

Named Entity Recognition as Dependency Parsing

WHY?

- Flat NER
- 传统的命名实体识别方法（Flat NER）：抽取出 句子中可能为实体的所有元素，这些实体间不存在交集问题。

```
query: 感冒吃什么药物?  
NER 结果: {  
    感冒: 症状  
}
```

- 问题：但是现实中，所有句子所抽取出的实体都是没有交集的么？

WHY?

- Nested NER
- 比如下例 “腰间盘突出吃什么药物？”，NER 所 抽取出的 “腰间盘” 和 “腰间盘突出” 存在交集，或者说 “腰间盘” 包含于 “腰间盘突出”。此时，如果采用 Flat NER 进行处理，那么只能选取其中的一个。

```
query: 腰间盘突出吃什么药物?  
NER 结果: {  
  腰间盘: 部位,  
  腰间盘突出: 症状  
}
```

WHY?

- 识别出 嵌套实体 有什么用呢？
- 介绍：如果一个模型能够识别出 “腰椎间盘突出” 是一个症状，它倾向于将所有出现的 “腰椎间盘突出” 都标记成症状。但如果它能够在识别前者的同时将 “腰椎间盘” 标记成部位，我们就认为它有能力将所有 “[部位]突出” 的模式都识别出来，因为后者的角度，模型学到的是一种pattern，而非记住了一种具体情况。此外，提取出来的额外信息也能作为辅助特征，增强其他任务的效果。

HOW?

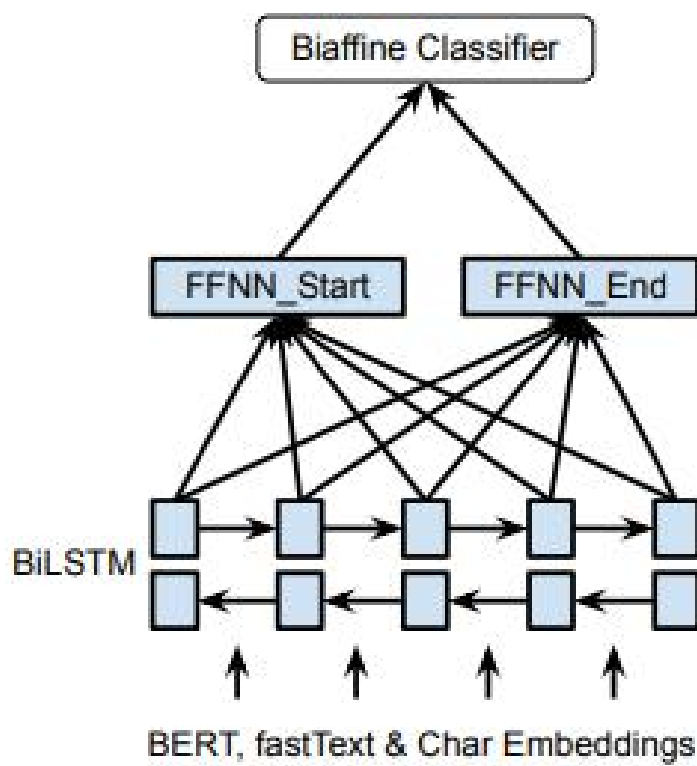


Figure 1: The network architectures of our system.

HOW?

1. embedding layer
2. BiLSTM: 获得单词表示形式;
3. FFNN: 应用两个单独的FFNN为 span 的开始/结束创建不同的表示形式 (hs / he)。对 span 的开始/结束使用不同的表示, 可使系统学会单独识别 span 的开始/结束;
4. biaffine model: 在句子上使用biaffine模型来创建 $1 \times 1 \times c$ 评分张量 (rm), 其中1是句子的长度, c 是 NER 类别的数量 +1 (对于非实体)。

