Entity Alignment

Group 1
Ann Tan, Steffen Brandenburg, Mustafa Yasin
Motivation

“AI systems make decision by using rules and logic to reason about **facts** or deduce new facts.”

**Knowledge Bases**
- Collection of facts/ground truths about the world.
- Structured information stored by a computer system.
- Used by expert systems for inference and deduction.

**Internet** as a knowledge base
- multiple sources in different languages
- i.e. Wikipedia, DBPedia, Wikidata, IMDB
- HOW TO to unify information? ⇨ **Entity Alignment**
Knowledge Graph

- knowledge base represented as a graph
  - Entity - represented by itself
  - Relationship - inferred; with other entities, facts, circumstances
- a form of semantic network, limited to a specific domain.
- organized, easy to understand, to extract and to infer information from.

**Uses of KGs:**
1. Dialog Systems
2. Natural Language Generation
3. Question-Answering System
4. NER in Computational Argumentation
Entity Alignment

**TASK:** Find entities in 2 KGs that represent the same real-world entities.

1. **Input** includes:
   a. Two (2) KGs - left and right, each with:
      - List of **entities**, **relations**; some include attributes
      - **Triples** in the form (head entity, relation, tail entity)
   b. **Pre-aligned entities** from left KG to right KG

2. Do **supervised training** using:
   a. Representation of entities / relations / attributes (i.e. **embeddings**).
   b. Triples to represent the **graph structure** (Adjacency, Degree).
   c. **Alignment** as supervision.

3. Predict correspondence of entities from both KGs using **similarity** metrics on the entities’ representations.

Source: https://dl.acm.org/doi/pdf/10.1145/3336191.3371804
Group Tasks

1. Analyze datasets.
2. Compare various approaches to KG Alignment.
   a. Read papers.
   b. Run published codes.
   c. Retrieve baseline results
3. Adapt from DBS Framework or implement new modules for the approaches in #2.
4. Test, Train and Evaluate.
## Datasets

<table>
<thead>
<tr>
<th>Sources</th>
<th>Name</th>
<th>Subset</th>
<th>Triple Size</th>
<th>Top Entities</th>
<th>Top Relations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wikipedia (n-n)</td>
<td>WK3I15k</td>
<td>en-de-fr</td>
<td>WK3I120K (en) largest 1.3M</td>
<td>Countries Music Genre</td>
<td>(25% - 47% of triples) Name SPORTS Position</td>
</tr>
<tr>
<td></td>
<td>WK3I120k</td>
<td></td>
<td>triples</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Countries Music Genre</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DBPedia (1-1)</td>
<td>DBP15k (Full)</td>
<td>zh-en-fr</td>
<td>DBP15k JAPE (zh) smallest 70K</td>
<td>Countries Producer Record Label</td>
<td>(14% - 25% of triples) Starring Birthplace Writer / Producer</td>
</tr>
<tr>
<td></td>
<td>DBP15k (JAPE)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wikipedia + Wikidata + DBPedia (1-1)</td>
<td>DWY100k</td>
<td>en-wd-en-yg</td>
<td>Between 400-500K triples</td>
<td>Countries Year Sports Position</td>
<td>(36% - 76% of triples) Birthplace Year / Place of Death Name Team / Goals</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Datasets: Degree

**DBP15k (JAPE) zh-en**

- Avg Degree: 6.57 | 8.57
- A very small proportion of nodes have > 600

**WK315k en-de**

- Avg Degree: 21.62 | 16.51
- Some nodes have considerably larger degrees (max 7,000)
General Approaches to KG Alignment

- **Architecture**
  a. GCN
  b. GAT (MRAEA)
  c. TransE/MTransE (OTEA & KAGAN)

- **Embeddings**
  a. Node
  b. Edges
  c. TransE
  d. Attributes
Approach #1: Graph Convolutional Network

- **Message Passing:**
  - Messages are the node embeddings.
  - For each time step, at each reference node, messages from its neighbors are aggregated ($\Sigma$).
  - The aggregated messages becomes the updated embedding of the reference node.

