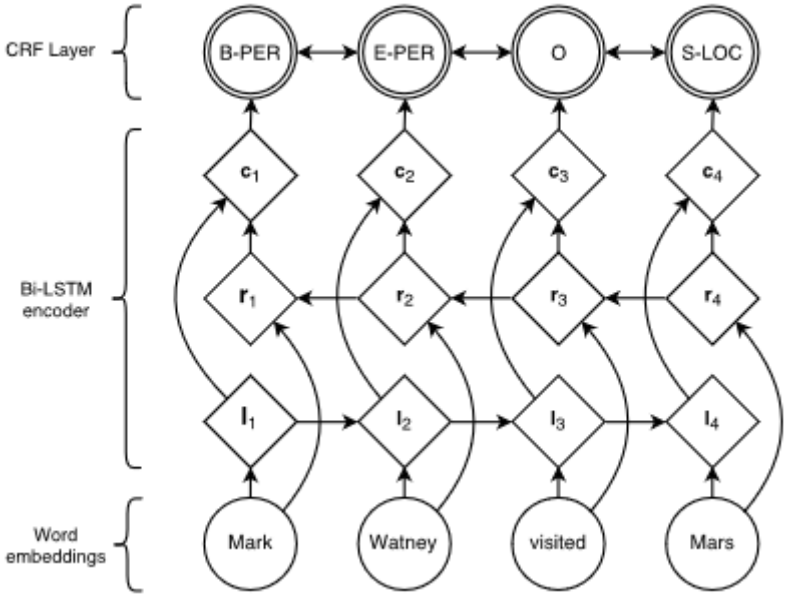
序列标注之命名实体识别

1. 任务描述

命名实体识别属于典型的序列标注的问题。输入一段特定格式的文本，对每个词进行标注。可标注的有意义实体有 LOC, PER, ORG and MISC 分别表示地点(location)，人名(person)，组织(orgnization)和其他混杂的(orgnizations)。如果不含有上述四种实体就用 O 标注。详细说明见 CoNNL-2003 的任务描述[1]。

2. 模型构建

NER 问题网络结构采用经典的采用经典的 Bi-LSTM 和 CRF 结构。如下图所示：



在上图中，Word embedings 输入给 Bi-LSTM，l_i表示单词 i 左侧的 context，r_i表示单词 i 右侧的 context，所以将二者连结在一起就表示一个单词的 context，即上下文信息。LSTM 隐式地建立了序列类别之间的关系，而 CRF 则用于显式地建立目标类别之间的关系。

注：上图参考了参考文献中的论文[7]。

3. 实验环境

此次的实验环境如下表所示：

平台名称	版本
操作系统	Ubuntu 14.04
Python	2.7
TensorFlow	GitHub 源码安装的最新版本,master 分支[5]
词表示学习模型	GloVe 2.0，见参考文献链接[4]

编程实现部分源码参考了何老师给的 tagging_rnn_tf 目录下的源码。

实验的数据集是 CoNLL 2003 的数据集，该数据集的描述见参考文献[1]中的链接，该链接对数据的输入格式有详细的说明。该数据集本身是由路透社提供，要想获取数据集需要去 NIST[2]去申请，由于时间限制，来不及申请，就到 GitHub 的链接[3]去下载了处理过以后的数据，即 eng.testa、eng.testb 和 eng.train。词表示模型的学习用到了 Stanford 的 Wikipedia 2014+Gigaword 5 word embedings，下载 glove.6B.zip，该文件较大(压缩后 800M 左右)，此次实验使用 glove.6B.zip 解压后的 glove.6B.300d.txt(该文件解压后 1 个 G 左右)。glove.6B.zip 具体下载地址见参考文献给出的链接[6]。

整个模型的输入文件的格式：

word \t tag，即每一行只有两个字段：word 和 tag，它们之间以 tab 键分隔。

从链接[3]上下载的数据 eng.testa、eng.testb 和 eng.train 还需要进一步地预处理，可以利用 Linux 的 sed 命令很方便地对文件进行预处理，预处理的命令已经写入 shell 脚本 split.sh 中，直接进入源码目录，在终端执行该 shell 脚本就可以对原始的数据文件进行分割得到 testa.iob、testb.iob 和 train.iob，这三个文件就是输入文件。shell 脚本的内容如下：

sed '/-DOCSTART-/,+1d' ./data/eng.testa | ./toIOB.py | cut -f 1,4 > testa.iob

sed '/-DOCSTART-/,+1d' ./data/eng.testb | ./toIOB.py | cut -f 1,4 > testb.iob

sed '/-DOCSTART-/,+1d' ./data/eng.train | ./toIOB.py | cut -f 1,4 > train.iob

注：如果 shell 脚本 split.sh 和 python 脚本 toIOB.py 不具备执行权限，请给它授予执行权限。同时请正确设置 eng.testa、eng.testb 和 eng.train 文件的路径。

实际运行时，按照如下步骤：

首先，执行以下命令，从路透社数据集和 Stanford 的 word embedings 获取词向量：

$ python build_data.py

然后，执行以下命令训练和测试模型：

$ python main.py

最终的结果会输入在 results 文件夹，评估的结果是 F1 值和准确率，召回率。

注：在具体执行前，请到 config.py 文件中正确设置输入文件的路径以及斯坦福 word embedings 的路径，e.g.：

glove_filename = "data/glove.6B/glove.6B.{}d.txt".format(dim)

dev_filename = "data/testa.iob"

test_filename = "data/testb.iob"

train_filename = "data/train.iob"

4. 参考文献

[1] CoNLL2003. Language-Independent Named Entity Recognition[EB/OL]. http://www.cnts.ua.ac.be/conll2003/ner/,2017,07

[2] NIST. Reuters Corpora (RCV1, RCV2, TRC2)[EB/OL]. http://trec.nist.gov/data/reuters/reuters.html,2017,07

[3] GitHub. RCV1 DataSet[EB/OL]. https://github.com/glample/tagger/tree/master/dataset,2007,07

[4] Stanford. GloVe: Global Vectors for Word Representation[EB/OL]. https://nlp.stanford.edu/projects/glove/,2017,07

[5] GitHub. TensorFlow[EB/OL]. https://github.com/tensorflow/tensorflow,2017,07

[6] Stanford. glove.6B.zip[EB/OL]. https://nlp.stanford.edu/data/,2017,07

[7] Guillame Lample, et al. Neural Architectures for Named Entity Recognition[EB/OL]. https://arxiv.org/pdf/1603.01360.pdf,2017,07