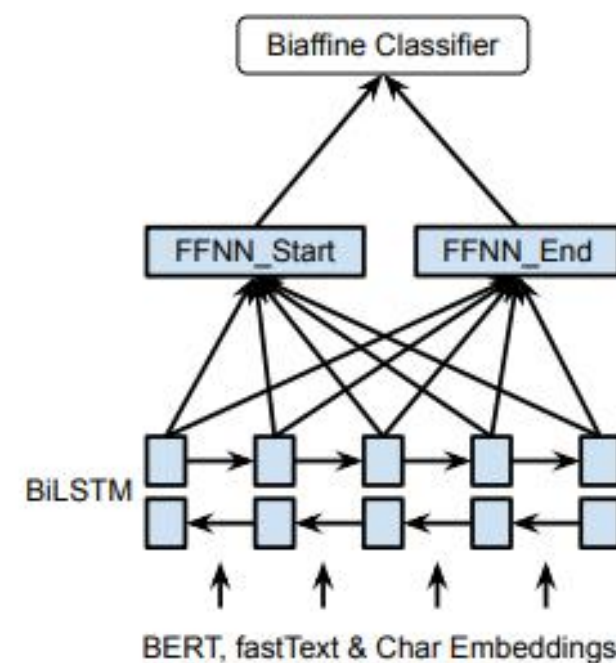


Figure 1: The network architectures of our system.

HOW?

5. loss 函数

因为该任务属于 多类别分类问题:

$$p_m(i_c) = \frac{\exp(r_m(i_c))}{\sum_{\hat{c}=1}^C \exp(r_m(i_{\hat{c}}))}$$

$$loss = - \sum_{i=1}^N \sum_{c=1}^C y_{i_c} \log p_m(i_c)$$

						1:部位	2:症状
呼	0	0	0	1	0	2	
吸	-1	0	0	0	0	0	
中	-1	-1	0	0	0	0	
枢	-1	-1	-1	0	0	0	
受	-1	-1	-1	-1	0	0	
累	-1	-1	-1	-1	-1	0	
...	呼	吸	中	枢	受	累	

(e) 多头标注

HOW?

6. focal_loss 损失函数

目标：解决分类问题中类别不平衡、分类难度差异的一个 loss；

思路：降低了大量简单负样本在训练中所占的权重，也可理解为一种困难样本挖掘。

损失函数形式：

$$L_{fl} = \begin{cases} -\alpha(1-y')^\gamma \log y' & , \quad y = 1 \\ -(1-\alpha)y'^\gamma \log(1-y'), & y = 0 \end{cases}$$

WHAT?

Nested NER

Model	P	R	F1
ACE 2004			
Katiyar and Cardie (2018)	73.6	71.8	72.7
Wang et al. (2018)	-	-	73.3
Wang and Lu (2018)	78.0	72.4	75.1
Straková et al. (2019)	-	-	84.4
Luan et al. (2019)	-	-	84.7
Our model	87.3	86.0	86.7
ACE 2005			
Katiyar and Cardie (2018)	70.6	70.4	70.5
Wang et al. (2018)	-	-	73.0
Wang and Lu (2018)	76.8	72.3	74.5
Lin et al. (2019)	76.2	73.6	74.9
Fisher and Vlachos (2019)	82.7	82.1	82.4
Luan et al. (2019)	-	-	82.9
Straková et al. (2019)	-	-	84.3
Our model	85.2	85.6	85.4
GENIA			
Katiyar and Cardie (2018)	79.8	68.2	73.6
Wang et al. (2018)	-	-	73.9
Ju et al. (2018)	78.5	71.3	74.7
Wang and Lu (2018)	77.0	73.3	75.1
Sohrab and Miwa (2018) ⁵	93.2	64.0	77.1
Lin et al. (2019)	75.8	73.9	74.8
Luan et al. (2019)	-	-	76.2
Straková et al. (2019)	-	-	78.3
Our model	81.8	79.3	80.5

Table 2: State of the art comparison on ACE 2004, ACE 2005 and GENIA corpora for nested NER.

