NER 实验分析报告

内容：尝试使用多种不同的模型（包括 HMM，CRF，Bi-LSTM，Bi-LSTM+CRF）来解决命名实体识别问题，进行了实验分析。

数据集：

中文数据集：论文 ACL 2018 Chinese NER using Lattice LSTM 中收集的简历数据

train: 3821 的句子

dev: 463 个句子

test: 477 个句子

tag: 24 个标签

格式：

```
1 O
9 O
8 O
4 O
年 O
至 O
1 O
9 O
9 O
3 O
年 O
在 O
苏 B-ORG
州 M-ORG
市 M-ORG
财 M-ORG
政 M-ORG
局 E-ORG
任 O
科 B-TITLE
员 E-TITLE
。 O
```

模型：HMM、CRF、Bi-LSTM、Bi-LSTM+CRF

HMM：

隐马尔可夫模型描述由一个隐藏的马尔可夫链随机生成不可观测的状态随机序列，再由各个状态生成一个观测而产生观测随机序列的过程。隐马尔可夫模型由初始状态分布，状态转移概率矩阵以及观测概率矩阵所确定。命名实体识别本质上可以看成是一种序列标注问题，在使用 HMM 解决命名实体识别这种序列标注问题的时候，所能观测到的是字组成的序列（观测序列），观测不到的是每个字对应的标注（状态序列）。初始状态分布就是每一个标注的初始化概率，状态转移概率矩阵就是由某一个标注转移到下一个标注的概率。HMM 模型的训练过程对应隐马尔可夫模型的学习问题，实际上就是根据训练数据根据最大似然的方法估计模型的三个要素，即上面提到的初始状态分布、状态转移概率矩阵以及观测概率矩阵，模型训练完毕之后，利用模型进行解码，即对给定观测序列，求它对应的状态序列，这里就是对给定的句子，求句子中的每个字对应的标注，针对这个解码问题，使用的是维特比（viterbi）算法。

CRF：

HMM 模型中存在两个假设，一是输出观察值之间严格独立，二是状态转移过程中当前状态只与前一状态有关。也就是说，在命名实体识别的场景下，HMM 认为观测到的句子中的每个字都是相互独立的，而且当前时刻的标注只与前一时刻的标注相关。但实际上，命名实体识别往往需要更多的特征，比如词性，词的上下文等等，同时当前时刻的标注应该与前一时刻以及后一时刻的标注都相关联。另外，HMM 模型的三个要素是根据极大似然估计的方法从训练数据中得来的，测试集里面有些字是不在训练集当中的，存在 OOV（out-of-vocabulary）问题。显然 HMM 模型在解决命名实体识别的问题上是存在缺陷的。

条件随机场通过引入自定义的特征函数，不仅可以表达观测之间的依赖，还可表示当前观测与前后多个状态之间的复杂

依赖，可以有效克服 HMM 模型面临的问题。

为了建立一个条件随机场，首先要定义一个特征函数集，该函数集内的每个特征函数都以标注序列作为输入，提取的特征作为输出。假设该函数集为：

$$\Phi(x_1, \ldots, x_m, s_1, \ldots, s_m) \in \mathbb{R}^d$$

其中 x=(x_1, ..., x_m)表示观测序列，s = (s_1, ..., s_m)$表示状态序列。然后，条件随机场使用对数线性模型来计算给定观测序列下状态序列的条件概率：

$$p(s|x; w) = \frac{\exp(w \cdot \Phi(x, s))}{\sum_{s'} \exp(w \cdot \Phi(x, s'))}$$

其中 s’ 是是所有可能的状态序列，w 是条件随机场模型的参数，可以把它看成是每个特征函数的权重。CRF 模型的训练其实就是对参数 w 的估计。假设我们有 n 个已经标注好的数据{(x^i, s^i)}_i=1,2,⋯,n,则其对数似然函数的正则化形式如下：

$$L(w) = \sum_{i=1}^{n} \log p(s^i|x^i; w) - \frac{\lambda_2}{2}\|w\|_2^2 - \lambda_1\|w\|_1$$

那么，最优参数 w*就是：

$$w^* = \arg\max_{w \in \mathbb{R}^d} L(w)$$

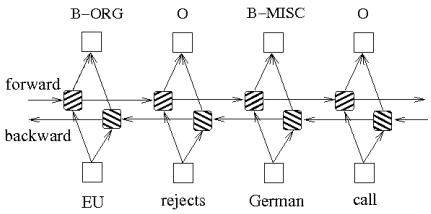
模型训练结束之后，对给定的观测序列 x，它对应的最优状态序列应该是：

$$s^* = \arg\max_s p(s|x; w^*).$$

解码的时候与 HMM 类似，也可以采用维特比算法。

Bi-LSTM:

