.554	0.660
KBLN	2666	0.487	0.405	0.531	0.642
MTKGNN [TTPH17]	2970	0.481	0.398	0.527	0.634
DistMult-LiteralE	1642	0.479	0.4	0.525	0.627
ComplEx-LiteralE	2508	0.485	0.412	0.527	0.618
ConvE-LiteralE	**1037**	**0.525**	**0.448**	**0.572**	**0.659**

UNIVERSITÄT BONN
Rheinische Friedrich-Wilhelms-Universität Bonn

Results

Table: Link prediction results for DistMult-LiteralE on FB15k-237, with both numerical and text literals. "N" and "T" denotes the usage of numerical and text literals, respectively.

Models	MRR	Hits@1	Hits@10	MRR Improv.
DistMult	0.241	0.155	0.419	-
DistMult-LiteralE (N)	0.317	0.232	0.483	+31.54%
DistMult-LiteralE (N+T)	**0.32**	**0.234**	**0.488**	**+32.78%**

UNIVERSITÄT BONN
Rheinische Friedrich-Wilhelms-Universität Bonn

Results

Nearest Neighbor Analysis

Entity	Methods	Nearest Neighbors
Roman Republic	DistMult	Republic of Venice, Israel Defense Force, Byzantine Empire
	KBLN	Republic of Venice, Carthage, Retinol
	MTKGNN	Republic of Venice, Carthage, North Island
	Num. lits. only	Alexandria, Yerevan, Cologne
	LiteralE	**Roman Empire**, **Kingdom of Greece**, **Byzantine Empire**

Conclusion & Future Work

1. We introduced LiteralE: a simple method to incorporate literals into latent feature methods for knowledge graph analysis.
2. We showed that augmenting various state-of-the-art models (DistMult, ComplEx, and ConvE) with LiteralE significantly improves their link prediction performance.
3. LiteralE is a promising candidate for improving other tasks in the field of knowledge graph analysis, such as entity resolution and knowledge graph clustering.

References I

- Tim Dettmers, Minervini Pasquale, Stenetorp Pontus, and Sebastian Riedel, *Convolutional 2d knowledge graph embeddings*, Proceedings of the 32th AAAI Conference on Artificial Intelligence, February 2018.
- Alberto Garcia-Duran and Mathias Niepert, *KBLRN: End-to-end learning of knowledge base representations with latent, relational, and numerical features*, arXiv preprint arXiv:1709.04676 (2017).
- Diederik P Kingma and Jimmy Ba, *Adam: A method for stochastic optimization*, 3rd International Conference for Learning Representations, ICLR, 2015.
- Pouya Pezeshkpour, CA Irvine, Liyan Chen, and Sameer Singh, *Embedding multimodal relational data*.

References II

- Yi Tay, Luu Anh Tuan, Minh C Phan, and Siu Cheung Hui, *Multi-task neural network for non-discrete attribute prediction in knowledge graphs*, Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, ACM, 2017, pp. 1029–1038.
- Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard, *Complex embeddings for simple link prediction*, International Conference on Machine Learning, 2016, pp. 2071–2080.
- Bishan Yang, Scott Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng, *Embedding entities and relations for learning and inference in knowledge bases*, 3rd International Conference for Learning Representations, ICLR, 2015.

UNIVERSITÄT BONN Rheinische Friedrich-Wilhelms-Universität Bonn

Thank you!

UNIVERSITÄT BONN
Rheinische Friedrich-Wilhelms-Universität Bonn