Two are Better than One: Joint Entity and Relation Extraction with Table-Sequence Encoders

Jue Wang[1] and **Wei Lu**[2]
[1]College of Computer Science and Technology, Zhejiang University
[2]StatNLP Research Group, Singapore University of Technology and Design
zjuwangjue@zju.edu.cn, luwei@sutd.edu.sg

EMNLP 2020（句子级 NER 和 RE 联合抽取模型）

本文的模型是 table filling 的方法，并且作者关于传统的 table filing 方法指出了两个缺陷。
1、仅仅用一个特征就解决 NER 和 RE 的问题，会造成 feature confusion 的问题
2、以前的方法并未充分利用 table 的结构信息
因此作者基于此两点，提出了两个 encoder 架构，一个用于编码句子，一个用于用于编码 table，并且两个 encoder 之间会有信息的交互，也就是计算 table 的特征会使用句子的特征，计算句子的特征会使用 table 的特征。如此交替更迭多层，得到最终的 table 和句子的特征，分别用于关系分类和实体识别

Text encoder：
1、每个词使用 word embeddings，character embeddings 以及 LSTM 的拼接表示，并且经过一个线性变化

$$\boldsymbol{S}_0 = \text{Linear}([\boldsymbol{x}^c; \boldsymbol{x}^w; \boldsymbol{x}^\ell]) \qquad (1)$$

Table encoder：
1、将全部 token 两两组合，初始化一个表，并经过一个线性变换和激活函数

$$X_{l,i,j} = \text{ReLU}(\text{Linear}([S_{l-1,i}; S_{l-1,j}])) \qquad (2)$$

2、将得到的初始化表格经过 MD-RNN 网络（GRU），得到表格每个单元格的最终表示

$$T_{l,i,j} = \text{GRU}(X_{l,i,j}, T_{l-1,i,j}, T_{l,i-1,j}, T_{l,i,j-1}) \quad (3)$$

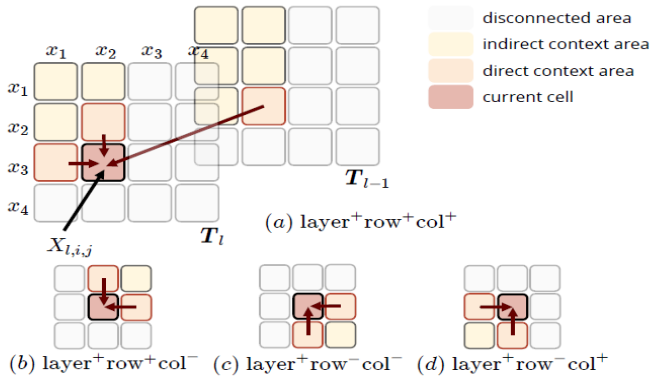
上述中的 Ti-1,j，Ti,j-1 代表了与当前表格相邻的表格的特征（上下左右四个方向）

在后面的论述中，作者会引入 BERT 中的 attention 矩阵，因此会更新公式（2），其他不变

$$X_{l,i,j} = \text{ReLU}(\text{Linear}([S_{l-1,i}; S_{l-1,j}; T^\ell_{i,j}])) \quad (11)$$

其中的 T 为 BERT 中的 attention 矩阵，也就是 BERT 中两个词之间的注意力得分，作者将其取出来，融入到 table encoder 中

下图为模型的图形化解释:



3、作者发现四个方向（图上的 a,b,c,d）汇聚来的信息不如两个方向（图上的 a,c）汇聚来的信息效果好，因此作者最终只取 a,c 方向，并将最后结果拼接在一起

$$T^{(a)}_{l,i,j} = \mathrm{GRU}^{(a)}(X_{l,i,j}, T^{(a)}_{l-1,i,j}, T^{(a)}_{l,i-1,j}, T^{(a)}_{l,i,j-1}) \quad (4)$$

$$T^{(c)}_{l,i,j} = \mathrm{GRU}^{(c)}(X_{l,i,j}, T^{(c)}_{l-1,i,j}, T^{(c)}_{l,i+1,j}, T^{(c)}_{l,i,j+1}) \quad (5)$$

$$T_{l,i,j} = [T^{(a)}_{l,i,j}; T^{(c)}_{l,i,j}] \quad (6)$$

Sequence encoder:

