

# Span-Level Model for Relation Extraction

**Kalpita Dixit**

Amazon AWS AI, USA  
kddixit@amazon.com

**Yaser Al-Onaizan**

Amazon AWS AI, USA  
onaizan@amazon.com

ACL2019

之前讲过，信息抽取包括实体抽取与关系抽取。

有的人是pipeline的形式，有的人是joint的形式。

然后HBP、CopyMTL这些尽管解决了relation extraction中的实体重叠问题，但是并没有解决NER的实体重叠问题。

怎么说，比如 曹孟德爷爷，曹孟德是一个实体，曹孟德爷爷也是一个实体，他们产生了重叠，传统的token-level的标注无法解决这个问题。。

所以这篇paper就提出了span-level的做法。

相当于对于每一个span（token 序列）单独去做标注，所以长度L，就有 $L(L+1)/2$  个span。

这当然带了很大的复杂度啦，所以实验的时候也限制了span的最大长度，好像是10。

模型也很简单粗暴，其本质就是给每个span一个independent的表示。

首先每个词语都有word embedding，由下面这三个表示

1. Fixed Contextual Word Embeddings
2. Fixed Word Embeddings
3. Trained from scratch Character Embeddings

其中2和3都是先预训练好的，1采用Elmo。

然后呢每个span就有多个token，就可以现在内部做一个类似于attention的工作

$$\boxed{\text{token}_t} = \text{MLP}_\alpha(\mathbf{x}_t^*)$$

$$\beta_{i,t} = \frac{\exp(\alpha_t)}{\sum_{k=\text{START}(i)}^{\text{END}(i)} \exp(\alpha_k)}$$

$$\hat{\mathbf{x}}_i = \sum_{k=\text{START}(i)}^{\text{END}(i)} \beta_{i,t} \mathbf{x}_t$$

之后得到span的表示

$$\mathbf{g}_i = [\mathbf{x}_{\text{START}(i)}^*, \mathbf{x}_{\text{END}(i)}^*, \hat{\mathbf{x}}_i, \phi(i)]$$

where  $\phi(i)$  encodes the size of span  $i$  in number of tokens. Each component of  $\mathbf{g}_i$  is a span-specific

然后这个span就可以被送进MLP+classifier去分类进行NER。

然后两个span也可以被同时送进MLP+classifier区进行relation extraction。

最后在ACE 2005 dataset上取得了sota。

但我个人觉得不足的是，既然这个就是为了解决NER实体重叠的问题，是不是应该实验的时候看看是否真的做到了呢？

比如找几个case分析一下，有这种情况的，然后被model识别出来了。

毕竟其余模型是没有办法做到这一点的。