Relational Graph Attention Network for Aspect-based Sentiment Analysis

-- Stargazers

CNN

- Convolutional Neural Networks (CNNs) have been successfully applied to tackle problems such as image classification, semantic segmentation or machine translation, where the underlying data representation has a grid-like structure.
- These architectures efficiently reuse their local filters, with learnable parameters, by applying them to all the input positions.
- However, many interesting tasks involve data that cannot be represented in a grid-like structure and that instead lies in an irregular domain.
- This is the case of 3D meshes, social networks, telecommunication networks, biological networks or brain connectomes. Such data can usually be represented in the form of graphs.

GNN

- There have been several attempts to extend neural networks to deal with arbitrarily structured graphs.
- Early work used recursive neural networks to process data represented in graph domains as directed acyclic graphs.
- Graph Neural Networks (GNNs) were introduced as a generalization of recursive neural networks that can directly deal with a more general class of graphs, e.g. cyclic, directed and undirected graphs.
- GNNs consist of an iterative process, which propagates the node states until equilibrium; followed by a neural network, which produces an output for each node based on its state.
- In the past few years, different variants of Graph Neural Networks are being developed with Graph Convolutional Networks (GCN) being one of them.

GCN

- There is an increasing interest in generalizing convolutions to the graph domain.
- Advances in this direction are often categorized as spectral approaches and non-spectral approaches.
- **Spectral Approaches**
-
    - Spectral approaches work with a spectral representation of the graphs and have been successfully applied in the context of node classification.
    - In Spectral approach the learned filters depend on the Laplacian eigen-basis, which depends on the graph structure. Thus, a model trained on a specific structure can not be directly applied to a graph with a different structure.
- **Non Spectral Approaches**
    - Non spectral approaches applies convolutions directly on the graph, operating on groups of spatially close neighbours.
    - One of the challenges of these approaches is to define an operator which works with different sized neighbourhoods and maintains the weight sharing property of CNNs.

Graph Attention Network(GAT)

- Dependency tree can be represented by a graph G with n nodes, where each represents a word in the sentence.
- The edges of G denote the dependency between words. The neighborhood nodes of node i can be represented by N_i .
- GAT iteratively updates each node representation (e.g., word embeddings) by aggregating neighborhood node representations using multi-head attention.

Graph Attention

$$h_{att_i}^{l+1} = ||_{k=1}^{K} \sum_{j \in \mathcal{N}_i} \alpha_{ij}^{lk} W_k^l h_j^l$$