Approach #2: Graph Attention Network (GAT)

- Assign **varying levels of importance** to the node’s neighborhood.
- A single GAT layer can be described as $e_{ij} = a(W\bar{x}_i, W\bar{x}_j)$, where
  - $e_{ij}$ - **attention coefficient** or importance of edge $(e_i, e_j)$’s features for a source node $e_i$
  - graph structure is retained by allowing node $i$ to “attend” only to its neighborhood
  - $W$ - embedding weights
  - $a$ - any attention function
- The relative attention score is computed using **softmax** over all the values in the neighborhood.
  - $\bar{x}_i$ is the transformed node feature of $e_i$
- GAT employs **multi-head attention** to stabilize the learning process.

\[
x'_i = \frac{1}{K} \sum_{k=1}^{K} \sigma \left( \sum_{j \in N_i} \alpha_{ij}^k W^k \bar{x}_j \right)
\]
GAT: MRAEA (Mao et al., 2020)

- GAT Model
- Attention Score: node embedding + embeddings of neighbors + relation embeddings (type, direction, inverse - dashed lines)

Source: https://petar-v.com/GAT/
Approach #3: TransE [Bordes et al., 2013]

- Embeds \((h, r, t)\) of a KG into a different space
- Goal is that \(h + r \approx t\) holds (boldfaced \(h, r, t\) are the embedded \(h, r, t\) respectively)
Approach #3: MTransE  [Chen et al., 2016]

- 2 KGs are embedded using TransE
- A linear transformation between the spaces is learned via L2-Loss
OTEA (Optimal Transport Entity Alignment)

- **Extends** MTransE [Chen et al., 2016]
- Additionally considers **Group-Level Loss** (distance between E and M’E’)
- Group-Level Loss computed using Wasserstein GAN [Arjovsky et al., 2017]
Based on MTransE

Generator creates fake examples

Discriminator minimizes difference btw. the aligned triplets & triplets in target Graph
## DBS Framework Adaptation

<table>
<thead>
<tr>
<th>DBS Modules</th>
<th>OTEA</th>
<th>KAGAN</th>
<th>MRAEA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dataset Loader</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Embeddings</td>
<td>+ (MTransE)</td>
<td>+ (MTransE)</td>
<td>✎ (Relation)</td>
</tr>
<tr>
<td>Graph (Message Passing)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Layers</td>
<td>+ (GAN)</td>
<td>+ (GAN)</td>
<td>+ (GAT)</td>
</tr>
<tr>
<td>Similarity</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Trainer</td>
<td>✓</td>
<td>✓</td>
<td>✎ (Bootstrap)</td>
</tr>
<tr>
<td>Evaluator</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

✓: Adapted as is  
✎: Revised  
+: New  
N/A: Not Applicable
## Results - OTEA

<table>
<thead>
<tr>
<th>Wk-31-15k Dataset</th>
<th>en-fr</th>
<th>en-de</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Hits@1</td>
<td>Hits@5</td>
</tr>
<tr>
<td>1. Results in Paper</td>
<td>0.375</td>
<td>0.574</td>
</tr>
<tr>
<td>2. Published Code Results</td>
<td>0.371</td>
<td>0.465</td>
</tr>
<tr>
<td>3. DBS Framework</td>
<td>0.080</td>
<td>0.156</td>
</tr>
</tbody>
</table>
# Results - MRAEA

<table>
<thead>
<tr>
<th>DBP15k_JAPE (zh_en)</th>
<th>Hits @ 1 (%)</th>
<th>Hits @ 5 (%)</th>
<th>MRR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Results in Paper</td>
<td>75.28</td>
<td>92.31</td>
<td>0.824</td>
</tr>
<tr>
<td>2. Published code Results</td>
<td>62.56</td>
<td>82.34</td>
<td>0.715</td>
</tr>
<tr>
<td>3. DBS Framework</td>
<td>11.18</td>
<td>40.27</td>
<td>0.206</td>
</tr>
<tr>
<td>4. - 49k Epochs</td>
<td>19.55</td>
<td>52.67</td>
<td>0.303</td>
</tr>
</tbody>
</table>

**Difference (#2 - #4)**: 43.01 | 29.77 | 0.412

“All parameters being equal...”

- **emb_dim**: 100
- **margin loss**: 3
- **layers**: 2
- **heads**: 2
- **emb_dropout**: 0.3
- **Adam lr**: 0.005
- **bias**: yes
- **train-test split**: 0.30
- **eval split**: None
- **batch_size**: num_entities
- **epochs**: 5000
Insights & Challenges

- Steep learning curve when using the DBS Framework
- Difficult to implement approach just by looking at the paper description (formula, etc.)
  - Some had no published code.
  - Some published code required debugging.
  - Published code included details that were not described in the paper.
- Difficult to work in the group because we were assigned different papers
Future Work

- Find out why results differ from the published results
- DBS Framework documentation
  - class organization, inventory of methods
- Continue on different approaches
Thank You!

Questions?