上一次讨论：2019.09.03

根据我看的那篇综述文章，以及我谈的对自监督的理解，老师总结：一种常见的自监督：利用大量unlabeled的数据，auto-generate一个标签，使模型学出一个对数据的encoding。然后fine-tune一下就可以迁移到其他任务上。这是现在流行的对self-supervised的理解。

- **BERT再看一下。**
- **Mark**：关注ICML2019 一个关于自监督的Workshop

对于现有的内容先进行一个“自建模”，然后再在一个specific task上做一个知识迁移。跟snowball中的自监督确实不是一回事。

- Multi-Headed Attention的笔记 整理一下

“独立子空间”的思想 独立特征集

目前深度学习都不讲理论模型，而是讲评测模型，对于某些评测指标（比如BLEU score），它效果上去了，就可以发文章。

snowball作为一个十几年前的研究，如果放到现在来看，文章是肯定会被拒的，因为算不上一个完整的科学研究。它需要打一个“补丁”——你的方法需要有一个理论上的、数学模型上的一个理论支撑（最优化理论）。比如Multi-Headed Attention，其子空间内部的耦合度达到了最高，子空间和子空间之间的耦合度达到最低。（提到概念：“**谱聚类**”。在**图分析**里用的比较多。社区内部的耦合度比较高，社区之间的耦合度比较低。通过一个“拉普拉斯矩阵”，求它的次小特征值，就可以求出图上的最小割，不断地做二划分，最后就可以分为多个社区。）如果你的机器学习目标最后会被证明到是这个理论最优上去，那么这篇文章是完整的。

能不能用一个更加高端的理论模型来监督到它内部的耦合度在不断地增加，同时还能去除掉一些坏样本。平时进来的坏样本，在后面经过投票机制可能会被投出去。并不是一个只进不出的过程。“狄利克雷过程”。拟合的峰值在样本空间里滑动。我们估计的极值和样本空间实际的极值是有偏差的，在不断靠近。

目前的研究大家都不这么玩。相当于钻了一个空子。BLEU score并不能完全等价于similarity，并不代表实际的分布，有的文章只是拟合了BLEU score。这个相当于是一个小想法。按着这个思路来读文章的话，会发现有一些文章就是在拟合BLEU score。老师建议的研究方案是脑袋里有一些最优化的理论基础，看看能不能提出一些最优化的理论方案。能不能用一种数学模型的方法监测出来它是对的。比如说做relation embedding，然后你可以证明它在训练的过程中relation的方差是逐渐变小的，这样才具有说服力。BLEU Score本身有它的弊端。在Benchmark比较低的时候BLEU score没问题。但高了之后就有问题了。这时候建议思考另外一个问题。（最优化理论的坑需要填一下）空间低秩化以后，需要的参数变小，但是表达能力并没有变差（此处想到了Word Embedding）。很可能是跟你语料的内容是有关系的。8个头可能是对于它训练集中数据的空间里是最优的。相当于最优子空间的个数就是8个。

现在说的自监督是直接在某一个特定的任务上扩充他的训练集。

EM算法回去再看一看。

话题抽取，文本话题分类

EM算法的简化版——K-Means

在自身这个样本集上定义出一个好的损失函数，让他去拟合。比如说，定义出一个超规则，相似度就按BLEU算。只要能定义出这个损失，那就是自监督。“在数据集上天然存在的一种监督信号，被你捕捉到了，然后设计一个合理的方案，以一个不错的效果解决了你的问题，那就是自监督学习。”要在当前的数据集里边，通过你的理解，定义出在这个天然的数据集里边能够直接被观测到的一个损失函数，作为一个标准。利用这个标准来监督你的学习过程，就叫做自监督“。

Relation Extraction。一开始的5个样本叫seed，不算训练数据。就像K-Means里面的启动类。按照这几个种子先似然出一个概率分布，这个概率分布是最大化这几个种子，在样本集里用这5个样本把其他符合规则的样本也找出来。我们把样本及里面所有跟seed相关的样本都挖掘出来了，同时对于符合这些样本的model也学出来了，这就是自监督学习。因此，一开始的几个种子并不是核心问题。核心问题是损失函数定义的是不是好。在relation Extraction这个问题上有可能定义出自己的一个损失函数，它自己不断地迭代，迭代之后发现最后出来的结果质量挺好的，比较好地收敛到了我数据样本的内容上，充分的利用上了，因此就把关系挖掘的很好。同时，如果还能在数学上给出一个最优化的解释，表明它不是局部最优，而是全局最优，那就厉害了。需要解决两个问题，一是如何避免陷入局部最优，二是如何避免其“跑偏”，即偏离了原来seed的范围内（与种子无关了）。这时候能不能有一个机制来控制它。如果真的跑丢了，种子没了，那就变成纯的无监督了，一次性挖掘所有关系，那就太难了。它的理论极限就是纯无监督的，一次性挖掘所有关系，但是感觉这个有点不切实际。比较实际的就是人机互助的，

