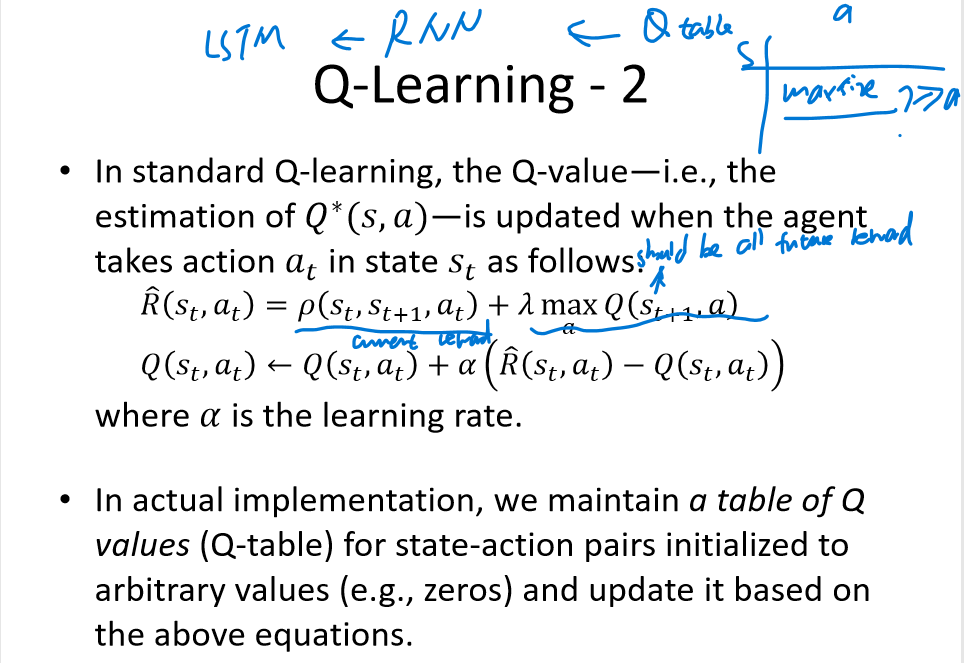
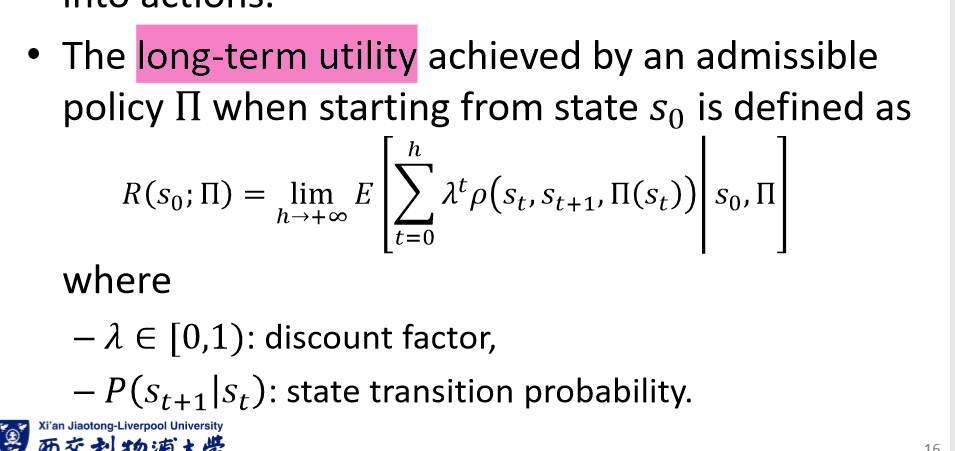
