整理了一下刚才NER的一些讨论和资源，方便大家阅读～

分享一些我在医疗方面做NER的经验教训：

- 提升NER性能的方式往往不是直接去堆一个BERT+CRF，推断速度更是堪忧；
- 在NER任务上，也不要试图对BERT进行蒸馏，很可能吃力不讨好～
- NER任务是一个重底层的任务，上层模型再深其实提升也是有限的，所以不要搭建很深的网络、不要用各种attention了；
- NER任务不同的解码方式（CRF/指针网络/Biaffine）之间的差异其实也是有限的，所以不要拘泥于解码方式；
- 通过MRC-QA的方式进行NER任务，效果也许会提升，但任务复杂度上来了，你需要对同一文本进行多次编码（同一文本会构造多个question）；
- 设计NER任务时，尽量不要引入嵌套实体任务，不好做，大概率是长尾的。
- 不要直接将Transformer用来做NER，这是不适合的，详细可参考TENER。

那么，什么才是NER的正确打开方式：直接点：1层lstm+crf！

1. 如果这么直接，效果达不到业务目标，那就引入更丰富的词向量进行拼接：比如bigram，dictionary-embedding，elmo等等，还有业务相关的特征；如果有领域相关的词向量，那是最好不过了～总之，底层的特征越丰富、差异化越大越好。如果是打比赛，也可以**选取不同的预训练语言模型进行底层拼接**（做特征集成，不是finetune！），比如BERT和XILNet进行拼接，语言模型的差异越大，拼接的效果越好；如果采取特征集成的方式进行BERT向量化，对于一些定时任务仍然可以落地应用，我们可以离线计算这些向量的～
2. 如果你的NER任务，有的实体span比较长（比如医疗中的手术名称是很长的），这时候只用CRF-loss也许不够，可以尝试引入pointer net，进行多任务学习（要注意调参了，pointer net收敛是较慢的～）
3. 如果你面临的是一个低资源NER任务，这时候就采取比如文本增强、半监督学习。有很多paper就在研究这方面的工作，这里不再赘述了，有机会再讲。不过，对于NER任务，文本增强或者弱监督方法提升还是有限的。
4. 不过，如果一个NER任务如果标注数据还是少，最应该想到的就是直接去补标数据了，不过这时候可以采取一些省成本的标注方式，比如主动学习、辅助标注等。更极端的情况下，直接抛弃模型，对于医疗领域，**维护一个好的词典吧**～对于通用领域，除了词典，也可以通过多种分词工具和句法短语工具直接融合进行NER吧～
5. 其实，还有一种立竿见影的方法，我叫它「**词汇增强**」，还是在底层引入词汇信息。为了防止分词误差积累，一般中文NER都是基于字的，但词汇信息的边界对于NER是很有用的～「**词汇增强**」很有效，特别是ACL2020的两篇paper：Simple-Lexicon：Simplify the Usage of Lexicon in Chinese NER和 FLAT: Chinese NER Using Flat-Lattice Transformer再此证明这类方法的有效性，这种方法很轻量化，也可以比肩甚至超过BERT的效果了。
6. 有时候我们面临的可能不是一个实体抽取任务，而是一个段落抽取（通常用来切割事件），这时候的做法比NER难度要高，我们可以采取一些指针网络的做法，也可以借鉴图像领域的RCNN MASK的做法～
7. 有的NER任务要确保高召回和高准确，我们可以采取pipeline的方式，第一步抽取比较粗粒度的实体，通过模型+规则+词典保证高召回；第二步进行细粒度的实体分类，通过模型+规则保证高准确；
8. 。。。想起来，再补充吧

总之，当我们面对一个NER任务的时候，就是1层lstm+CRF，再底层embedding引入更丰富的特征是最佳方式，特别是在一些领域化场景，挖掘更丰富的业务特征，会比直接堆一个BERT好、而且又轻量。

（纯属个人观点，不喜勿喷～）

一个交流

@一一: lstm+crf做实体提取时，保证精度的情况下，在提升模型速度上有没有什么好的办法或者建议?

