Augmenting Low-Resource Text Classification with Graph-Grounded Pre-training and Prompting

Zhihao Wen and Yuan Fang
School of Computing and Information Systems
Singapore Management University
Outline

- Introduction
- Methodology
- Experiment
- Conclusion & Future work
Low-resource multi-task text classification

Low-resource Training texts

Labels

The BERT Model......

NLP

Novel Recommender Systems using......

RecSys

......

Deep Learning for Image Captioning ......

......

Computer vision

Many tasks and each task is a different text classification task

Training $f$

Prediction

Task (1) DM v.s. Recys

Task (2) NLP v.s. Software

Task (N) CV v.s. DB

e.g., for each class, we have only one labeled training samples

Recommender models are ......

Stackoverflow summarition ......

Swin transformer for ......

Language models are ......

Computer vision

Language models are ......

Computer vision

Language models are ......

Computer vision

Language models are ......

Computer vision
Text data are grounded on network structures

- Text data are frequently grounded on **network structures**
- Graph structures expose valuable **relationships**
- **GNNs** are designed to learn from graph structures
Challenges and present work

Q1: How do we capture fine-grained textual semantics, while leveraging graph structure information jointly?

We propose a graph-grounded contrastive pre-training, to maximize the alignment between text and graph representations based on three types of graph interaction.

Q2: How do we augment low-resource multi-task text classification given a jointly pre-trained graph-text model?

We propose a novel approach of "prompting" a jointly pre-trained graph-text model instead of fine-tuning it.
Outline

- Introduction
- Methodology
- Experiment
- Conclusion & Future work
Preliminary: Graph-grounded text corpus

- Consider a set of documents $\mathcal{D}$, which is grounded on a graph $\mathcal{G}$ such that each document $d_i$ is a node $v_i$ in the graph.
- Documents are linked via edges.
- Each node $v_i$ is also associated with a feature vector $X_i$.
- Each document/node has a class label.

Language models are ...

The translation ...

Visual QA ...

The BERT model ...

Label: NLP
Overall framework of our proposed G2P2

(a) Graph-grounded contrastive pre-training
(b) Graph-grounded prompt tuning (few-shot classification)

Overall framework of G2P2. (a) During pre-training, it jointly trains a text and a graph encoder through three contrastive strategies. (b) During testing, it performs prompt-assisted zero- or few-shot classification.
Preliminary: prompt learning

• Prompt learning in NLP: the process of formulating effective prompts or instructions to guide pre-trained language models to generate desired outputs.

Figure from [1]

Our proposed graph-grounded contrastive pre-training

- Learn a dual-modal embedding space jointly training a text encoder and graph encoder through 3 contrastive strategies.
Graph-grounded contrastive pre-training

Dual-encoders

1. Text-encoder: a transformer
   \[ t_i = \Phi_T(d_i; \theta_T) \]

2. Graph-encoder: a GCN
   \[ z_i = \Phi_Z(v_i; \theta_G) \]
Text-node interaction

- Graph-grounded texts naturally implies a **bijection** between nodes and texts
- Predict the **text** of a document **matches** which **node** in the graph.
- Given **n documents** and the corresponding **n nodes**, there are **n^2** possible document node pairs
- Only **n** pairs with \( i = j \) are true matching
- The remaining **n^2–n** pairs are **false matching**
- Maximize the cosine similarity of **n matching** pairs, while **minimizing** that of the **n^2 – n unmatching** pairs
Text-summary interaction

- Each document has a set of neighboring documents defined by graph topology.
- The neighboring documents are a summary of the target document.
- Employ a simple mean pooling to generate the summary embedding:
  \[
  s_i = \frac{1}{|N_i|} \sum_{j \in N_i} t_j
  \]
- Align the text embedding and its corresponding summary text embedding.
Node-summary interaction

- Neighborhood based summary $s_i$ for document $d_i$ also serves as a semantic description of node $v_i$.
- Align the node embedding $z_i$ and its neighborhood-based summary text embedding $s_i$. 
Overall pre-training objective

- Integrate the three contrastive losses based on the text-node, text-summary and node-summary interactions

- Obtain a pre-trained model $\theta^0$ consisting of the parameters of the dual encoders

$$\theta^0 = \arg \min_{\theta_T, \theta_G} \mathcal{L}_1 + \lambda(\mathcal{L}_2 + \mathcal{L}_3)$$

Hyperparameter
Prompt-assisted text classification

- Discrete prompt for zero-shot classification

- Predict the class whose label text embedding has the highest similarity to the node embedding

- **Classification weights** can be generated by the text encoder based on the class label texts

  \[ \mathbf{w}_y = \phi_T(\text{“prompt [CLASS]”}; \theta^0_T) \]

  e.g., “A paper of” label text, e.g., “NLP”

- Class distribution is predicted as

  \[
p(y | \mathbf{z}_i) = \frac{\exp (\langle \mathbf{z}_i, \mathbf{w}_y \rangle)}{\sum_{y=1}^{N} \exp (\langle \mathbf{z}_i, \mathbf{w}_y \rangle)}
  \]

  cosine similarity
Graph-grounded prompt tuning

- Discrete prompts are difficult to optimize.
- Resort to **prompt tuning**, substituting discrete prompts with learnable continuous vectors, while keeping the parameters of PLM **frozen**.
- Instead of a sequence of **discrete tokens**, we use a sequence of **continuous embeddings**

\[ w_y = \phi_T([h_1, \cdots, h_M, h_{\text{CLASS}}]; \theta^0_T) \]