Flat NER

Model	P	R	F1
ONTONOTES			
Chiu and Nichols (2016)	86.0	86.5	86.3
Strubell et al. (2017)	-	-	86.8
Clark et al. (2018)	-	-	88.8
Fisher and Vlachos (2019)	-	-	89.2
Our model	91.1	91.5	91.3
CONLL 2003 English			
Chiu and Nichols (2016)	91.4	91.9	91.6
Lample et al. (2016)	-	-	90.9
Strubell et al. (2017)	-	-	90.7
Devlin et al. (2019)	-	-	92.8
Straková et al. (2019)	-	-	93.4
Our model	93.7	93.3	93.5
CONLL 2003 German			
Lample et al. (2016)	-	-	78.8
Straková et al. (2019)	-	-	85.1
Our model	88.3	84.6	86.4
CONLL 2003 German revised ⁶			
Akbik et al. (2018)	-	-	88.3
Our model	92.4	88.2	90.3
CONLL 2002 Spanish			
Lample et al. (2016)	-	-	85.8
Straková et al. (2019)	-	-	88.8
Our model	90.6	90.0	90.3
CONLL 2002 Dutch			
Lample et al. (2016)	-	-	81.7
Akbik et al. (2019)	-	-	90.4
Straková et al. (2019)	-	-	92.7
Our model	94.5	92.8	93.7

Table 3: State of the art comparison on CONLL 2002, CONLL 2003, ONTONOTES corpora for flat NER.

WHAT?

- Biaffine Classifier
- 操作：将 Biaffine Classifier 替换成 CRF，性能下降 0.8%；
- 解释：巨大的性能差异显示了添加biaffine模型的好处，并证实了我们的假设，即依赖项解析框架是我们系统高精度的重要因素。

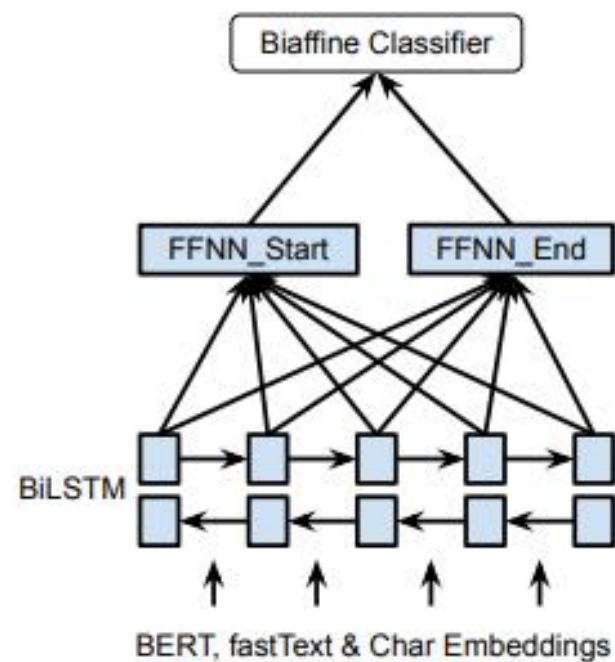


Figure 1: The network architectures of our system.

WHAT?

