论文阅读笔记

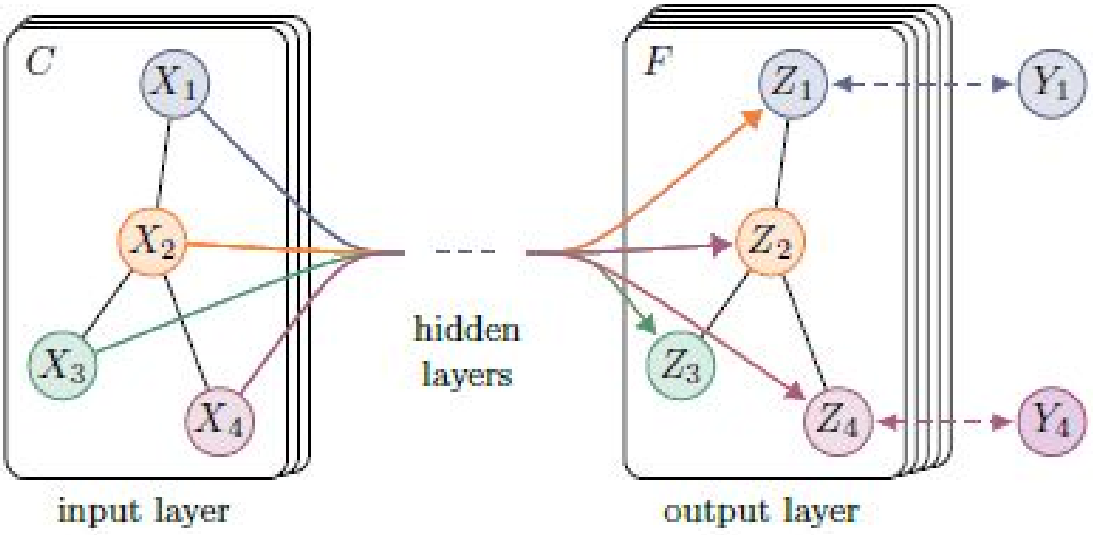
Step8

MF1833063, 史鹏, spwannasing@gmail.com

2019 年 9 月 5 日

1 SEMI-SUPERVISED CLASSIFICATION WITH GRAPH CONVOLUTIONAL NETWORKS

本文提出了一种图卷积网络（graph covolutional networks, GCNs），该网络是传统卷积算法在图结构数据上的一个变体，可以直接用于处理图结构数据。从本质上讲，GCN 是谱图卷积（spectral graph convolution）的局部一阶近似（localized first-order approximation）。GCN的另一个特点在于其模型规模会随图中边的数量的增长而线性增长。总的来说，GCN 可以用于对局部图结构与节点特征进行编码。



图卷积神经网络的（单层）最终形式： $H^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right)$

2 Semantic-Unit-Based Dilated Convolution for Multi-Label Text Classification

本文是基于seq2seq的多标签文本分类在attention上的一些改进工作。主要贡献有两点：

1.提出了所谓的“语义单元”，因为在多标签文本分类中，word-level的作用没有那么大，而是“semantic units”来决定文本的分类。

2.使用了Hybrid Attention将semantic units和word level的attention混合起来。

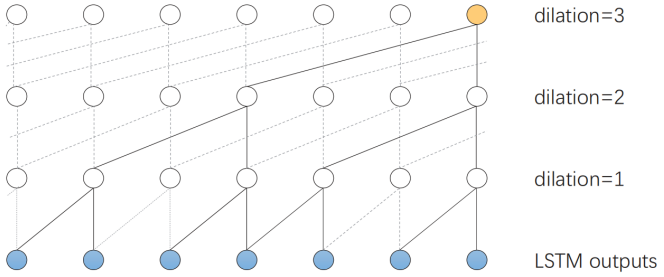


Figure 1: **Structure of Multi-level Dilated Convolution (MDC).** A example of MDC with kernel size $k = 2$ and dilation rates $[1, 2, 3]$. To avoid gridding effects, the dilation rates do not share a common factor other than 1.

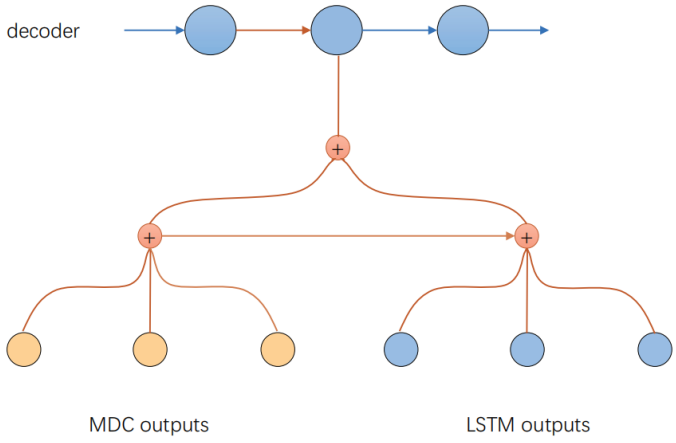


图 1: Structure of Hybrid Attention.