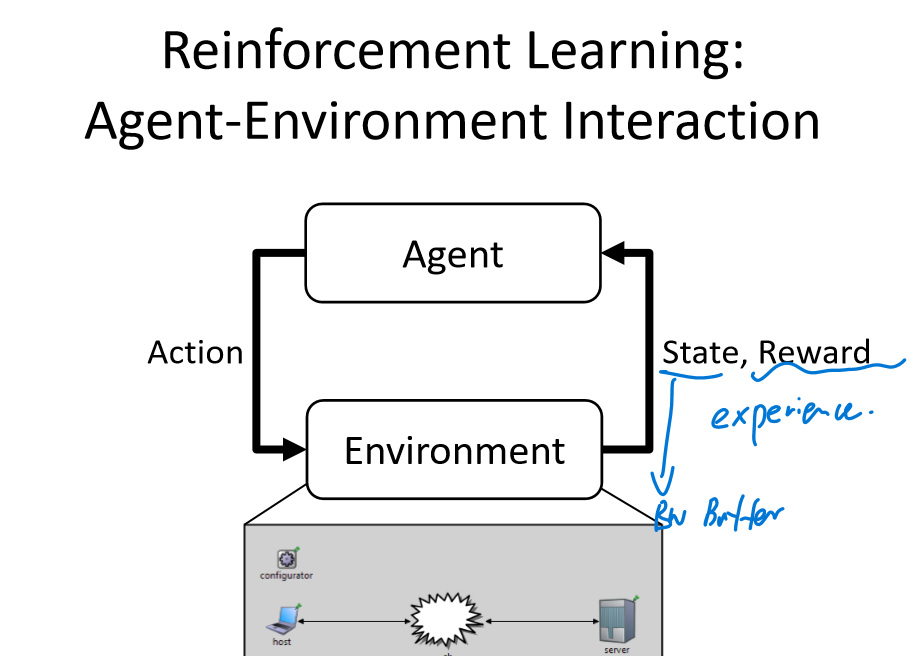
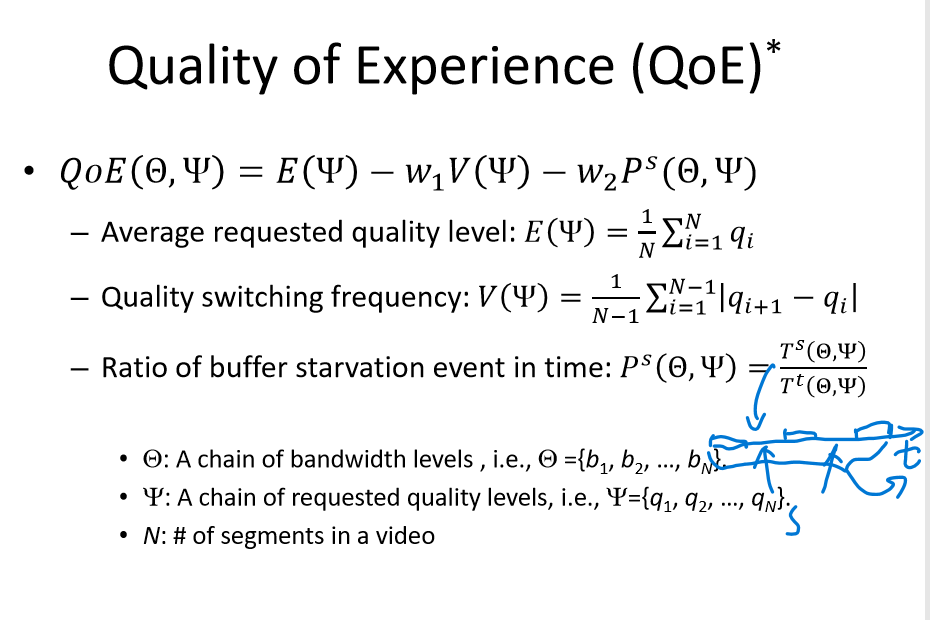
