LiteralE: Incorporating Literals into Knowledge Graph Embeddings

A. Kristiadi¹, A. Khan¹, D. Lukovnikov, J. Lehmann, A. Fischer

October 28, 2019

¹Equal Contribution
Outline

Motivation

Background
- Latent Feature Methods

LiteralE
- LiteralE Formulation

Experimental Setup & Results
- Results

Conclusion & Future Work

References
**Motivation**

Literals for link prediction

**Background**

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

**LiteralE**

**Experimental Setup & Results**

**Conclusion & Future Work**

**References**
Background
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.
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 $\mathcal{E} = \{e_1, \ldots, e_{N_e}\}$ is the set of entities,
  $\mathcal{R} = \{r_1, \ldots, r_{N_r}\}$ is the set of relations connecting two entities,
  $\mathcal{D} = \{d_1, \ldots, d_{N_d}\}$ 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.
Link Prediction

• A knowledge graph $\mathcal{G}$ is a subset of $\{E \times E \times R\} \cup \{E \times L \times D\}$ representing the facts that are assumed to hold.
  where $E = \{e_1, \cdots, e_N\}$ is the set of entities,
  $R = \{r_1, \cdots, r_N\}$ is the set of relations connecting two entities,
  $D = \{d_1, \cdots, d_N\}$ 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 $(e_i, e_j, r_k) \in E \times E \times R$ 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 $\psi: E \times E \times R \rightarrow \mathbb{R}$, where a higher score implies the triple is more likely to be true.
Latent Feature Methods

- \( \psi(e_i, e_j, r_k) \overset{\text{def}}{=} f(e_i, e_j, r_k) \).
Latent Feature Methods

- $\psi(e_i, e_j, r_k) \overset{\text{def}}{=} f(e_i, e_j, r_k)$.
- $f : \mathbb{R}^H \times \mathbb{R}^H \times \mathbb{R}^H \rightarrow \mathbb{R}$ computes a score of triple $(e_i, e_j, r_k)$ that correlates with the truth value of the triple.
Latent Feature Methods

\begin{itemize}
\item $\psi(e_i, e_j, r_k) \overset{\text{def}}{=} f(e_i, e_j, r_k)$.
\item $f : \mathbb{R}^H \times \mathbb{R}^H \times \mathbb{R}^H \rightarrow \mathbb{R}$ computes a score of triple $(e_i, e_j, r_k)$ that correlates with the truth value of the triple.
\item Latent Feature Methods are a class of methods defined to learn low dimensional vector representations of entities and relations, called embeddings or latent features.
\end{itemize}
Latent Feature Methods - Models

- DistMult $[YYH^{+}15]$:

$$f_{\text{DistMult}}(e_i, e_j, r_k) = \langle e_i, e_j, r_k \rangle = e_i^T \text{diag}(r_k) e_j,$$
Latent Feature Methods - Models

- **DistMult [YYH⁺15]:**
  \[
  f_{\text{DistMult}}(e_i, e_j, r_k) = \langle e_i, e_j, r_k \rangle = e_i^\top \text{diag}(r_k) e_j ,
  \]

- **ComplEx [TWR⁺16]:**
  \[
  f_{\text{ComplEx}}(e_i, e_j, r_k) = \text{Re}(\langle e_i, \bar{e}_j, r_k \rangle)
  = \langle \text{Re}(e_i), \text{Re}(e_j), \text{Re}(r_k) \rangle
  + \langle \text{Im}(e_i), \text{Im}(e_j), \text{Re}(r_k) \rangle
  + \langle \text{Re}(e_i), \text{Im}(e_j), \text{Im}(r_k) \rangle
  - \langle \text{Im}(e_i), \text{Re}(e_j), \text{Im}(r_k) \rangle,
  \]
Latent Feature Methods - Models

• DistMult [YYH+15]:

\[ f_{\text{DistMult}}(e_i, e_j, r_k) = \langle e_i, e_j, r_k \rangle = e_i^T \text{diag}(r_k) e_j , \]