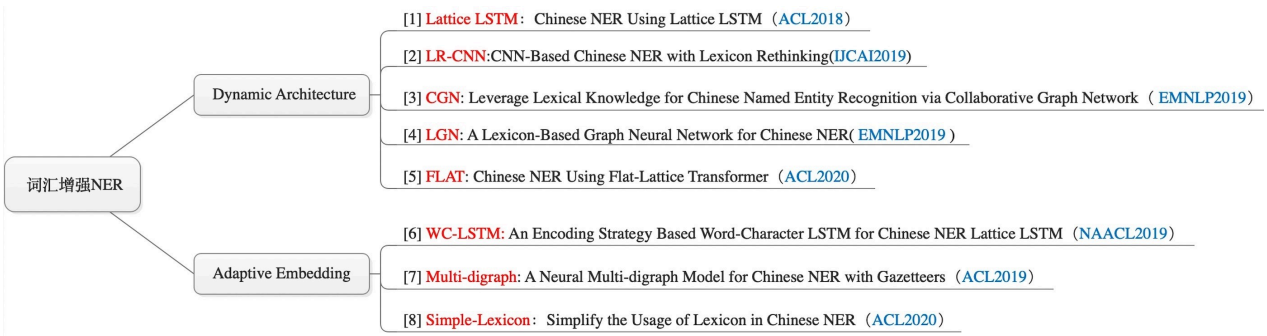
我个人觉得，第一种是：把lstm换成逼的cnn或transformer也许更快一些，不过效果好不好要具体分析，lstm对于NER任务的方向性特征和局部特征捕捉会好于别的编码模型；第二种是：crf的解码速度慢，引入label attention机制把crf拿掉，比如这篇论文：Hierarchically-Refined Label Attention Network for Sequence Labeling；当然可以用指针网络替换crf，不过指针网络收敛慢一些。总之，各有利弊吧。此外，对lstm+crf做量化剪枝也是一个需要权衡的工作，有可能费力不讨好～

在实际业务上，我们一般不去在模型层面去提高推断速度的，主要还是采取显存调度，能够更充分的利用显存吧，还是一些工程化的优化手段，比如一些分桶操作去更充分占用显存。

当前中文NER任务的主要方法汇总

- **各主流方法在主要中文NER数据集上的表现情况** 具体说明

	lexicon	Ontonotes	MSRA	Resume	Weibo
biLSTM	----	71.81	91.87	94.41	56.75
Lattice LSTM	词表1	73.88	93.18	94.46	58.79
WC-LSTM	词表1	74.43	93.36	94.96	49.86
LR-CNN	词表1	74.45	93.71	95.11	59.92
CGN	词表2	74.79	93.47	94.12	63.09
LGN	词表1	74.85	93.63	95.41	60.15
Simple-Lexicon	词表1	75.54	93.50	**95.59**	61.24
FLAT	词表1	**76.45**	94.12	95.45	60.32
FLAT	词表2	75.70	**94.35**	94.93	**63.42**
BERT	----	80.14	94.95	95.53	68.20
BERT+FLAT	词表1	**81.82**	**96.09**	**95.86**	**68.55**



自己做的一个基于深度学习的信息抽取项目

DeepIE：https://github.com/loujie0822/DeepIE

之前写过的一些关于NER&信息抽取的文章和论文汇总

- **知乎专栏文章**：nlp中的实体关系抽取方法总结
- **知乎专栏文章**：如何有效提升中文NER性能？词汇增强方法总结
- **知乎专栏文章**：如何解决Transformer在NER任务中效果不佳的问题？
- ACL2020信息抽取相关论文汇总
- IJCAI2020信息抽取相关论文汇总
- 2019各顶会中的关系抽取论文汇总
- 事件抽取论文汇总
- 历年来NER论文汇总

未来NER的一些展望

- 低资源NER
- 跨语言NER
- 多模态NER