snowball最终结果就是要那个模型，就是它的上下文向量，就是那三个（left，middle，right）。所有的样本最后就是为了学一个语言模型。假设这个vector的效果非常好，极限情况就是任意拿过来一个句子，代入这个vector，都可以把关系抽取出来。对于这篇文章所指的“关系”就是两个名词——location和organization，关系相当于是“locate at”，它相当于是抽取出来的是两个实体和一个特定的relation。这种方法只能针对某一种特定的关系能够work。

后面又对比了一下snowball与COMET。snowball相当于是一个“判定模型”，从现有的数据集中筛选；COMET相当于是一个“生成模型”。只要是能够隶属于一个特定的损失函数，就可以算是一个（）。snowball的思想用于生成模型也是可以的。因为生成模型里有一个pre-trained的set，然后后面还有另一个set，用于fine-tune。第一步是得到pre-trained模型，第二步是把模型放到对应的task上训练。不管是COMET还是snowball中用的vector，本质上都是一个模型，对于这个模型利用snowball的过程，不断迭代，最后也是可以得到生成模型的。

最终的想法：在COMET外面再包一层snowball。第一部分是一个pre-trained好的语言模型，第二步在对其localize的过程中，采取snowball的机制，不断迭代、拟合，最终得到想要的模型（这里指的是适用于Relation Extraction的模型？）

P.S. 听完这段之后，我想到了两个问题：

- 一是如何选取这个语言模型。这个可以先搁置，后面再解决，不过大致方向是不是看ELMo、GPT、Bert这些最近流行的，用于NLP里面一些任务的pre-trained模型（或者它们的变种）？就是注意力是不是应该放在这些上？到时候看看一些采取上述预训练模型，或经过轻微改动之后，解决具体任务的，看看它们的具体操作方法以及实现细节?
- 二是这个第二步，也就是将预训练模型localize的过程，我们要仿照snowball的模式，定义输入输出、损失函数，以及一个监督其是否收敛/发散的规则，还有何时才算收敛的判断条件。感觉这个是应该先考虑的问题？然而现在只是这个思路大概能理解了，具体怎么设计并没有任何头绪....

后面我提了一下这次讨论涉及到的数学/统计学内容比较多，基本都没学过，大一大二我们学院只上了四门最基础的数学课（高等数学、线性代数、概率统计、数值分析），难一点的统计学、随机过程以及机器学习都不太了解，需要补。

开始做的话，可以先复现一下COMET。至于后面具体怎么设计Loss、怎么定义规则来判断收敛等等，这些层面的问题留给我后面自己来完成。因为它不需要太多的理论知识，而是需要在特定数据集上对数据的观察，在过程中自己摸索。对于这个特定的问题，要用哪些小的trick去过滤好的样本。这就涉及一个精细的损失函数的设计。现在许多工作只关注这些小的细节就能发出文章。我们是关注稍微大一些的问题，提一个大的解决方案，但是这些小的细节也是需要的。

理论确实是有帮助，是鉴定一个研究质量高低的重要的标志。但也不是没有它就发不出文章来。就现在来说，因为是远程指导，还是建议能够先做出一些东西来，快出成果，积累一些小文章，哪怕是低一点的也可以，给自己建立信心。理论直觉方面没有问题，可以用当前流行的深度学习方法来试着做一下，如果能做出效果来以后，想最后来作出一篇漂亮的理论文章，后面会有实质性的投入。

一方面，把现有的模型（COMET）去train一下，复现一下。另一方面，提到了谱聚类，提到了EM算法，回去看，学到哪个点不懂可以来问。最近就先往下做，碰到不清楚的东西就学，反正也不着急出成果。理论模型先往后放一放，不要主攻理论模型，不然可能信心会受到打击。

“Graph Theory——图理论，图分析”，是机器学习里边一个重要的理论，感兴趣的话可以学一学，社区，标签传播算法，随机块模型，等等。

本次讨论收获名词：

谱聚类；拉普拉斯矩阵；次小特征值；最小割；文本话题分类；狄利克雷过程