除了以上两种基于概率图模型的方法，LSTM 也常常被用来解决序列标注问题。和 HMM、CRF 不同的是，LSTM 是依靠神经网络超强的非线性拟合能力，在训练时将样本通过高维空间中的复杂非线性变换，学习到从样本到标注的函数，之后使用这个函数为指定的样本预测每个 token 的标注。下方就是使用双向 LSTM（双向能够更好的捕捉序列之间的依赖关系）进行序列标注的示意图：



Bi-LSTM+CRF:

LSTM 的优点是能够通过双向的设置学习到观测序列（输入的字）之间的依赖，在训练过程中，LSTM 能够根据目标（比如识别实体）自动提取观测序列的特征，但是缺点是无法学习到状态序列（输出的标注）之间的关系。但是，在命名实体识别任务中，标注之间是有一定的关系的，比如 B 类标注（表示某实体的开头）后面不会再接一个 B 类标注，所以 LSTM 在解决 NER 这类序列标注任务时，虽然可以省去很繁杂的特征工程，但是也存在无法学习到标注上下文的缺点。

相反，CRF 的优点就是能对隐含状态建模，学习状态序列的特点，但它的缺点是需要手动提取序列特征。所以一般的做法是，在 LSTM 后面再加一层 CRF，以获得两者的优点。

实验效果：

	HMM	CRF	Bi-LSTM	Bi-LSTM+CRF
准确率	91.49%	95.43%	95.60%	95.64 %
召回率	91.22%	95.43%	95.54%	95.62%
F1	91.30%	95.42%	95.52%	95.61%
标注对的句子/总句子	70.23%	86.37%	76.94%	80.01%

错误分析：

在整体准确率和召回率指标上 Bi-LSTM+CRF 表现最佳，但 CRF、Bi-LSTM、Bi-LSTM+CRF 整体表现差别不大。

CRF、Bi-LSTM、Bi-LSTM+CRF 这三个模型相较于 HMM 提升较大是因为之前提到过的马尔可夫模型本身的局限性，观察单个标签的准确率可以发现 HMM 对 LOC 和 PRO 这两类实体识别表现出很差的效果。（图中标为红色的部分）

	precision	recall	f1-score	support
E-EDU	0.9167	0.9821	0.9483	112
B-RACE	1.0000	0.9286	0.9630	14
E-TITLE	0.9514	0.9637	0.9575	772
B-NAME	0.9800	0.8750	0.9245	112
M-NAME	0.9459	0.8537	0.8974	82
M-CONT	0.9815	1.0000	0.9907	53
M-ORG	0.9002	0.9327	0.9162	4325
B-CONT	0.9655	1.0000	0.9825	28
B-EDU	0.9000	0.9643	0.9310	112
B-LOC	0.3333	0.3333	0.3333	6
B-ORG	0.8422	0.8879	0.8644	553
B-TITLE	0.8811	0.8925	0.8867	772
E-CONT	0.9655	1.0000	0.9825	28
E-ORG	0.8262	0.8680	0.8466	553
E-NAME	0.9000	0.8036	0.8491	112
M-TITLE	0.9038	0.8751	0.8892	1922
E-LOC	0.5000	0.5000	0.5000	6
B-PRO	0.5581	0.7273	0.6316	33
M-LOC	0.5833	0.3333	0.4242	21
O	0.9568	0.9177	0.9369	5190
M-PRO	0.4490	0.6471	0.5301	68
M-EDU	0.9348	0.9609	0.9477	179
E-PRO	0.6512	0.8485	0.7368	33
E-RACE	1.0000	0.9286	0.9630	14
avg/total	0.9149	0.9122	0.9130	15100

通过观察错误样本发现主要是以下两种情况：

一、实体未被识别出来。比如下面这个例子。在训练集中未出现过如皋这个地名，所以 LOC-如皋的发射概率为 0（在代码中赋了一个极小值），一个句子的概率等于每个字的发射概率与字与字之间的转移概率相乘，导致正确序列的概率极小，所以被错误标注。HMM 模型的一大缺点就是如果测试语料没有在训练集中出现过，就会获得一个极小的发射概率，导致整个句子都被错误判断。（HMM 只是学习了训练集中的知识，没有学习新知识的能力。）

