2020.2.19日学习：

强化学习的特点：

基于评估，强化学习利用环境评估当前策略，以此为依据进行优化。

在交互中产生，数据与环境的交互中产生。

序列决策过程，这些决策过程往往是前后关联。

强化学习是建立在马尔科夫过程，是在环境交互中进行。

强化学习中的概念：

智能主体(agent):

按照某种策略，根据当前的状态，选择合适的动作;

状态指的是智能主体对环境的一种解释

　　动作反映了智能主体主观能动的影响，动作带来的收益称为奖励。

　　　智能主体可能知道也可能不知道环境变化的规律。

环境：

　　　系统中智能主体以外的部分;

向智能主体反馈状态和奖励

　　　按照一定的规律发生变化

根据一下特征定位强化学习：

有无可靠的反馈信息；

　　基于评估/基于监督信息

　　序列决策/单步决策

　　基于采样/基于穷举

离散马尔科夫用状态转移矩阵来算；但是这个没法体现交互性，于是定义了奖励也叫反馈，用来反映累加的奖励，其中有衰退系数。

马尔科夫决策过程包含随机变量序列{St},动作集合，状态转移概率，奖励函数，衰退系数；

马尔科夫过程中产生的状态序列成为轨迹，轨迹长度可以是无限也可以是有终止状态的，有终止状态的叫分段(episodic)，否则叫持续（continuing）;分段问题中，一个从初始状态到终止状态的完整轨迹成为一个片段。

马尔科夫的决策学习中动作是受策略驱动，

价值函数：在第ｔ步状态为ｓ时，按照策略pi 行动后在未来所获得反馈值的期望；

动作－价值函数：在第ｔ步，按照策略ｐｉ采取动作ａ后，在未来获得反馈值的期望。

序列决策包括有限状态强化学习和无限状态强化学习，基于评估的包括序列决策；

基于采样的包括无限状态强化学习和有监督学习

策略优化与策略评估

第一部分:策略优化，给定当前策略，价值函数和行动价值函数，可构造新的策略pi,pi满足如下条件：

Pi(s) = argmax(a)

第二部分：策略评估

通过迭代贝尔曼方程进行策略评估

１。动态规划：缺点智能体需要事先知道状态转移概率，无法处理状态集合大小无限的情况

２。蒙特卡罗采样：优点：智能主体不需要知道状态转移概率，容易扩展到无限状态集合的问题中。缺点：状态集合比较大时，一个状态在轨迹可能非常稀疏，不利于估计期望。

在实际问题中，最终反馈需要在终止状态才能知晓，导致周期比较长

３。时序差分：从某个状态开始采样某个动作，利用蒙特卡罗采样的思想，通过采样ａ和ｓ’的期望

蒙特卡罗采样选择不同的起始状态，按照当前策略采样若干轨迹，来更新某个状态的价值总值

第一部分和第二部分的结合：策略优化和策略评估结合

基于Ｑ-learning的时序差分方法，基于价值的方法不直接对策略建模，因此在策略优化采样和更新两步中之max操作上得以间接体现。

在同一次循环中策略评估和策略优化交替进行，由于策略优化要求计算动作－价值函数ｑ，因此Q学习直接利用ｑ函数的贝尔曼方程更新。

深度强化学习:

用神经网络拟合行动价值函数，由于状态空间太庞大，有些状态可能始终无法采样到，因此对这些状态ｑ函数进行估计是很困难。

状态数量无限，不可能用一张表来记录ｑ函数的值。

思路：将ｑ函数参数化，用一个非线性回归函数来拟合ｑ函数，例如深度神经网络，能够用有限的参数刻画无限的状态。

由于回归函数的连续性，没有探索过的状态也可通过周围的状态来估计。

深度学习的两个不稳定因素：

１。相邻的样本来自同一条轨迹，样本之间相关性太强，集中优化相关性强的样本可能导致神经网络在其他的样本上效果下降

２。在损失函数中，ｑ函数的值即用来估计目标值，又用来计算当前值，现在这两处的ｑ函数通过Sita有所关联，可能导致优化时不稳定。

博弈与安全：

这里明天补上

**自然语言的学习**

Transformerx学习:基于encoder和decoder模型

ALBert是Bert的瘦身，也加入很多内容;

Bert的输入有３部分,positional,token,segment;

Pre-train:MLM和NSP两种方式－－－有待理解。

subword-nmt，对与subword的使用需要看这个模型。

Bert的分词，do\_whole\_word mask不做subword mask;

李宏毅老师的自然语言课程学

Elmo模型是把各层的隐状态都要，乘上一个权重;

Bert是Encoder of transformer,　如果是中文的话应该采用字会更和是一点，如果是词的话，中文的情况是无法穷举的;

Bert的３中应用场景：

case1:

怎么使用Bert那节还没怎么看懂，怎么使用Bert，在后面接分类器，这个分类器的参数可以，这个分类器的参数可以用随机初始化，Bert是微调的这个怎么理解呢？他们俩怎么一起训练呢？

Case2:输入２个句子，输出一个class label

case3:slot fill,输出每个词对应的类别，这个时候把每个输出的词向量都输出到分类器中。

用了12层来训练，但是用了４层来测试，这个效果也没有差太远？？？－－不明白

bert的可视化bertviz

Bert 对ＱＡ那些复杂的场景的提升比较好，对那个NER的提升不大，

Bert 比Seq2Seq的速度快，是可以并行计算，

Bert能小样本，当句子编码。

在Bert中分类为什么只用CLS这个标签的输出

Bert增加错字可以增加鲁棒性

不同的任务学习到的Bert不同层上的信息。

GPT模型很庞大，对于one-shot　learning 效果很小。