

Code-Switched Named Entity Recognition with Embedding Attention

1. 论文介绍
2. 论文内容和创新点

1. 论文介绍

发表在 ACL2018 的一篇 paper, 主要领域为 code-Switched NER, 看了下论文中的介绍发现是双语种的命名实体识别

2. 论文内容和创新点

2.1 论文内容

本篇论文在全文并没有给出自己的模型图, 我这边先从输入到各个模型结构最后到输出介绍一下他的模型结构

2.1.1 输入

输入包括三个方面分别为 word level, character level 和 capitalization 但是并不是在一层就联系到一起,

character level: 首先通过 bilstm 来提取 character level 的 embedding 如下:

$$\begin{aligned}\vec{\mathbf{h}}_t^c &= \overrightarrow{\text{LSTM}}_t(\mathbf{c}^1, \dots, \mathbf{c}^t) \\ \overleftarrow{\mathbf{h}}_t^c &= \overleftarrow{\text{LSTM}}_t(\mathbf{c}^t, \dots, \mathbf{c}^T) \\ \mathbf{w}_{char} &= \max(\{[\vec{\mathbf{h}}_t^c, \overleftarrow{\mathbf{h}}_t^c]\}_{t=1, \dots, T})\end{aligned}$$

word level: 这里双语种的 word-level embedding 分别包含各自语种中的 word embedding (其实开始阶段分别用各自语种的预料通过 fasttext 进行预训练出各自的 embedding, 在各自为出现过则进行初始化), 可以获得 word+char embedding 为:

$$\begin{aligned}\alpha_i &= \text{softmax}(U \tanh(V [\mathbf{w}_{L_1}, \mathbf{w}_{L_2}, \mathbf{w}_{char}])), \\ \mathbf{w}_{word+char} &= [\alpha_1 \mathbf{w}_{L_1}, \alpha_2 \mathbf{w}_{L_2}, \alpha_3 \mathbf{w}_{char}]\end{aligned}$$

capitalization level: 咋在考虑到大小写的影响在输入阶段还 concat 大小写

的 embedding, 具体为 (没有大写, 首字母大写, 全部大写), 可以得到最终的输入为:

$$\mathbf{w} = [\mathbf{w}_{word+char}, \mathbf{w}_{cap}]$$

2.1.2 模型

模型主要用了堆叠残差 bilstm, 并在每一次时刻的输入为

$$\mathbf{x}_t^i = \begin{cases} \mathbf{w}_t & i = 1 \\ [\mathbf{w}_t, \mathbf{h}_t^1, \dots, \mathbf{h}_t^{i-1}] & i > 1 \end{cases}$$

这里考虑了前面隐层所有时刻的输出, 最后还是通过 CRF 来进行解码

2.2 论文创新点

看了下论文主要利用各大 paper 在各个局部比较好的思路, 主要包括 fasttext 进行初始化、embedding embedding 和残差堆叠 bilstm

本文讲解的 pdf 可以在我的 github 上查看: <https://github.com/GrinAndBear/NLP-Paper-Share/tree/master/NER>

看完麻烦大家点个赞或者 github 上 star 一下多谢啦