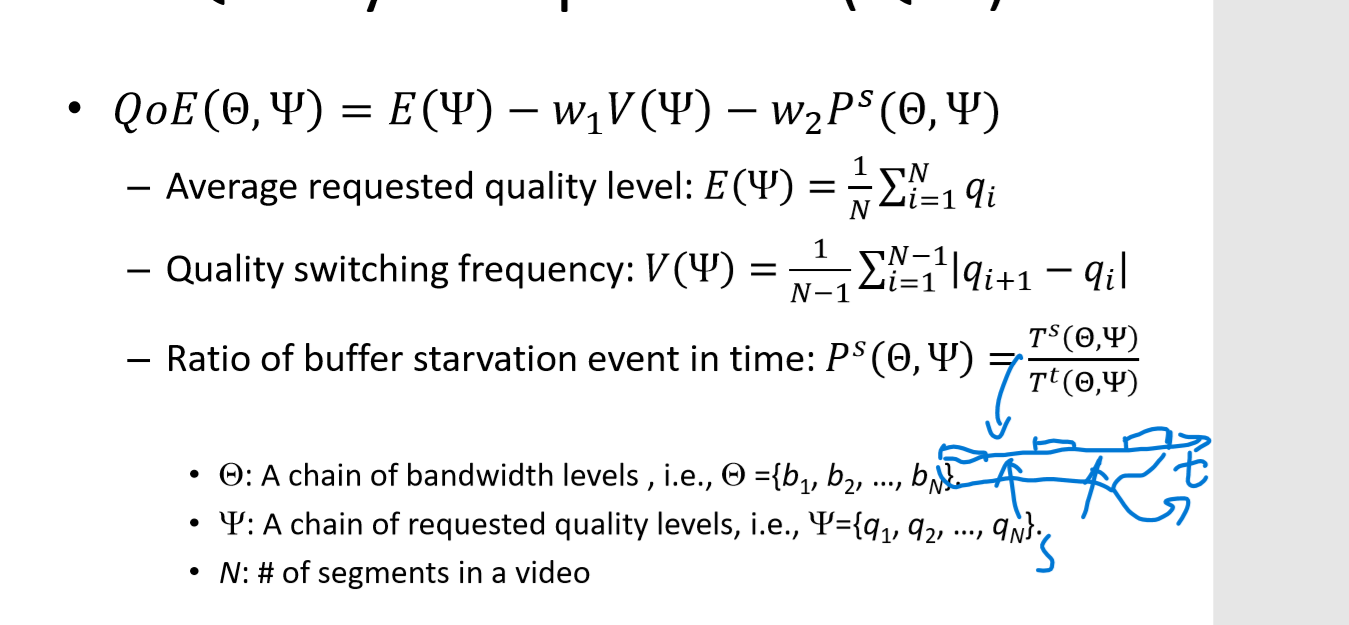
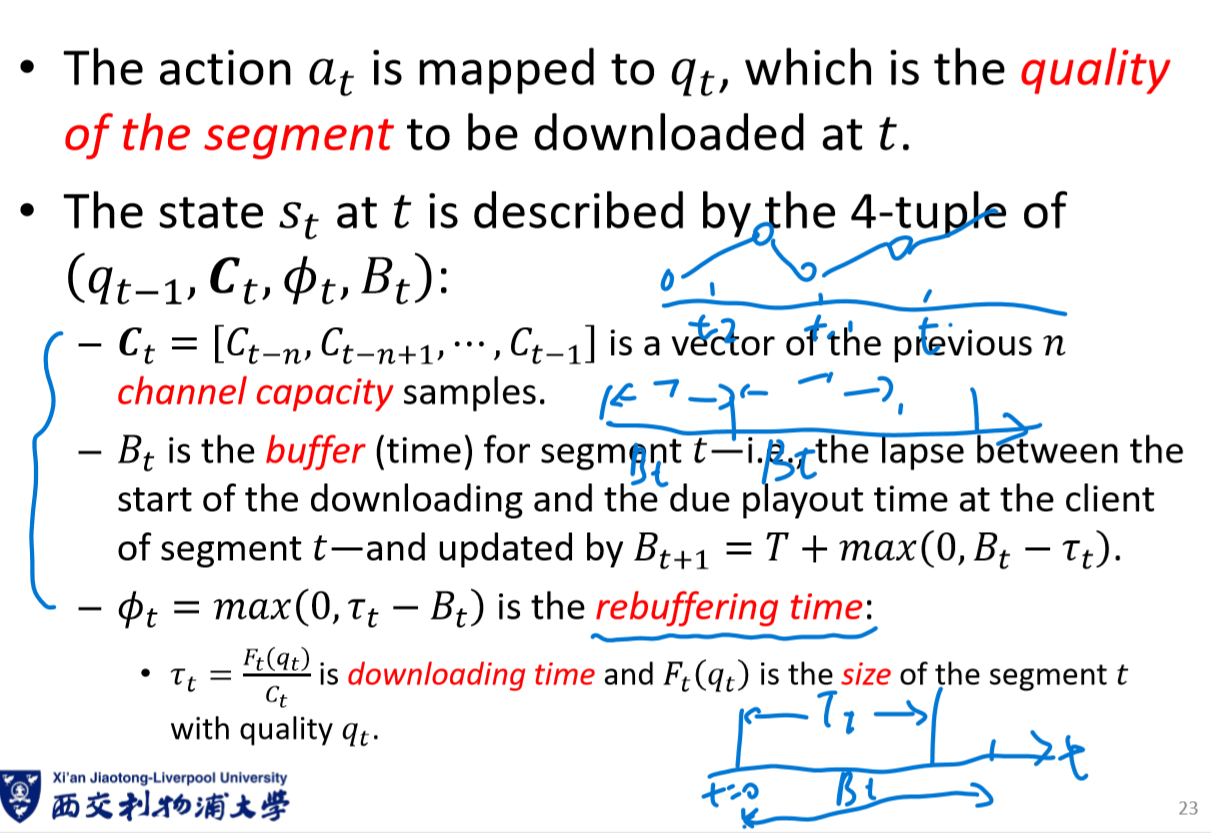