• ComplEx [TWR+16]:

\[ f_{\text{ComplEx}}(e_i, e_j, r_k) = \text{Re}(\langle e_i, \bar{e}_j, r_k \rangle) \]
\[ = \langle \text{Re}(e_i), \text{Re}(e_j), \text{Re}(r_k) \rangle \]
\[ + \langle \text{Im}(e_i), \text{Im}(e_j), \text{Re}(r_k) \rangle \]
\[ + \langle \text{Re}(e_i), \text{Im}(e_j), \text{Im}(r_k) \rangle \]
\[ - \langle \text{Im}(e_i), \text{Re}(e_j), \text{Im}(r_k) \rangle , \]

• ConvE [DPPR18]:

\[ f_{\text{ConvE}}(e_i, e_j, r_k) = h(\text{vec}(h([e_i, r_k] * \omega)))W) e_j . \]
LiteralE
Our Contribution

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

\(^2\)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

²A 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.

²A literal-extended version of YAGO3-10 is provided by Pezeshkpou et al. [PICS17]
LiteralE - entity and literal vectors

• $e_i \in \mathbb{R}^H$: “vanilla” embedding for entity $i$ — trainable parameters

• $l_i \in \mathbb{R}^{N_d}$: 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
    • $l_i$ for a person: $[1.80, 1990, 0, 0]^T$
    • $l_i$ for NZ: $[0, 0, 268000, 0]^T$
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 $f_X(e_i, e_j, r_k)$ with

$$f_X(g(e_i, l_i), g(e_j, l_j), r_k)$$
LiteralE

- (Linear version of $g$):

$$g_{\text{lin}}(e_i, l_i) = W^T[e_i, l_i],$$
• (Linear version of $g$):

$$g_{\text{lin}}(e_i, l_i) = W^T[e_i, l_i],$$

• We define $g$ for numerical literals as:

$$g : \mathbb{R}^H \times \mathbb{R}^{N_d} \rightarrow \mathbb{R}^H$$

$$e, l \mapsto z \odot h + (1 - z) \odot e,$$

(1)

$$z = \sigma(W_{ze}^T e + W_{zl}^T l + b)$$

$$h = h(W_h^T [e, l]).$$

(2)
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:

\[
g : \mathbb{R}^H \times \mathbb{R}^{N_d} \times \mathbb{R}^{N_t} \rightarrow \mathbb{R}^H
\]

\[
e, l, t \mapsto z \odot h + (1 - z) \odot e, \quad (3)
\]

\[
z = \sigma(W^T_{ze}e + W^T_{zl}l + W^T_{zt}t + b)
\]

\[
h = h(W^T_h[e, l, t]). \quad (4)
\]

\(^3\)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.
We use three widely used datasets for evaluating link prediction performance: FB15k, FB15k-237, and YAGO3-10.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>FB15k</th>
<th>FB15k-237</th>
<th>YAGO3-10</th>
</tr>
</thead>
<tbody>
<tr>
<td># Entities ($N_e$)</td>
<td>14,951</td>
<td>14,541</td>
<td>123,182</td>
</tr>
<tr>
<td># Relations ($N_r$)</td>
<td>1,345</td>
<td>237</td>
<td>37</td>
</tr>
<tr>
<td># Data rel. ($N_d$)</td>
<td>121</td>
<td>121</td>
<td>5</td>
</tr>
<tr>
<td># Literals ($</td>
<td>\mathcal{L}</td>
<td>$)</td>
<td>18,741</td>
</tr>
<tr>
<td># Relational triples</td>
<td>592,213</td>
<td>310,116</td>
<td>1,089,040</td>
</tr>
<tr>
<td># Literal triples</td>
<td>70,257</td>
<td>70,257</td>
<td>111,406</td>
</tr>
</tbody>
</table>
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 \( p \in [0, 1]^{N_e} \) and the ground truth labels \( y \in \{0, 1\}^{N_e} \):
  - \( y_j = 1 \iff \text{triple } (e_i, e_j, r_k) \text{ exists in the KG} \)
  - \( N_e \) is the number of entities
  - \( p_j = \sigma(f_X(\cdot)) \)
  \[
  L(p, y) = -\frac{1}{N_e} \sum_{j=1}^{N_e} (y_j \log(p_j) + (1 - y_j) \log(1 - p_j))
  \]
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.
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