3 A Deep Reinforced Sequence-to-Set Model for Multi-Label Classification

本文的主要Motivation是解决多标签文本分类中Seq2Seq模型的输出序列的顺序问题，因为label之间本来应该是无序的，即交换不变性，这里所做的工作则是提出用强化学习来解决label之间的顺序问题，reward即为预测label序列和答案之间的F1值。

$$\mathcal{L}(\theta) = -\mathbb{E}_{\boldsymbol{y}\sim p_\theta}[r(\boldsymbol{y})] \tag{3.1}$$

$$\nabla_\theta \mathcal{L}(\theta) \approx -\left[r\left(\boldsymbol{y}^s\right) - r\left(\boldsymbol{y}^g\right)\right] \nabla_\theta \log\left(p_\theta\left(\boldsymbol{y}^s\right)\right) \tag{3.2}$$

$$r(\boldsymbol{y}) = \mathrm{F}_1\left(\boldsymbol{y}, \boldsymbol{y}^*\right) \tag{3.3}$$

其它的结构基本一致。

4 Compositional Questions Do Not Necessitate Multi-hop Reasoning

这篇文章主要是提出了在HotPotQA数据集中存在的一个问题：所谓的multi-hop其实不是必要的，大多数问题可以在只提供sigle paragraph的情况下就正确的回答出来。然后提出了一个Single-Paragraph QA模型。

分别将paragraph送入Bert

$$S' = \mathrm{BERT}(S) \in \mathbb{R}^{h\times(m+n+1)} \tag{4.1}$$

然后选取y_{empty} 最小的作为答案输出。

$$[y_{\text{span}}\,; y_{\text{yes}}\,; y_{\text{no}}\,; y_{\text{empty}}\,] = W_1\ \text{maxpool}\ (S') \tag{4.2}$$

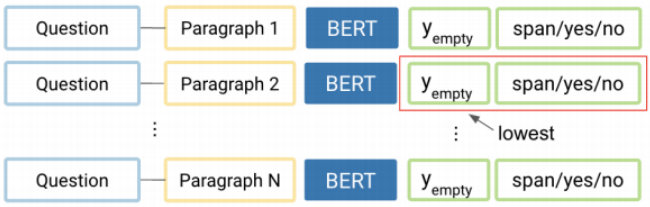


Figure 2: Our model, single-paragraph BERT, reads and scores each paragraph independently. The answer from the paragraph with the lowest y_{empty} score is chosen as the final answer.

5 Attention Guided Graph Convolutional Networks for Relation Extraction

motivation是现有的方法采用基于规则的硬剪枝策略来选择相关的部分依赖结构，但并不总是能得到最优的结果。本文提出了使用GCN来soft-pruning自动学习如何有选择地处理对关系提取任务有用的相关子结构。被称为Attention Guided Graph Convolutional Networks。

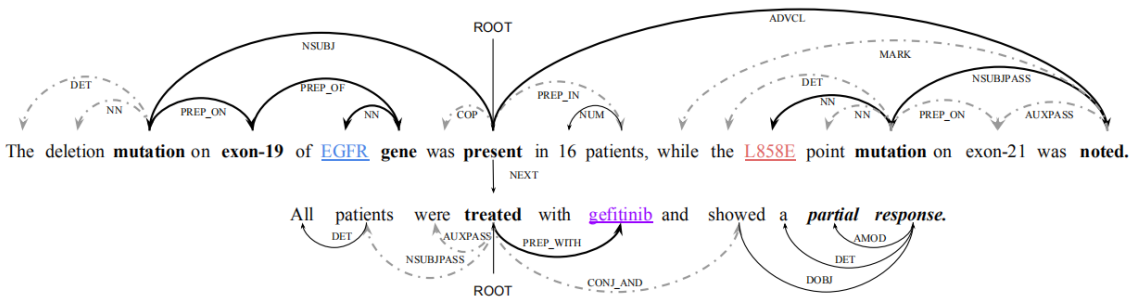


Figure 1: An example dependency tree for two sentences expressing a relation (sensitivity) among three entities. The shortest dependency path between these entities is highlighted in bold (edges and tokens). The root node of the LCA subtree of entities is *present*. The dotted edges indicate tokens K=1 away from the subtree. Note that tokens *partial response* off these paths (shortest dependency path, LCA subtree, pruned tree when K=1).

