Recurrent Interaction Network for Jointly Extracting Entities and Classifying Relations

Kai Sun
BDBC and SKLSDE
Beihang University,China
sunkai@buaa.edu.cn

Richong Zhang*
BDBC and SKLSDE
Beihang University, China
zhangrc@act.buaa.edu.cn

Samuel Mensah
BDBC and SKLSDE
Beihang University, China
samensah@buaa.edu.cn

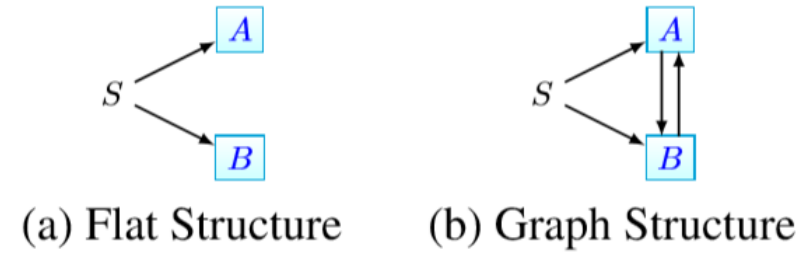
Yongyi Mao
School of EECS
University of Ottawa,Canada
ymao@uottawa.ca

Xudong Liu
BDBC and SKLSDE
Beihang University, Beijing, China
liuxd@act.buaa.edu.cn

EMNLP2020

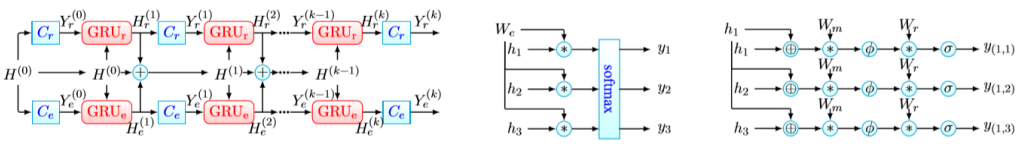
这篇做的也是joint，但是感觉思路不错，跟其他千篇一律的不相同。

之前做joint可能是类似pipeline然后loss sum起来一起训，或者是multi-task learning的方式share底层编码结果。但是这样做的后果就是NER跟RE两个task是implicit的学习到对方的feature，这篇paper的核心就是我要explicit来做，我要让他们的feature直接交互。



(a) Flat Structure (b) Graph Structure

思路是下面这样的。



(a) Framework of RIN. (b) The ER module (C_e). (c) The RC module (C_r).

Figure 2: (a) The framework of RIN. (b) The entitiy recognition module. (c) The relation classification module. In (b) and (c), we use a toy example of shared features $H = \{h_1, h_2, h_3\}$ to demonstrate the entity prediction for word w_i and relation prediction for all pairs $(w_1, w_1), (w_1, w_2), (w_1, w_3)$. $+$, $\oplus$, $*$, ϕ, and σ denote a summation operator, a concatenation operator, a matrix multiplication, relu activation function and sigmoid function respectively.

首先是通过encoder拿到整个句子的这个编码，相当于H是一个矩阵，每一列是

对应位置token的一个feature vector。
之后它被送进NER模块Ce以及RE模块Cr做预测，拿到了Y矩阵。
对于NER模块，它的Y矩阵每个列向量是预测出来的这个token在BIOES上的distributional probability。
对于RE模块，它的Y矩阵每个列向量是这个token与其他位置token的relation distribution的max pooling结果。

$$y_i = \text{maxpool}\left(Y_r(w_i)\right),$$

之后是各自通过GRU来进行更新，注意每个位置都是 自己更新，跟其他位置无关，自己只管自己！！！

$$\begin{aligned} z &= \sigma\left(W_{\text{z}}(h \oplus y)\right) \\ u &= \sigma\left(W_{\text{u}}(h \oplus y)\right) \\ \check{h} &= \tanh\left(W_{\text{o}}((u * h) \oplus y)\right) \\ h_e &= (1 - z) * h + z * \check{h} \end{aligned}$$

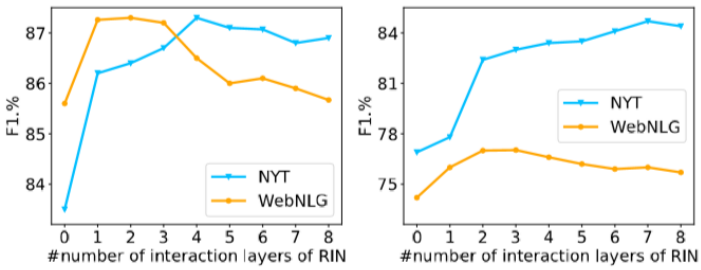
最后就是交叉熵训练了。
奇怪的是好像只用了最后的预测结果来训，中间那些Y并没有给监督信号。

实验是在NYT跟WebNLG上做的，两种评价方式是partial match跟exact match。
partial match只要求relation跟每个entity的head对了就行。
exact match要求relation跟每个entity的head和tail都对。

Evaluation	Model	NYT			WebNLG		
		Prec	Rec	F1	Prec	Rec	F1
Partial Match	OneDecoder	59.4	53.1	56.0	32.2	28.9	30.5
	MultiDecoder	61.0	56.6	58.7	37.7	36.4	37.1
	OrderRL	77.9	67.2	72.1	63.3	59.9	61.6
	$\text{RIN}_{\text{w/o interaction}}$	83.9±0.6	83.1±0.6	83.5±0.2	84.9±0.6	86.3±0.8	85.6±0.3
	RIN	**87.2±0.2**	**87.3±0.3**	**87.3±0.1**	**87.6±0.1**	**87.0±0.9**	**87.3±0.4**
Exact Match	NovelTagging	62.4	31.7	42.0	52.5	19.3	28.3
	GraphRel_{1p}	62.9	57.3	60.0	42.3	39.2	40.7
	GraphRel_{2p}	63.9	60.0	61.9	44.7	41.1	42.9
	CopyMLT-One	72.7	69.2	70.9	57.8	60.1	58.9
	CopyMLT-Mul	75.7	68.7	72.0	58.0	54.9	56.4
	$\text{RIN}_{\text{w/o interaction}}$	77.4±1.1	76.4±0.7	76.9±0.3	75.0±1.1	73.3±0.7	74.2±0.3
	RIN	**83.9±0.5**	**85.5±0.5**	**84.7±0.4**	**77.3±0.7**	**76.8±1.0**	**77.0±0.2**

Table 1: Precision, Recall and F1 performance of different models on the datasets. Results for the compared models are retrieved from their original papers. We report the mean results over five runs and the standard deviation. The best performance is bold-typed.

注意有一个很重要的超参数是GRU的更新次数K，paper也做了探究。



(A) Partial Match (B) Exact Match

Figure 3: Curves of F1 performance on different number of interaction layers K.

和一些ablation study

Model	NYT	WebNLG
RIN	84.7	77.0
$\text{RIN}_{\text{w/o ER}}$	83.9	76.4
$\text{RIN}_{\text{w/o RC}}$	77.3	76.0
$\text{RIN}_{\text{w/o interaction}}$	76.9	74.2
$\text{RIN}_{\text{w/o POS}}$	84.1	76.6

Table 5: F1 performance of different ablation models on the datasets. The Exact Match evaluation is used.

- w/o ER指的是H的更新不考虑ER的GRU output结果
- w/o RC同上，不考虑RC的
- w/o 表示没有interactionn

- w/o POS指的是input没有使用pos embedding