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

### FB15k-237

<table>
<thead>
<tr>
<th>Models</th>
<th>MR</th>
<th>MRR</th>
<th>Hits@1</th>
<th>Hits@3</th>
<th>Hits@10</th>
</tr>
</thead>
<tbody>
<tr>
<td>DistMult</td>
<td>633</td>
<td>0.282</td>
<td>0.203</td>
<td>0.309</td>
<td>0.438</td>
</tr>
<tr>
<td>ComplEx</td>
<td>652</td>
<td>0.290</td>
<td>0.212</td>
<td>0.317</td>
<td>0.445</td>
</tr>
<tr>
<td>ConvE</td>
<td>297</td>
<td>0.313</td>
<td>0.228</td>
<td>0.344</td>
<td>0.479</td>
</tr>
<tr>
<td>KBLN [GDN17]</td>
<td>358</td>
<td>0.301</td>
<td>0.215</td>
<td>0.333</td>
<td>0.468</td>
</tr>
<tr>
<td>MTKGNN [TTPH17]</td>
<td>532</td>
<td>0.285</td>
<td>0.204</td>
<td>0.312</td>
<td>0.445</td>
</tr>
<tr>
<td>DistMult-LiteralE</td>
<td>280</td>
<td><strong>0.317</strong></td>
<td><strong>0.232</strong></td>
<td><strong>0.348</strong></td>
<td><strong>0.483</strong></td>
</tr>
<tr>
<td>ComplEx-LiteralE</td>
<td>357</td>
<td>0.305</td>
<td>0.222</td>
<td>0.336</td>
<td>0.466</td>
</tr>
<tr>
<td>ConvE-LiteralE</td>
<td>255</td>
<td>0.303</td>
<td>0.219</td>
<td>0.33</td>
<td>0.471</td>
</tr>
</tbody>
</table>
## Results

### YAGO3-10

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

<table>
<thead>
<tr>
<th>Models</th>
<th>MRR</th>
<th>Hits@1</th>
<th>Hits@10</th>
<th>MRR Improv.</th>
</tr>
</thead>
<tbody>
<tr>
<td>DistMult</td>
<td>0.241</td>
<td>0.155</td>
<td>0.419</td>
<td>-</td>
</tr>
<tr>
<td>DistMult-LiteralE (N)</td>
<td>0.317</td>
<td>0.232</td>
<td>0.483</td>
<td>+31.54%</td>
</tr>
<tr>
<td>DistMult-LiteralE (N+T)</td>
<td>0.32</td>
<td>0.234</td>
<td>0.488</td>
<td>+32.78%</td>
</tr>
</tbody>
</table>
## Results

### Nearest Neighbor Analysis

<table>
<thead>
<tr>
<th>Entity</th>
<th>Methods</th>
<th>Nearest Neighbors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Roman Republic</td>
<td>DistMult</td>
<td>Republic of Venice, Israel Defense Force, Byzantine Empire</td>
</tr>
<tr>
<td>KBLN</td>
<td></td>
<td>Republic of Venice, Carthage, Retinol</td>
</tr>
<tr>
<td>MTKGNN</td>
<td></td>
<td>Republic of Venice, Carthage, North Island</td>
</tr>
<tr>
<td>Num. lits. only</td>
<td></td>
<td>Alexandria, Yerevan, Cologne</td>
</tr>
<tr>
<td>LiteralE</td>
<td></td>
<td>Roman Empire, Kingdom of Greece, Byzantine Empire</td>
</tr>
</tbody>
</table>
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


References II


Thank you!