错误案例：

样本：[1，9，5，4，年，1，1，月，生，，，江，苏，如，皋，人，，]

真实：[O，O，O，O，O，O，O，O，O，O，B-LOC，M-LOC，M-LOC，M-LOC，E-LOC，O]

预测：[O，O，O，O，O，O，O，O，O，O，O，O，O，O，O，O]

二、LOC 和 ORG 识别混淆。因为 B-LOC 和 B-ORG 比较相似，并且 B-ORG 标签数据较多，而 HMM 模型的概率矩阵是统计得到的，比如，O 到 B-ORG 的转移概率肯定远大于 O 到 B-LOC 的转移概率，所以对于少数几个 B-LOC 的数据很容易被判断成 B-ORG，而

后一时刻的标签又跟前一时刻的标签有关，所以后面时刻的标签也都被影响导致错误判断。

错误案例：

样本：[1，9，6，8，年，生，于，辽，宁，省，鞍，山，市，。]

真实：[O，O，O，O，O，O，O，B-LOC，M-LOC，M-LOC，M-LOC，M-LOC，E-LOC，O]

预测：[O，O，O，O，O，O，O，B-ORG，M-ORG，M-ORG，M-ORG，M-ORG，M-ORG，M-ORG]

对于 PRO 原因同上。

这两种情况最根本的原因在于 ORG、LOC 和 PRO 都是比较长的词语，HMM 不能很好的获得这些长词语信息，越长的实体类型表现越差，对于比较短的实体类型例如 RACE、CONT、NAME 就表现很好。

CRF 相较于 HMM 的提升在于它定义的特征模板。HMM 只考虑当前状态与当前观测的关系，以及当前状态与上一状态的关系，而 CRF 可以获得更丰富的上下文之间的信息。对中文命名实体识别来说，下面展示的是实验中 CRF 使用的特征模板。

```
# 使用的特征：
# 前一个词，当前词，后一个词，
# 前一个词+当前词，  当前词+后一个词
features = {
    'w': word,
    'w-1': prev_word,
    'w+1': next_word,
    'w-1:w': prev_word+word,
    'w:w+1': word+next_word,
    'bias': 1
}
```

后面对特征模板进行改进，加入了特征函数：是否是数字、是否是字母等一类系列特征，得到了更好的效果，准确和召回从 95.43 提升到了 95.69。分析原因主要是因为数据集中大量的英文 ORG 被判断准确的可能性提高了很多，ORG 后面一般会跟着 TITLE，所以 TITLE 指标也提升了很多。

	precision	recall	f1-score	support
E-EDU	0.9910	0.9821	0.9865	112
B-RACE	1.0000	1.0000	1.0000	14
E-TITLE	0.9857	0.9819	0.9838	772
B-NAME	1.0000	0.9821	0.9910	112
M-NAME	1.0000	0.9756	0.9877	82
M-CONT	1.0000	1.0000	1.0000	53
M-ORG	0.9523	0.9563	0.9543	4325
B-CONT	1.0000	1.0000	1.0000	28
B-EDU	0.9820	0.9732	0.9776	112
B-LOC	1.0000	0.8333	0.9091	6
B-ORG	0.9636	0.9566	0.9601	553
B-TITLE	0.9376	0.9339	0.9358	772
E-CONT	1.0000	1.0000	1.0000	28
E-ORG	0.9199	0.9132	0.9165	553
E-NAME	1.0000	0.9821	0.9910	112
M-TITLE	0.9248	0.9022	0.9134	1922
E-LOC	1.0000	0.8333	0.9091	6
B-PRO	0.9091	0.9091	0.9091	33
M-LOC	1.0000	0.8095	0.8947	21
O	0.9630	0.9732	0.9681	5190
M-PRO	0.8354	0.9706	0.8980	68
M-EDU	0.9824	0.9330	0.9570	179
E-PRO	0.9091	0.9091	0.9091	33
E-RACE	1.0000	1.0000	1.0000	14
avg/total	0.9543	0.9543	0.9542	15100

CRF 相较于 HMM 各个标签都有了很大的提升，尤其是在 HMM 表现不佳的 LOC 和 ORG 上提升巨大。CRF 通过特征模板定义的很多特征函数对长词语可以进行更好的识别。

