Document-Level Relation Extraction with Adaptive Thresholding and Localized Context Pooling

Wenxuan Zhou[1], Kevin Huang[2], Tengyu Ma[3], Jing Huang[2]
[1]University of Southern California [2]JD AI Research [3]Stanford University
[1]zhouwenx@usc.edu [2]{kevin.huang, jing.huang}@jd.com [3]tengyuma@stanford.edu

收录：Arxiv2020

动机：Graph-based model和transformer-based model都有提取长距离信息的作用，存在重叠且简单使用transformer-based model给出的embedding会导致不同的实体对中使用的entity embedding是相同的，故需改进；之前的方法在预测时使用Global Thresholding，较为机械且影响准确度

方法：使用额外的上下文信息提升entity embedding，额外的上下文信息是通过充分利用BERT中的Transformer信息（代替Graph-based model ）得到的，这样既解决了Graph-based model和transformer-based model潜在的重叠问题，又使不同实体对预测中使用的entity embedding不同；预测时使用Adaptive Thresholding代替Global Thresholding，提高准确度

Entity Encoder

$$[\boldsymbol{h}_1, \boldsymbol{h}_2, ..., \boldsymbol{h}_l] = \mathrm{BERT}([x_1, x_2, ..., x_l]).$$

$$\boldsymbol{h}_{e_i} = \log \sum_{j=1}^{N_{e_i}} \exp\left(\boldsymbol{h}_{m_j^i}\right).$$

Entity Localized Context Pooling

$$\boldsymbol{A}^{(s,o)} = \boldsymbol{A}_s^E \cdot \boldsymbol{A}_o^E,$$

$$\boldsymbol{q}^{(s,o)} = \sum_{i=1}^{H} \boldsymbol{A}_i^{(s,o)},$$

$$\boldsymbol{a}^{(s,o)} = \boldsymbol{q}^{(s,o)} / \mathbf{1}^\intercal \boldsymbol{q}^{(s,o)},$$

$$\boldsymbol{c}^{(s,o)} = \boldsymbol{H}^\intercal \boldsymbol{a}^{(s,o)},$$

将BERT中最后一层Transformer的Attention矩阵记为A_{H*L*L}，其中H是Attention头数，L是passage中总词数，在第二个维度上，按照某个entity mention是passage中第几个词进行截取，得到矩阵A_{H*L}，将同一个entity的所有mention对应位置截取到的矩阵进行平均得到A^E

Entity Embedding

$$\boldsymbol{z}_s^{(s,o)} = \tanh\left(\boldsymbol{W}_s \boldsymbol{h}_{e_s} + \boldsymbol{W}_{c_1} \boldsymbol{c}^{(s,o)}\right),$$

$$\boldsymbol{z}_o^{(s,o)} = \tanh\left(\boldsymbol{W}_o \boldsymbol{h}_{e_o} + \boldsymbol{W}_{c_2} \boldsymbol{c}^{(s,o)}\right),$$

Classifier

$$\mathrm{P}\left(r|e_s, e_o\right) = \sigma\left(\boldsymbol{z}_s^\intercal \boldsymbol{W}_r \boldsymbol{z}_o + b_r\right),$$

Adaptive Thresholding

将Thresholding视为一个类，以DocRED为例，原DocRED共96类，现在将Thresholding 加入作为一类，共97类，为这97类均进行上述Classifier的计算，通过训练得到Adaptive Thresholding的值

Adaptive Thresholding Loss

$$\mathcal{L}_1 = -\sum_{r \in \mathcal{P}_T} \log\left(\frac{\exp\left(\mathrm{logit}_r\right)}{\sum_{r' \in \mathcal{P}_T \cup \{\mathrm{TH}\}} \exp\left(\mathrm{logit}_{r'}\right)}\right),$$

$$\mathcal{L}_2 = -\log\left(\frac{\exp\left(\mathrm{logit}_{\mathrm{TH}}\right)}{\sum_{r' \in \mathcal{N}_T \cup \{\mathrm{TH}\}} \exp\left(\mathrm{logit}_{r'}\right)}\right),$$

$$\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2.$$

Model	Dev		Test	
	Ign F_1	F_1	Ign F_1	F_1
Sequence-based Models				
CNN (Yao et al., 2019)	41.58	43.45	40.33	42.26
BiLSTM (Yao et al., 2019)	48.87	50.94	48.78	51.06
Graph-based Models				
BiLSTM-AGGCN (Guo et al., 2019)	46.29	52.47	48.89	51.45
BiLSTM-LSR (Nan et al., 2020)	48.82	55.17	52.15	54.18
BERT-LSR$_{\text{BASE}}$ (Nan et al., 2020)	52.43	59.00	56.97	59.05
Transformer-based Models				
BERT$_{\text{BASE}}$ (Wang et al., 2019b)	-	54.16	-	53.20
BERT-TS$_{\text{BASE}}$ (Wang et al., 2019b)	-	54.42	-	53.92
HIN-BERT$_{\text{BASE}}$ (Tang et al., 2020a)	54.29	56.31	53.70	55.60
CorefBERT$_{\text{BASE}}$ (Ye et al., 2020)	55.32	57.51	54.54	56.96
CorefRoBERTa$_{\text{LARGE}}$ (Ye et al., 2020)	57.84	59.93	57.68	59.91
Our Methods				
BERT$_{\text{BASE}}$ (our implementation)	54.27 ± 0.28	56.39 ± 0.18	-	-
BERT-E$_{\text{BASE}}$	56.51 ± 0.16	58.52 ± 0.19	-	-
BERT-ATLOP$_{\text{BASE}}$	59.22 ± 0.15	61.09 ± 0.16	59.31	61.30
RoBERTa-ATLOP$_{\text{LARGE}}$	$\mathbf{61.32 \pm 0.14}$	$\mathbf{63.18 \pm 0.19}$	**61.39**	**63.40**