- Contextual Embedding
- 操作：去掉 Bert emb，性能 下降 2.4%；
- 解释：BERT嵌入是以下之一 精度的最重要因素

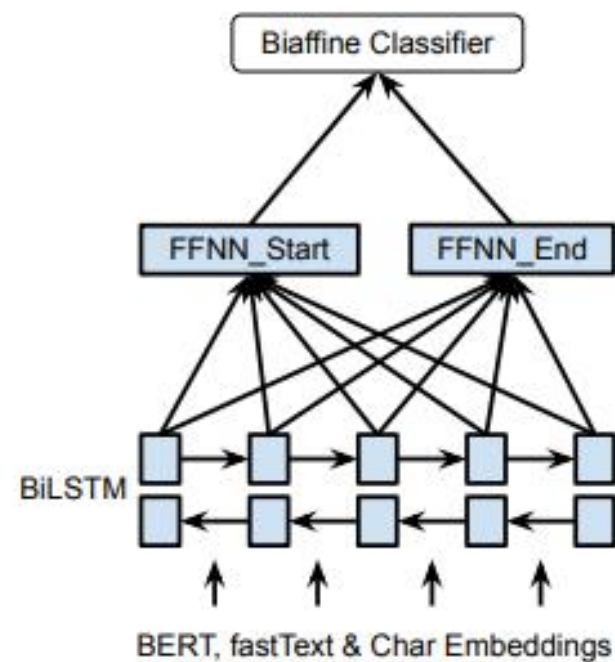


Figure 1: The network architectures of our system.

WHAT?

- Context Independent Embeddings
- 操作：去掉 上下文依赖的 fastText，性能下降 0.4%
- 解释：即使启用了BERT嵌入，与上下文无关的嵌入仍然可以对系统做出相当明显的改进。

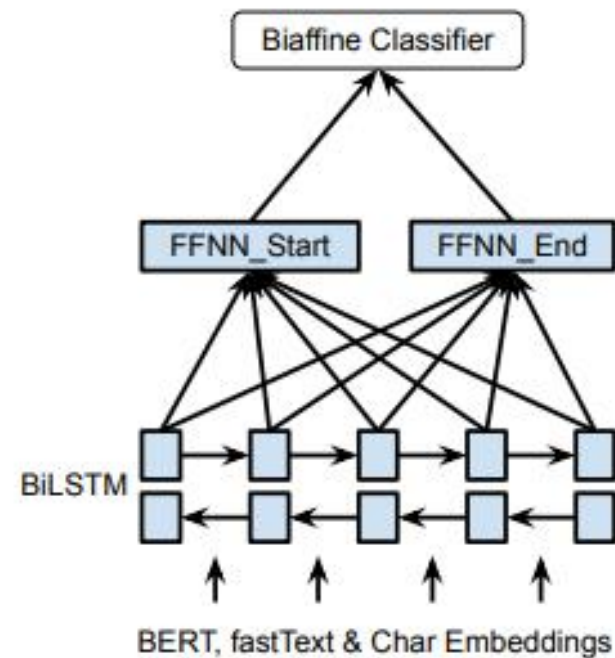


Figure 1: The network architectures of our system.

WHAT?

- Character Embeddings
- 操作：删除 character embeddings，性能下降 0.1%
- 解释：一种解释是英语不是一种形态丰富的语言，因此无益于主要来自角色级别的信息和BERT嵌入本身基于单词已经捕获了一些字符级信息。

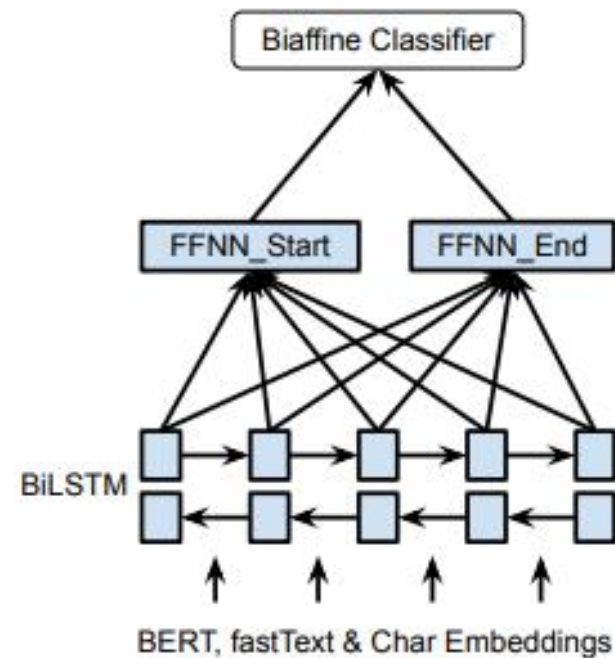


Figure 1: The network architectures of our system.