Bi-LSTM 在汉字级别进行训练，每一个时刻的输入是汉字向量，训练共用时 630 秒。Bi-LSTM 训练中 30 轮 epoch 的 loss 一直在下降，降到了 0.01，20 轮左右的时候验证集的 loss 就降到了 0.1232，之后在这个范围上下波动，说明过拟合。

	precision	recall	f1-score	support
B-ORG	0.9724	0.9566	0.9644	553
E-TITLE	0.9948	0.9819	0.9883	772
M-LOC	1.0000	0.8095	0.8947	21
E-LOC	1.0000	0.8333	0.9091	6
E-PRO	0.8286	0.8788	0.8529	33
B-TITLE	0.9330	0.9197	0.9263	772
M-PRO	0.7412	0.9265	0.8235	68
B-LOC	0.8333	0.8333	0.8333	6
M-ORG	0.9619	0.9586	0.9603	4325
E-NAME	1.0000	0.9643	0.9818	112
E-EDU	0.9818	0.9643	0.9730	112
O	0.9488	0.9925	0.9701	5190
B-CONT	1.0000	1.0000	1.0000	28
E-CONT	1.0000	1.0000	1.0000	28
B-EDU	0.9730	0.9643	0.9686	112
B-NAME	0.9717	0.9196	0.9450	112
E-ORG	0.9411	0.8951	0.9175	553
E-RACE	1.0000	1.0000	1.0000	14
B-RACE	1.0000	0.9286	0.9630	14
M-EDU	0.9540	0.9274	0.9405	179
M-NAME	0.9630	0.9512	0.9571	82
B-PRO	0.9062	0.8788	0.8923	33
M-TITLE	0.9568	0.8757	0.9144	1922
M-CONT	1.0000	1.0000	1.0000	53
avg/total	0.9560	0.9554	0.9552	15100

Bi-LSTM 平均准确率和召回都比 CRF 略好，具体分析标签类别，观察得到 Bi-LSTM 指标的提升来自于 ORG 和 TITLE 实体类别，在其他实体类别上表现不如 CRF。因为 ORG 占比较大所以整体得到提升。别的标签类别被误判为 ORG 和 TITLE 的情况减少了很多，但 TITLE 和 ORG 被判断成别的标签的情况增加。

观察错误样本发现，因为 Bi-LSTM 并不考虑标签之间的关系，所以会出现以下不符合逻辑的情况：

错误案例：

[全，国，优，秀，党，务，工，作，者，；]

[B-TITLE，M-TITLE，M-TITLE，M-TITLE，M-TITLE，M-TITLE，M-TITLE，M-TITLE，E-TITLE，O]

[B-TITLE，B-TITLE，M-TITLE，M-TITLE，M-TITLE，M-TITLE，M-TITLE，O，E-TITLE，O]

[2，0，1，0，年，6，月，1，日，起，至，今，任，楚，天，高，速，董，事，。]

[0，0，0，0，0，0，0，0，0，0，0，0，0，B-ORG，M-ORG，M-ORG，E-ORG，B-TITLE，E-TITLE，O]

[0，0，0，0，0，0，0，0，0，0，0，0，0，B-ORG，M-ORG，M-ORG，M-TITLE，M-TITLE，E-TITLE，O]

Bi-LSTM 整合整个句子上下文信息能够更好的区分实体类别，判断出句子这一部分属于 TITLE，但由于缺少 BME 规则的约束，每个字都会选取最有可能的标签，由于 ORG 和 TITLE 都比较长，所以 M-ORG 和 M-TITLE 数量较多，大多数 M 标签被判断正确，但也有很多别的标签被判断成 M 标签。Bi-LSTM 学到了整个序列信息，但由于缺少 BME 规则的约束，所以会出现上述不合逻辑的情况。

ORG 和 TITLE 在整体样本中占比较大，所以在神经网络训练过程中中会被更 “重视”，其他实体类别准确率下降的原因就是因为被 ORG 和 TITLE 抢夺了注意力。ORG 和 TITLE 准确率提升，其他标签误判为 ORG 和 TITLE 的情况减少。

二者发生了转化。例如下面这个例子：

错误案例：

[全，国，优，秀，经，营，管，理，工，作，者，；]

[B-TITLE，M-TITLE，M-TITLE，M-TITLE，M-TITLE，M-TITLE，M-TITLE，M-TITLE，M-TITLE，M-TITLE，E-TITLE，O]

[O，O，O，O，M-TITLE，M-TITLE，M-TITLE，M-TITLE，O，O，O，O]

分析原因是由于 Bi-LSTM 缺乏标签之间的规则约束，不能很好的在判断对的标签上面进行扩展使别的标签被判断对的概率增加。所以要在 Bi-LSTM 加上 CRF 来学习状态序列的特点。