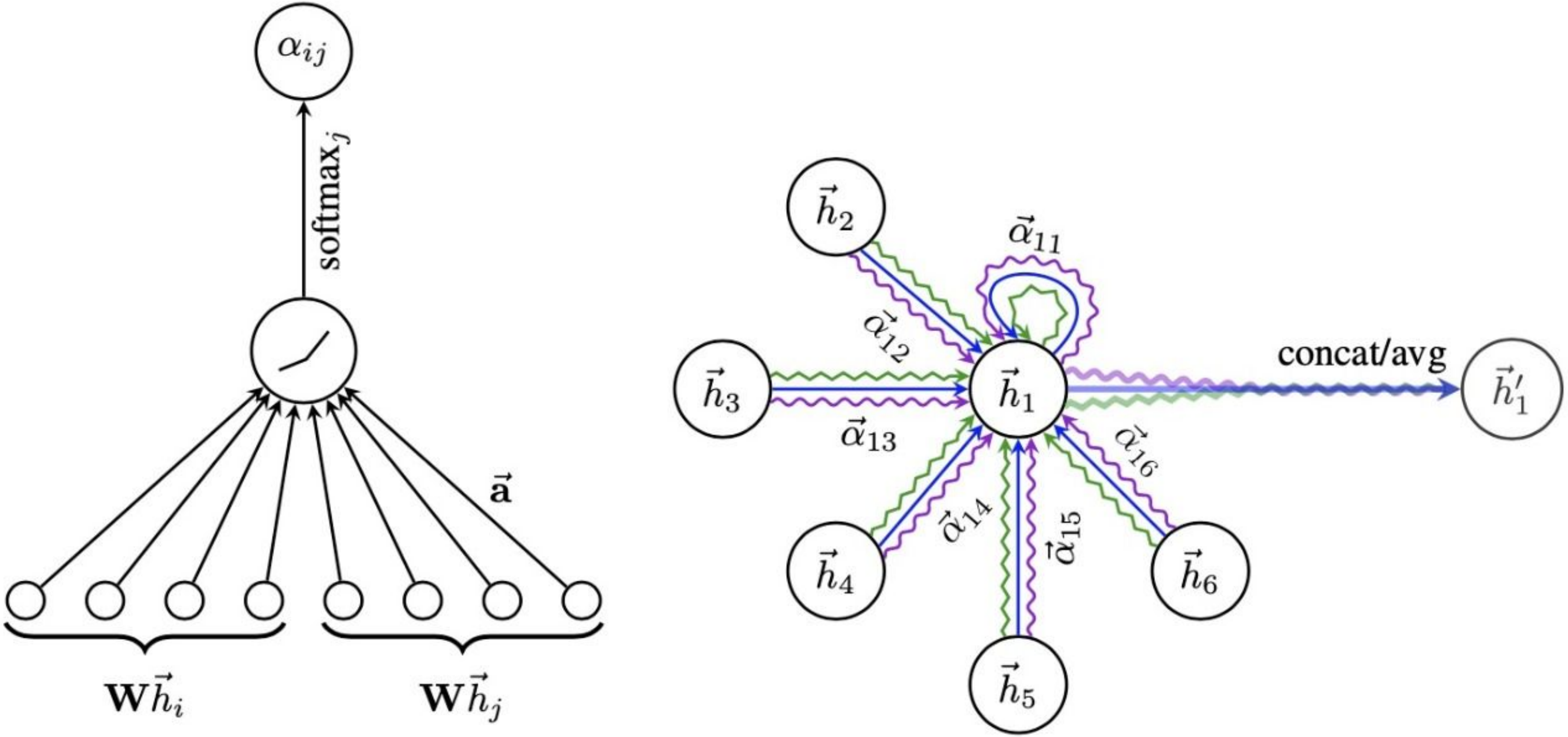
$$\alpha_{ij}^{lk} = attention(i, j)$$

$$z_i^{(l)} = W^{(l)} h_i^{(l)}, \tag{1}$$

$$e_{ij}^{(l)} = \text{LeakyReLU}(\vec{a}^{(l)^T}(z_i^{(l)}||z_j^{(l)})), \tag{2}$$

$$\alpha_{ij}^{(l)} = \frac{\exp(e_{ij}^{(l)})}{\sum_{k\in\mathcal{N}(i)} \exp(e_{ik}^{(l)})}, \tag{3}$$

$$h_i^{(l+1)} = \sigma\left(\sum_{j\in\mathcal{N}(i)} \alpha_{ij}^{(l)} z_j^{(l)}\right), \tag{4}$$



α_{ij}
softmax$_j$
$\vec{\mathbf{a}}$
$\mathbf{W}\vec{h}_i$
$\mathbf{W}\vec{h}_j$
$\vec{h}_2$
$\vec{\alpha}_{11}$
$\vec{\alpha}_{12}$
$\vec{h}_3$
$\vec{h}_1$
concat/avg
$\vec{h}'_1$
$\vec{\alpha}_{13}$
$\vec{\alpha}_{16}$
$\vec{\alpha}_{14}$
$\vec{\alpha}_{15}$
$\vec{h}_4$
$\vec{h}_6$
$\vec{h}_5$

Shortcomings from GAT

- GAT aggregates the representations of neighborhood nodes along the dependency paths.
- However, this process fails to take dependency relations into consideration, which may lose some important dependency information.
- Intuitively, neighborhood nodes with different dependency relations should have different influences.

Relational Graph Attention Network(RGAT)

- RGAT extend the original GAT with additional relational heads.
- These relational heads are used as relation-wise gates to control information flow from neighborhood nodes.
- Specifically, we first map the dependency relations into vector representations, and then compute a relational head.