未来的带宽信息可以有效防止请求过高的quality导致bitrate过高而引起starvation，也就是buffer溢出，引起等待







for input, target in dataset:

optimizer.zero\_grad()

output = model(input)

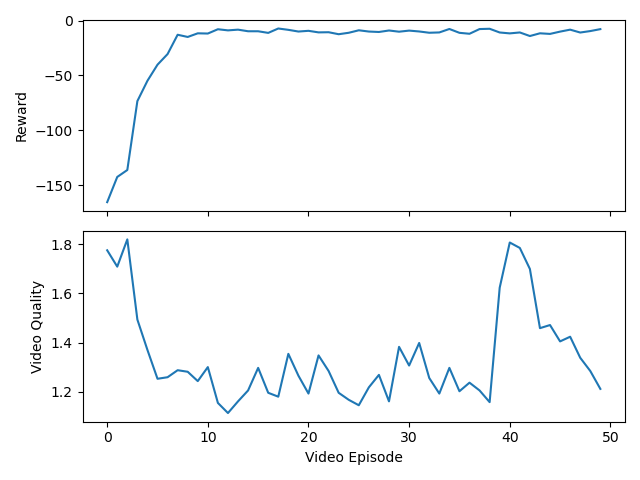
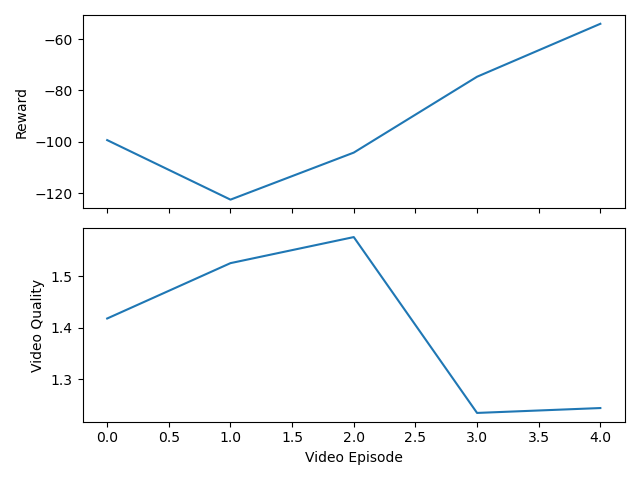
loss = loss\_fn(output, target)

