Collaborative Knowledge Graph (CKG)

Figure 1: Incorporating knowledge graph into user-item bipartite graph.

Components of CKG

- User-Item Bipartite Graph
  - User-Item Direct Interactions
    \[ u_i \rightarrow l_i \rightarrow e_i \rightarrow r_i \rightarrow u_j \]
  - Item-Item External Connections
    \[ l_i \rightarrow e_i \rightarrow l_j \rightarrow u_i \]

Collaborative Knowledge Graph

- High-order connectivity between users and items
  \[ u_i \rightarrow l_i \rightarrow e_i \rightarrow l_j \rightarrow u_j \]
- Possible reasons on potential recommendations

How to Achieve High-order Relation Modeling in an Explicit & End-to-End Manner?

Inspired by the recent success of graph neural networks, we propose Knowledge Graph Attention Network (KGAT) for KG-based Recommendation.

Knowledge Graph Attention Network (KGAT)

![Diagram of KGAT model]

Figure 4: Illustration of the proposed KGAT model.

I. CKG Embedding Layer

We adopt Transform function to parameterize entities and relations of CKG as vector representations, considering direct connectivity of each triplet \((h, r, t)\).

\[
g(h, r, t) = [W_h e_h + e_r - W_t e_t]^2, \quad L_{KG} = \sum_{(h, r, t) \in T} - \ln \sigma(g(h, r, t) - g(h, r, t)).
\]

II. Attention Embedding Propagation Layer

- Information Propagation: We perform information propagation between an entity \(h\) and its neighbors \(\mathcal{N}_h\):
  \[
e_N(h) = \sum_{(h, r, t) \in T} \tau(h, r, t) e_t,
\]

where \(\tau(h, r, t)\) controls how much information is propagated from tail entity \(t\) to head entity \(h\) conditioned to relation \(r\).

- Knowledge-aware Attention: We implement \(\tau(h, r, t)\) via relational attention mechanism, which is formulated as follows:
  \[
  \tau(h, r, t) = \|W_{h} e_h + e_r\|^2 \|W_{t} e_t + e_t\|^2.
  \]

- Information Aggregation: The final phase is to aggregate the entity representation \(e_h\) and its ego-network representations \(e_N(h)\) as the new representation of entity \(h\):
  \[
e_h^{(N)} = \text{LeakyReLU}(W_N e_N) + \text{LeakyReLU}(W_2 e_h + e_N).
\]

- High-order Propagation: We further stack more propagation layers to explore the high-order connectivity information, gathering the information propagated from the higher-hop neighbors:
  \[
e_h^{(N+1)} = \sum_{(h, r, t) \in T} \tau(h, r, t) e_t^{(N+1)}.
\]

III. Model Prediction

After performing \(L\) layers, we obtain multiple representations for user node \(u\), namely \(e_u^{(0)} \ldots e_u^{(L)}\); analogous to item node \(i\):

\[
e_u^{(0)} = e_u^{(0)} \ldots e_u^{(L)} = e_i^{(0)} \ldots e_i^{(L)}.
\]

Finally, we conduct inner product of user and item representations, so as to predict their matching score:

\[
\hat{y}(u, i) = e_u^T e_i,
\]

\[
L_{CF} = \sum_{(u,i) \in D} - \ln \sigma(\hat{y}(u, i) - \hat{y}(u, j)).
\]

IV. Model Optimization

Results

<table>
<thead>
<tr>
<th>Overall Performance Comparison</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amazon-Book</td>
</tr>
<tr>
<td>recall</td>
</tr>
<tr>
<td>FM</td>
</tr>
<tr>
<td>NFM</td>
</tr>
<tr>
<td>CKE</td>
</tr>
<tr>
<td>CFRG</td>
</tr>
<tr>
<td>MCRec</td>
</tr>
<tr>
<td>RippleNet</td>
</tr>
<tr>
<td>KGAT</td>
</tr>
<tr>
<td>Improve:</td>
</tr>
</tbody>
</table>

Figure 5: KGAT consistently yields the best performance on all the datasets.

Interaction Sparsity Levels

![Graph showing interaction sparsity levels]

Case Study for Explainable Recommendation

![Diagram showing case study]

Our Goal: Develop a model that can exploit high-order information in KG in an efficient, explicit, and end-to-end manner.

Summary & Limitations of Three-type Works

<table>
<thead>
<tr>
<th>Supervised Learning-based</th>
<th>Path-based</th>
<th>Regularization-based</th>
</tr>
</thead>
<tbody>
<tr>
<td>Knowledge Usage</td>
<td>a generic feature vector</td>
<td>Connectivity to paths connecting users &amp; items</td>
</tr>
<tr>
<td>Relation Usage</td>
<td>-</td>
<td>To define meta-path or select qualified paths</td>
</tr>
<tr>
<td>Limitations</td>
<td>* Fail to capture CF signals</td>
<td>* Require labor-intensive feature engineering</td>
</tr>
<tr>
<td>Examples</td>
<td>FM, NFM, TEM, WideDeep ...</td>
<td>MRec, RippleNet, FM, KPRN ...</td>
</tr>
</tbody>
</table>

Figure 3: Due to the characteristics of these models, high-order relations have not been fully and properly explored.

Finaly, we have the objective function to learn Equations (2) and (9) jointly, as follows:

\[
L_{KG} = L_{KG} + L_{CF}.
\]