Incorporating Syntactic and Semantic Information in Word Embeddings using Graph Convolutional Network

Shikhar Vashisht\textsuperscript{1}, Manik Bhandari\textsuperscript{2*}, Prateek Yadav\textsuperscript{3}, Piyush Rai\textsuperscript{4}, Chiranjib Bhattacharyya\textsuperscript{5}, Partha Talukdar\textsuperscript{1}
Indian Institute of Science, Carnegie Mellon University, Microsoft Research, and IIT Kanpur

**Motivation**

Word Embeddings have been widely adopted across several NLP applications. However, most of the existing methods utilize sequential context of a word to learn its representation.

In this work, we explore to utilize syntactic context of words for learning word embeddings using recently proposed Graph Convolutional Networks. Also, we propose a more effective way for incorporating semantic knowledge like synonyms, hyponyms in learned embeddings.

**Graph Convolutional Networks**

GCNs are a generalization of Convolutional Neural Networks for non-Euclidean data. In this work, we utilize first-order approximation of GCNs (Kipf et al. 2016). The update equation for node A in the graph is given as:

\[ h_v = f\left(\sum_{u:x_u \neq 0} W x_u + b\right). \]

**Contributions**

1. We propose SynGCN, a Graph Convolution based method for learning word embeddings. Unlike the previous methods, SynGCN utilizes syntactic context for learning word representations without increasing vocabulary size.

2. We also present SemGCN, a framework for incorporating diverse semantic knowledge in learned word embeddings, without requiring relation-specific special handling as in previous methods.

3. Through experiments on multiple intrinsic and extrinsic tasks, we demonstrate substantial improvement over state-of-the-art approaches, and also yield an advantage when used in conjunction with methods such as ELMo.

**SynGCN Overview**

For each word in the sentence, SynGCN learns its representation by aiming to predict the word based on its dependency context encoded using GCNs defined as:

\[ h_{v+1} = f\left(\sum_{i \in V(s_i)} g_{t_i}^{h} \times W^{k}_{h} h_{i}, h_{j}\right) \]  \[ h_{v} = \sum_{e \leq j \leq c, j \neq 0} h_{j}, \]

**Results**

**SynGCN Evaluation:** Performance comparison on multiple intrinsic and extrinsic tasks. Overall, we observe that SynGCN outperforms all the existing word embeddings methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>Concept Cat</th>
<th>SynVerb+</th>
<th>SynVerb-</th>
<th>SynVerb+ &amp; SynVerb-</th>
<th>SynVerb+ &amp; SynVerb-</th>
</tr>
</thead>
<tbody>
<tr>
<td>CBOW</td>
<td>79.6</td>
<td>80.2</td>
<td>80.4</td>
<td>80.4</td>
<td>80.4</td>
</tr>
<tr>
<td>SynGCN</td>
<td>80.4</td>
<td>81.6</td>
<td>82.0</td>
<td>82.0</td>
<td>82.0</td>
</tr>
<tr>
<td>SemGCN</td>
<td>81.6</td>
<td>82.0</td>
<td>82.4</td>
<td>82.4</td>
<td>82.4</td>
</tr>
<tr>
<td>ELMo</td>
<td>80.0</td>
<td>81.0</td>
<td>81.4</td>
<td>81.4</td>
<td>81.4</td>
</tr>
</tbody>
</table>

**SemGCN Overview**

SemGCN allows to incorporate both symmetric (e.g. synonyms) and asymmetric (e.g. hypernyms) semantic knowledge into learned word embeddings. Unlike SynGCN, SemGCN operates on a corpus-level directed labeled graph.

Formally, we aim to maximize the following in both the models:

\[ E = \sum_{i=1}^{[V]} (v_{w_i} h_{j} - \log \sum_{i=1}^{[V]} \exp(v_{w_i} h_{j})), \]

where \( h_{j} \) is the GCN representation of the target word \( w_i \) and \( v_{w_i} \) is its target embedding.

**Ablation Results and Performance with ELMo**

SemGCN gives considerable improvement on SQuAD dataset compared to other methods when provided with the same semantic information (synonyms) for fine-tuning SynGCN embeddings.

<table>
<thead>
<tr>
<th>Method</th>
<th>POS</th>
<th>SquAD</th>
<th>NER</th>
<th>Coref</th>
</tr>
</thead>
<tbody>
<tr>
<td>CBOW</td>
<td>79.0</td>
<td>82.0</td>
<td>82.2</td>
<td>82.2</td>
</tr>
<tr>
<td>SynGCN</td>
<td>80.2</td>
<td>81.6</td>
<td>81.8</td>
<td>81.8</td>
</tr>
<tr>
<td>SemGCN</td>
<td>81.6</td>
<td>82.0</td>
<td>82.4</td>
<td>82.4</td>
</tr>
<tr>
<td>ELMo</td>
<td>80.0</td>
<td>81.0</td>
<td>81.4</td>
<td>81.4</td>
</tr>
</tbody>
</table>

**Acknowledgement**

This work is supported in part by the Ministry of Human Resource Development (Government of India) and Google PhD Fellowship.

**Source Code**

Source code is available at: [github/malllabiisc/WordGCN](https://github.com/malllabiisc/WordGCN)

Contact: shikhar@iisc.ac.in