1、需要对每个句子进行 attention 计算，但是作者在这里对 self-attention 的计算式进行了一个变动，并未使用句子 X 句子来计算，而是直接使用 table encoder 的结果作为注意力矩阵

$$f(Q_i, K_j) = U \cdot T_{l,i,j} \quad (8)$$

2、然后将注意力矩阵与句子相乘得到融入了 table 信息的表示，接下来与 transformer 的 encoder 结构一样，进行 add 和 norm，再经过一个 FFN 后进行 add 和 norm，得到最终的句子表示

$$\tilde{\boldsymbol{S}}_l = \mathrm{LayerNorm}(\boldsymbol{S}_{l-1} + \mathrm{SelfAttn}(\boldsymbol{S}_{l-1})) \quad (9)$$

$$\boldsymbol{S}_l = \mathrm{LayerNorm}(\tilde{\boldsymbol{S}}_l + \mathrm{FFNN}(\tilde{\boldsymbol{S}}_l)) \quad (10)$$

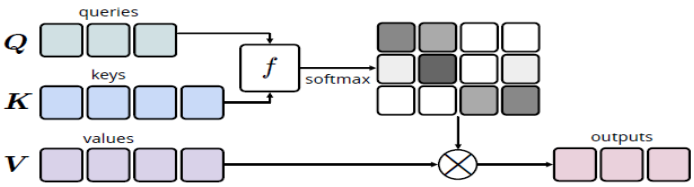


Figure 5: The generalized form of attention. The softmax function is used to normalize the weights of values $\boldsymbol{V}$ for each query Q_i.

上图为计算 self-attention 的一般步骤，只不过作者将 Q 与 K 相乘的结果直接由 table 的表示替换掉，不仅节省了时间，而且与实现了句子特征与 table 特征的交互，因为本身 table 就是一个方阵，里面的每一个表格可以代表为 token 之间的相关性

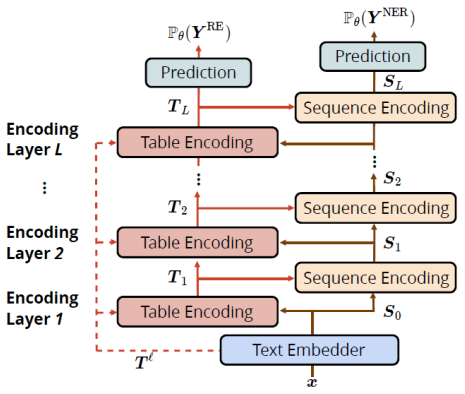
解码器:

1、对 table 的特征进行关系分类，对 sequence 的特征进行实体识别

$$P_\theta(\boldsymbol{Y}^{\mathrm{NER}}) = \mathrm{softmax}(\mathrm{Linear}(\boldsymbol{S}_L)) \qquad (12)$$

$$P_\theta(\boldsymbol{Y}^{\mathrm{RE}}) = \mathrm{softmax}(\mathrm{Linear}(\boldsymbol{T}_L)) \qquad (13)$$

模型结构:



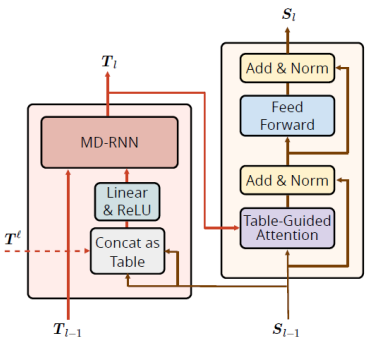


Figure 3: A layer in the table-sequence encoders.

表格示例:

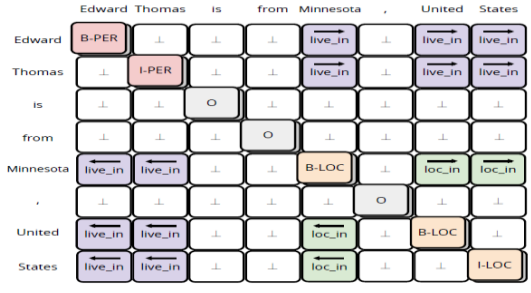


Figure 1: An example of table filling for NER and RE.

其中对角线上不表示关系，仅表示实体（B-I-O），其余正交符表示两个 token 无关系
作者在训练时会先识别实体，然后再对表格中的元素进行关系分类，最后在表格中找出已识别实体的各种关系