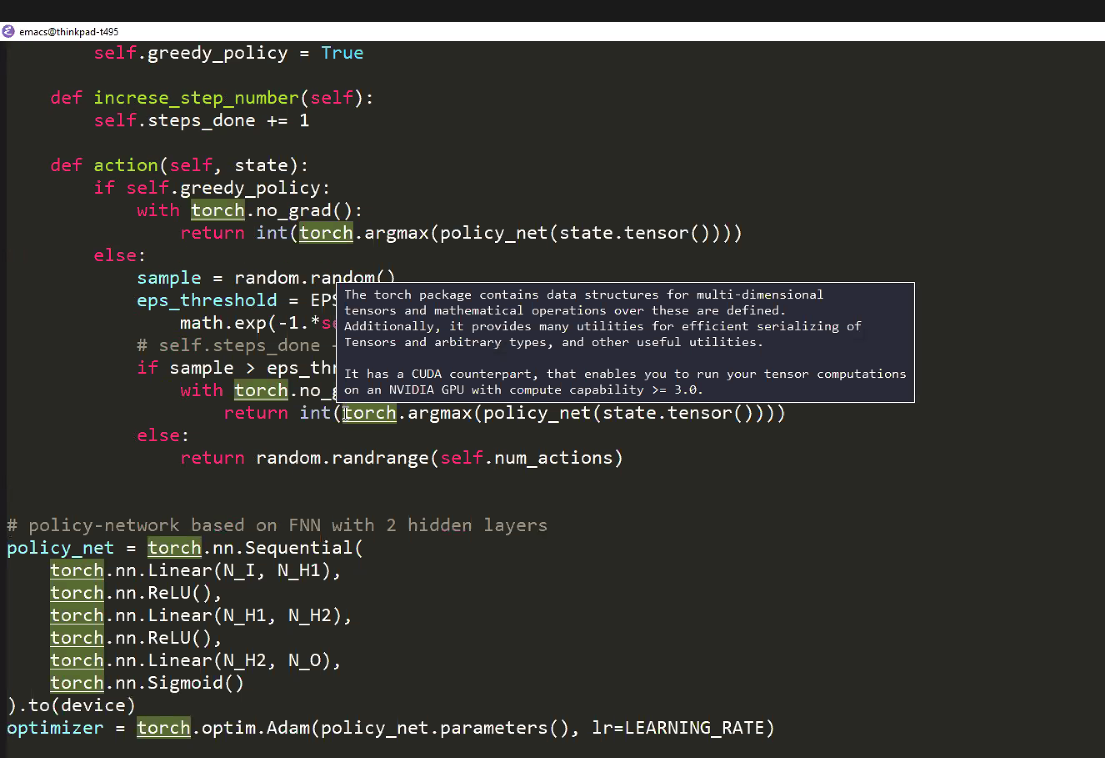
loss.backward()

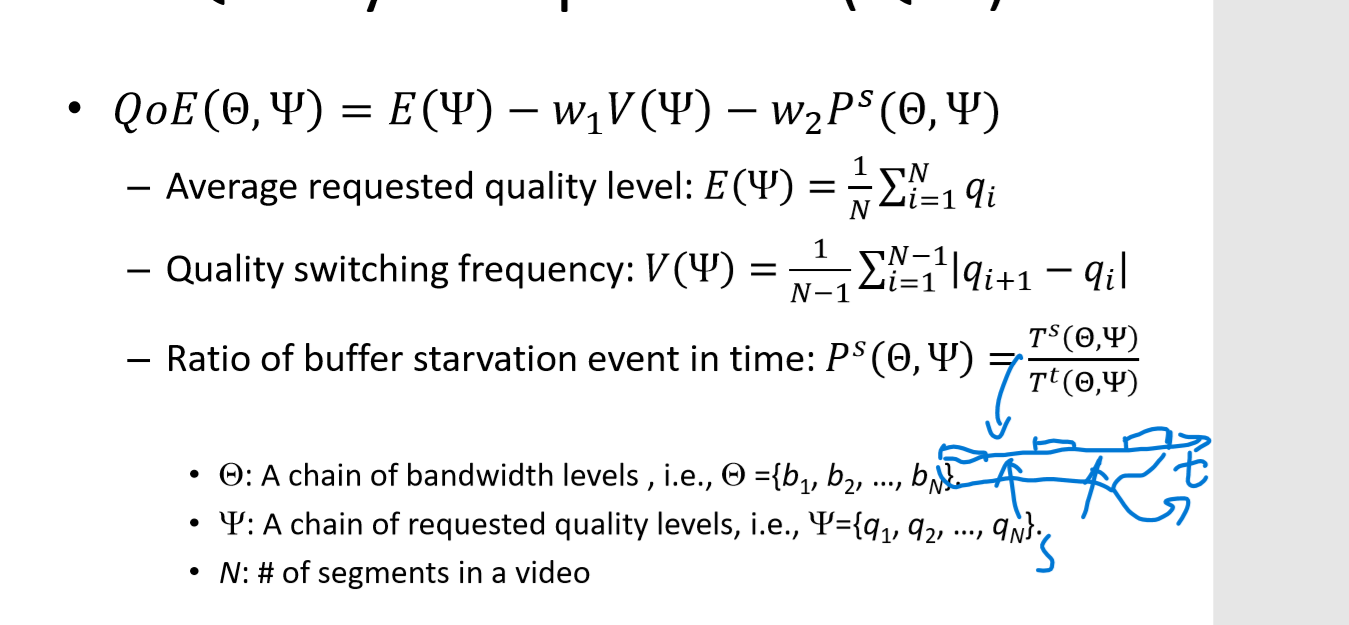
optimizer.step()