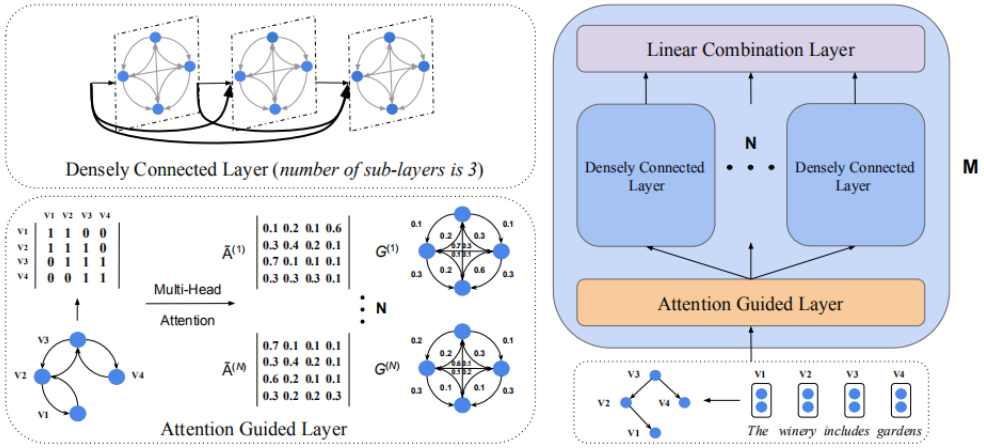


Figure 2: The AGGCN model is shown with an example sentence and its dependency tree. It is composed of M identical blocks and each block has three types of layers as shown on the right. Every block takes node embeddings and adjacency matrix that represents the graph as inputs. Then N attention guided adjacency matrices are constructed by using multi-head attention as shown at bottom left. The original dependency tree is transformed into N different fully connected edge-weighted graphs (self-loops are omitted for simplification). Numbers near the edges represent the weights in the matrix. Resulting matrices are fed into N separate densely connected layers, generating new representations. Top left shows an example of the densely connected layer, where the number (L) of sub-layers is 3 (L is a hyper-parameter). Each sub-layer concatenates all preceding outputs as the input. Eventually, a linear combination is applied to combine outputs of N densely connected layers into hidden representations.

GCNs：

$$\mathbf{h}_i^{(l)} = \rho\left(\sum_{j=1}^{n} \mathbf{A}_{ij}\mathbf{W}^{(l)}\mathbf{h}_j^{(l-1)} + \mathbf{b}^{(l)}\right) \tag{5.1}$$

Attention Guided Layer：

$$\tilde{\mathbf{A}}^{(\mathrm{t})} = \mathrm{softmax}\left(\frac{Q\mathbf{W}_i^Q \times \left(K\mathbf{W}_i^K\right)^T}{\sqrt{d}}\right) \tag{5.2}$$

Densely Connected Layer：

$$\mathbf{g}_j^{(l)} = \left[\mathbf{x}_j; \mathbf{h}_j^{(1)}; \ldots; \mathbf{h}_j^{(l-1)}\right] \tag{5.3}$$

$$\mathbf{h}_{t_i}^{(l)} = \rho\left(\sum_{j=1}^{n} \tilde{\mathbf{A}}_{ij}^{(t)} \mathbf{W}_t^{(l)} \mathbf{g}_j^{(l)} + \mathbf{b}_t^{(l)}\right) \tag{5.4}$$

Linear Combination Layer:

$$h_{comb} = W_{comb} h_{out} + b_{comb}$$

$$h_{out} = [h^{(1)}; ...; h^{(n)}]$$

6 Learning a Deep ConvNet for Multi-label Classification with Partial Labels

本文解决的问题是只有Partial Label的多标签分类问题。贡献点一个是提出了一个新的针对partial的损失函数，其次使用GNN来对multi-label之间的关系进行建模，最后还能通过此算法来对没有提供label的那一部分进行预测。

$$l(x,y) = \frac{g(p_y)}{C}\sum_{c=1}^{C}[1_{[y_c=1]}log(\frac{1}{1+exp(-x_c)}) + 1_{[y_c=-1]}log(\frac{exp(-x_c)}{1+exp(-x_c)})]$$

这是对应的损失函数，其中p_y是该条数据中，有标签的label所占的比例，g是正则化函数。

GNN用于Multi-Label：

Message update：

$$m_v^t = \frac{1}{|\Omega_v|}\sum_{u\in\Omega_v} f_M(h_u^t)$$

Hidden state update:

$$h_v^{t+1} = GRU(h_v^t, m_b^t)$$

7 Inferential Machine Comprehension: Answering Questions by Recursively Deducing the Evidence Chain from Text