- We initialize the prompt embeddings with **graph contexts**.
- A node \( v_i \) and its neighbor set \( \{v_j | j \in \mathcal{N}_i \} \) are collectively called the **graph contexts** of \( v_i \).
Outline

- Introduction
- Methodology
- Experiment
- Conclusion & Future work
Datasets

Table 1: Statistics of datasets.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Cora</th>
<th>Art</th>
<th>Industrial</th>
<th>M.I.</th>
</tr>
</thead>
<tbody>
<tr>
<td># Documents</td>
<td>25,120</td>
<td>1,615,902</td>
<td>1,260,053</td>
<td>905,453</td>
</tr>
<tr>
<td># Links</td>
<td>182,280</td>
<td>4,898,218</td>
<td>3,101,670</td>
<td>2,692,734</td>
</tr>
<tr>
<td># Avg. doc length</td>
<td>141.26</td>
<td>54.23</td>
<td>52.15</td>
<td>84.66</td>
</tr>
<tr>
<td># Avg. node deg</td>
<td>7.26</td>
<td>3.03</td>
<td>2.46</td>
<td>2.97</td>
</tr>
<tr>
<td># Classes</td>
<td>70</td>
<td>3,347</td>
<td>2,462</td>
<td>1,191</td>
</tr>
</tbody>
</table>

Cora is a collection of research papers

Art, Industrial and Music Instruments (M.I.) are 3 Amazon review datasets
## Performance comparison with baselines

<table>
<thead>
<tr>
<th></th>
<th>Cora</th>
<th>Art</th>
<th>Industrial</th>
<th>M.I.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Accuracy</td>
<td>Macro-F1</td>
<td>Accuracy</td>
<td>Macro-F1</td>
</tr>
<tr>
<td>GCN</td>
<td>41.15±2.41</td>
<td>34.50±2.23</td>
<td>22.47±1.78</td>
<td>15.45±1.14</td>
</tr>
<tr>
<td>SAGE_{sup}</td>
<td>41.42±2.90</td>
<td>35.14±2.14</td>
<td>22.60±0.56</td>
<td>16.01±0.28</td>
</tr>
<tr>
<td>TextGCN</td>
<td>59.78±1.88</td>
<td>55.85±1.50</td>
<td>43.47±1.02</td>
<td>32.20±1.30</td>
</tr>
<tr>
<td>GPT-GNN</td>
<td>76.72±2.02</td>
<td>72.23±1.17</td>
<td>65.15±1.37</td>
<td>52.79±0.83</td>
</tr>
<tr>
<td>DGI</td>
<td>78.42±1.39</td>
<td>74.58±1.24</td>
<td>65.41±0.86</td>
<td>53.57±0.75</td>
</tr>
<tr>
<td>SAGE_{self}</td>
<td>77.59±1.71</td>
<td>73.47±1.53</td>
<td>76.13±0.94</td>
<td>65.25±0.31</td>
</tr>
<tr>
<td>BERT</td>
<td>37.86±5.31</td>
<td>32.78±5.01</td>
<td>46.39±1.05</td>
<td>37.07±0.68</td>
</tr>
<tr>
<td>BERT*</td>
<td>27.22±1.22</td>
<td>23.34±1.11</td>
<td>45.31±0.96</td>
<td>36.28±0.71</td>
</tr>
<tr>
<td>RoBERTa</td>
<td>62.10±2.77</td>
<td>57.21±2.51</td>
<td>72.95±1.75</td>
<td>62.25±1.33</td>
</tr>
<tr>
<td>RoBERTa*</td>
<td>67.42±4.35</td>
<td>62.72±3.02</td>
<td>74.47±1.00</td>
<td>63.35±1.09</td>
</tr>
<tr>
<td>P-Tuning v2</td>
<td>71.00±2.03</td>
<td>66.76±1.95</td>
<td>76.86±0.59</td>
<td>66.89±1.14</td>
</tr>
<tr>
<td>G2P2-p</td>
<td>79.16±1.23</td>
<td>74.99±1.35</td>
<td>79.59±0.31</td>
<td>68.26±0.43</td>
</tr>
<tr>
<td>G2P2</td>
<td><strong>80.08±1.33</strong></td>
<td><strong>75.91±1.39</strong></td>
<td><strong>81.03±0.43</strong></td>
<td><strong>69.86±0.67</strong></td>
</tr>
<tr>
<td>(improv.)</td>
<td>(+2.12%)</td>
<td>(+1.78%)</td>
<td>(+5.43%)</td>
<td>(+4.44%)</td>
</tr>
</tbody>
</table>

- G2P2 outperforms the best baseline by around 3–7%, showing the advantage of our contrastive pre-training and graph grounded prompt tuning.
Outline

- Introduction
- Methodology
- Experiment
- Conclusion & Future work
Conclusion

Key contributions
• Addressed the problem of low-resource multi-task text classification;
• Proposed G2P2, consisting of three graph interaction-based contrastive strategies in pre-training, and a prompting mechanism for the jointly pre-trained graph-text model in downstream classification.

Limitations
• The need of a graph to complement the texts
• Cannot do prompt tuning for zero-shot
THANK YOU FOR YOUR ATTENTION

Paper, code, data...

www.yfang.site