for param in policy\_net.parameters():  
 param.grad.data.clamp\_(-1, 1)  
optimizer.step()

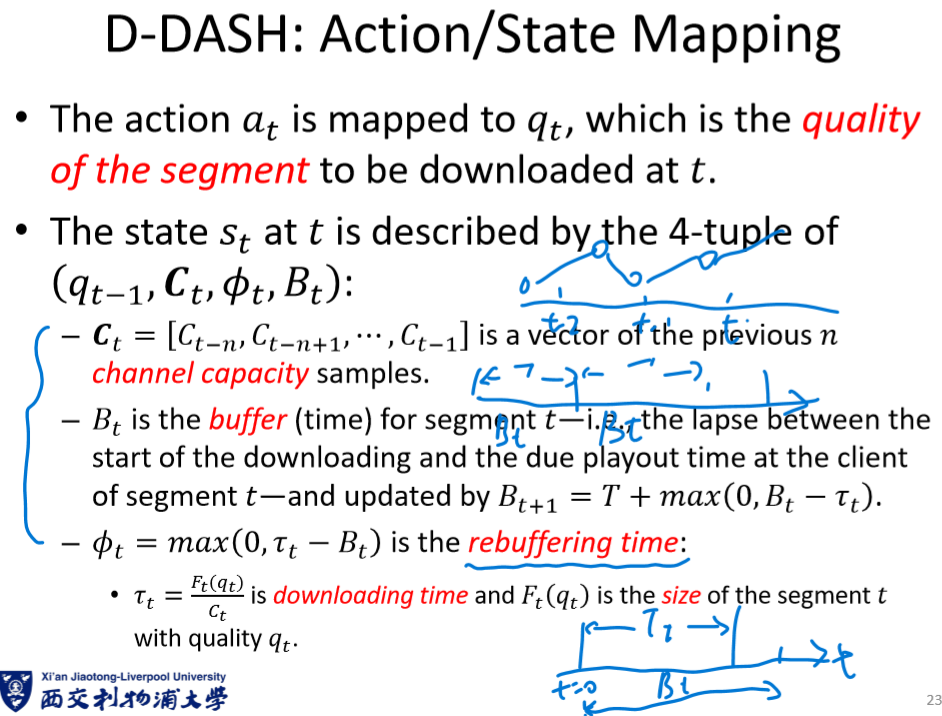
防止梯度消失或者梯度爆炸？



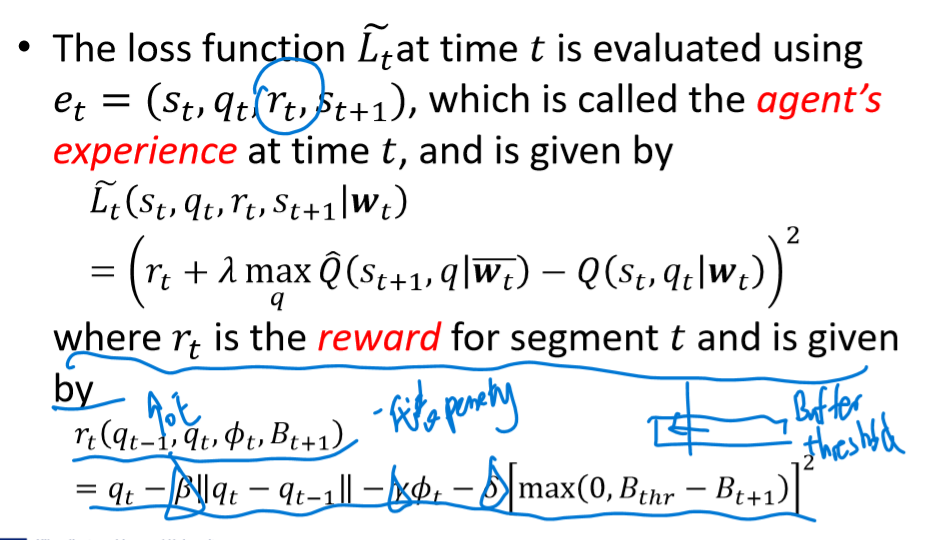




class State:  
 *"""  
 $s\_t = (q\_{t-1}, F\_{t-1}(q\_{t-1}), B\_t, \bm{C}\_t)$, which is a modified  
 version of the state defined in [1].  
 """* sg\_quality: int  
 sg\_size: float  
 buffer: float  
 ch\_history: np.ndarray



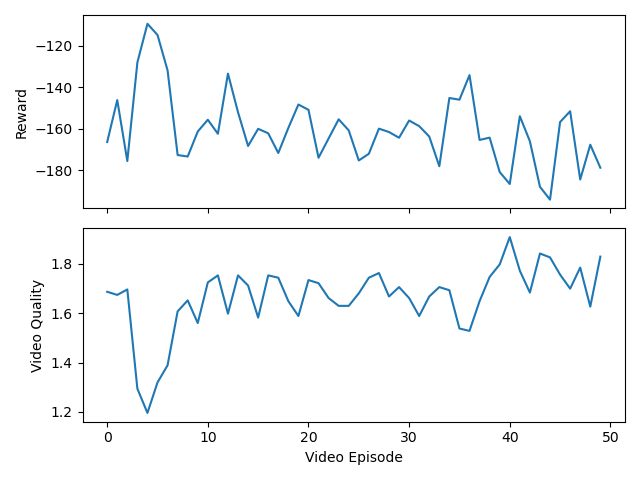
class Experience:  
 *"""$e\_t = (s\_t, q\_t, r\_t, s\_{t+1})$ in [1]"""* state: State  