实验结果：

Data	Model	NER	RE	RE+
ACE04	Li and Ji (2014) ▽	79.7	48.3	45.3
	Katiyar and Cardie (2017) ▽	79.6	49.3	45.7
	Bekoulis et al. (2018b) ▽	81.2	-	47.1
	Bekoulis et al. (2018a) ▽	81.6	-	47.5
	Miwa and Bansal (2016) ▽	81.8	-	48.4
	Li et al. (2019) ▽	83.6	-	49.4
	Luan et al. (2019) ▽	87.4	59.7	-
	Ours ▽	**88.6**	**63.3**	**59.6**
ACE05	Li and Ji (2014) ▽	80.8	52.1	49.5
	Miwa and Bansal (2016) ▽	83.4	-	55.6
	Katiyar and Cardie (2017) ▽	82.6	55.9	53.6
	Zhang et al. (2017) ▽	83.6	-	57.5
	Sun et al. (2018) ▽	83.6	-	59.6
	Li et al. (2019) ▽	84.8	-	60.2
	Dixit and Al (2019) ▽	86.0	62.8	-
	Luan et al. (2019) ▽	88.4	63.2	-
	Wadden et al. (2019) ▽	88.6	63.4	-
	Ours ▽	**89.5**	**67.6**	**64.3**
CoNLL04	Miwa and Sasaki (2014)▽	80.7	-	61.0
	Bekoulis et al. (2018a)▲	83.6	-	62.0
	Bekoulis et al. (2018b)▲	83.9	-	62.0
	Tran and Kavuluru (2019)▲	84.2	-	62.3
	Nguyen and Verspoor (2019)▲	86.2	-	64.4
	Zhang et al. (2017)▽	85.6	-	67.8
	Li et al. (2019)▽	87.8	-	68.9
	Eberts and Ulges (2019)▽	88.9	-	71.5
	Eberts and Ulges (2019)▲	86.3	-	72.9
	Ours▽	**90.1**	**73.8**	**73.6**
	Ours▲	**86.9**	**75.8**	**75.4**
ADE	Li et al. (2016) ▲	79.5	-	63.4
	Li et al. (2017) ▲	84.6	-	71.4
	Bekoulis et al. (2018b) ▲	86.4	-	74.6
	Bekoulis et al. (2018a) ▲	86.7	-	75.5
	Tran and Kavuluru (2019) ▲	87.1	-	77.3
	Eberts and Ulges (2019) ▲	89.3	-	79.2
	Ours ▲	**89.7**	**80.1**	**80.1**

因为中间使用了 table 的特征来代替注意力矩阵，所以作者想要验证 table 的有效性，特意将其与 ALBERT 的 attention 矩阵进行了可视化对比，并说明了 table-attention 的有效性

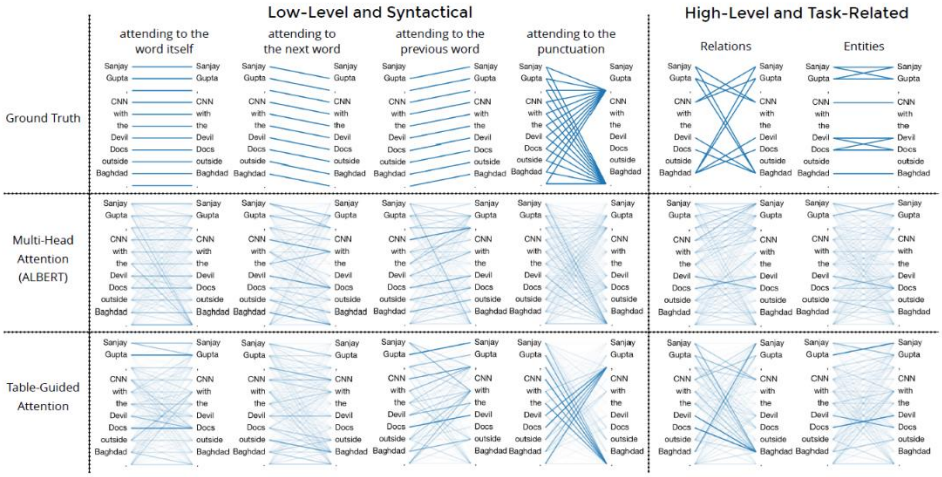


Figure 6: Comparison between ground truth and selected heads of ALBERT and table-guided attention. The sentence is randomly selected from the development set of ACE05.

作者发现，在 low-level 和 high-level 的情况下，table-attention 更能正确的捕获重要信息，将两个有关系的 token 联系的更紧密

总而言之，作者提出的双 encoder 架构是在前任 table filling 方法的基础上进行改进，利用了 sequence 的信息也利用了 table 的信息，并且通过将两者交互融合，得到了不错的结果，有比较好的借鉴意义，因为联合抽取模型是将两个任务联合在一起解决，变化为一个任务，因此 encoder 的特征好坏对于模型表现有很大影响，而以前的做法都是用一个 embedding 去进行 NER 和 RE，直觉来说实体的 embedding 与关系的 embedding 确实应该不同，但是又应该有交互，不然又变为 pipeline 的做法，忽略了实体和关系的联系。