Bi-LSTM+CRF:

	precision	recall	f1-score	support
B-ORG	0.9658	0.9693	0.9675	553
E-TITLE	0.9883	0.9845	0.9864	772
M-LOC	1.0000	0.8095	0.8947	21
E-LOC	1.0000	0.8333	0.9091	6
E-PRO	0.8857	0.9394	0.9118	33
B-TITLE	0.9446	0.9275	0.9359	772
M-PRO	0.7778	0.9265	0.8456	68
B-LOC	1.0000	0.8333	0.9091	6
M-ORG	0.9465	0.9660	0.9562	4325
E-NAME	1.0000	0.9732	0.9864	112
E-EDU	0.9732	0.9732	0.9732	112
O	0.9656	0.9746	0.9701	5190
B-CONT	1.0000	1.0000	1.0000	28
E-CONT	1.0000	1.0000	1.0000	28
B-EDU	0.9649	0.9821	0.9735	112
B-NAME	1.0000	0.9554	0.9772	112
E-ORG	0.9298	0.9096	0.9196	553
E-RACE	1.0000	1.0000	1.0000	14
B-RACE	1.0000	0.9286	0.9630	14
M-EDU	0.9766	0.9330	0.9543	179
M-NAME	0.9878	0.9878	0.9878	82
B-PRO	0.8485	0.8485	0.8485	33
M-TITLE	0.9461	0.8949	0.9198	1922
M-CONT	1.0000	1.0000	1.0000	53
avg/total	0.9564	0.9562	0.9561	15100

单个标签的指标与 CRF 大致分布一致。

在这个数据集上 Bi-LSTM+CRF 相比于 Bi-LSTM 没有明显提升，原因可能是在加上规则约束后，又带来了其他的误判。

Bi-LSTM ：[1, 9, 6, 3, 年, 出, 生, ，, 工, 科, 学, 士, ，]

真实：[O, O, O, O, O, O, O, O, B-PRO, E-PRO, B-EDU, E-EDU, O]

预测：[O, O, O, O, O, O, O, O, B-PRO, M-EDU, E-PRO, E-EDU, O]

Bi-LSTM+CRF ：[1, 9, 6, 3, 年, 出, 生, ，, 工, 科, 学, 士, ，]

真实：[O, O, O, O, O, O, O, O, B-PRO, E-PRO, B-EDU, E-EDU, O]

预测：[O, O, O, O, O, O, O, O, B-PRO, M-EDU, M-EDU, E-EDU, O]

BI-LSTM+CRF 比起 Bi-LSTM 效果并没有好很多，一种可能的解释是：

数据集太小，不足够让模型学习到转移矩阵，然后尝试在大的英文数据集上进行了实验。

英文数据集：CoNLL2002 数据集

train:14987 的句子

dev: 3466 个句子

test: 3684 个句子

tag: 9 个标签

实验效果：

	HMM	CRF	Bi-LSTM	Bi-LSTM+CRF
准确率	92.90%	93.01%	90.53%	93.34 %
召回率	93.12%	93.21%	93.90%	93.58%
F1	92.52%	92.65%	89.59%	93.36%
标注对的句子/总句子	57.73%	58.82%	53.20%	57.81%

可见在大的英文数据集上 BI-LSTM+CRF 相比 Bi-LSTM 效果明显好了很多。但是因为用的同一个模型，对于英文标注数据集，只有 word-level 的词向量，没有字符级别的处理，直接对单词向量进行了 Bi-LSTM 编码，导致整个单词字母之间的信息没有被充分考

虑，所以准确率一下子降到了 90%左右。加上 CRF 后上升到了 93%，如果加上字符级别的 bi-LSTM 或者预训练词向量的话可以达到更高。对英文数据集的 CRF 模型使用改进的特征模版准确率可以从 93%达到 95%。

指标标注对的句子/总句子，无论是在中文数据集还是英文数据集，都是 CRF 模型表现最好。分析原因是因为 CRF 把整个句子当成一个整体来考虑。不再是 N 个 K 分类问题，而是一个 K^N 分类问题。至于 Bi-LSTM+CRF 没有 CRF 效果好，可能是因为 Bi-LSTM 提取到的特征作为发射概率矩阵传输到 CRF 层时已经出现了误差，影响了整个句子某个地方出错，从而使指标下降。

TODO：

对于中文 NER，目前前沿的模型是 Lattice LSTM。学习更加复杂的模型，参考论文 Chinese NER using Lattice LSTM。