Relational Graph Attention

$$h_{rel_i}^{l+1} = ||_{m=1}^{M} \sum_{j \in \mathcal{N}_i} \beta_{ij}^{lm} W_m^l h_j^l$$

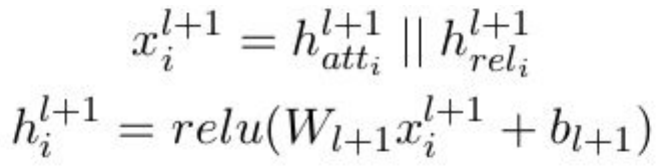
$$g_{ij}^{lm} = \sigma(relu(r_{ij} W_{m1} + b_{m1}) W_{m2} + b_{m2})$$

$$\beta_{ij}^{lm} = \frac{exp(g_{ij}^{lm})}{\sum_{j=1}^{\mathcal{N}_i} exp(g_{ij}^{lm})}$$

where r_{ij} represents relational embedding between nodes i and j

RGAT

R-GAT contains K attentional heads and M relational heads. The final representation of each node is computed by :



$$x_i^{l+1} = h_{att_i}^{l+1} \mid\mid h_{rel_i}^{l+1}$$

$$h_i^{l+1} = relu(W_{l+1} x_i^{l+1} + b_{l+1})$$

LSTM

- RNN Suffers from short-term memory. They leave out important information while processing of paragraph of Text.
- LSTM are created as solution of this problem.
- LSTM stands for Long short term memory.
- LSTM have internal gates that can regulate which data to keep and which to throw.

BiLSTM

- These are the variants of LSTM.
- In Bi directional LSTM, the learning algorithm with original data is feed once from beginning to end and once from end to beginning.
- BiLSTM effectively increase the amount of information hold by the network.
- BiLSTM also increase the context available to the network.

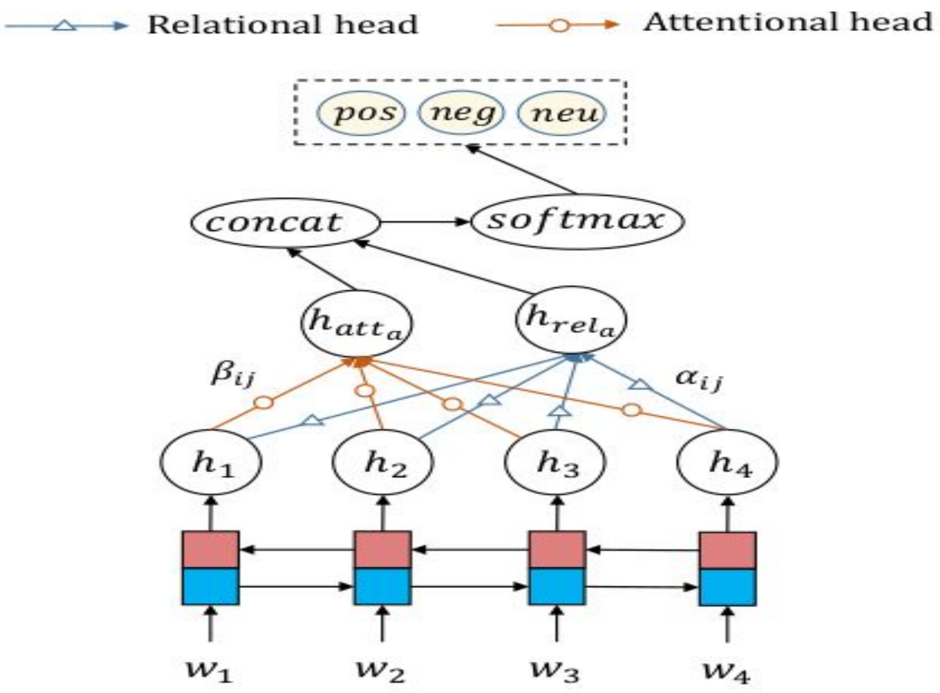
Training Model

- BiLSTM is used to encode the word embeddings of tree nodes, and obtain its output hidden state h_i for the initial representation h^0_i of leaf node i.
- Another BiLSTM is applied to encode the aspect words, and its average hidden state is used as the initial representation h^0_a of this root.
- After applying R-GAT on an aspect-oriented tree, its root representation h^l_a is passed through a fully connected softmax layer and mapped to probabilities over the different sentiment polarities.
- Finally, the standard cross-entropy loss is used as objective function.

Training Model

$$p(a) = softmax(W_p h_a^l + b_p)$$

$$L(\theta) = - \sum_{(S,A)\in\mathcal{D}} \sum_{a\in A} \log p(a)$$


Relational head
Attentional head
pos
neg
neu
concat
softmax
h_{att_a}
h_{rel_a}
β_{ij}
α_{ij}
h_1
h_2
h_3
h_4
w_1
w_2
w_3
w_4