 action: int  
 reward: float  
 next\_state: State



BATCH\_SIZE

和memory

50 1000怎么回事

、

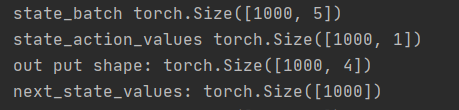
experiences.state torch.Size([5])

state\_batch torch.Size([1000, 5])

state\_action\_values torch.Size([1000, 1])

tag\_space = self.hidden2tag(lstm\_out.view(len(sentence), -1))

tag\_scores = F.log\_softmax(tag\_space, dim=1)

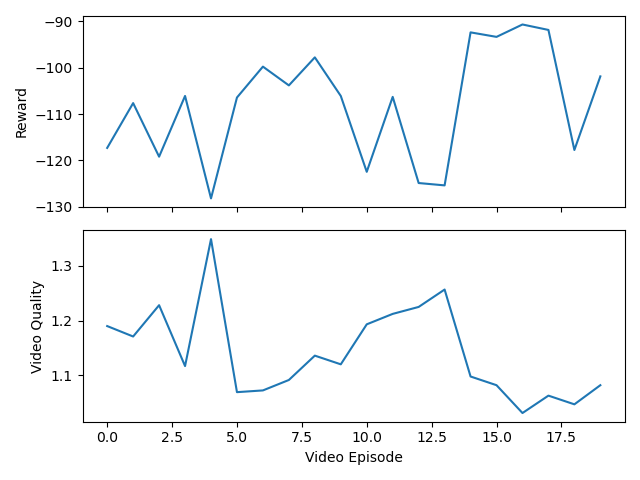


<https://zhuanlan.zhihu.com/p/105638031?utm_source=wechat_session&utm_medium=social&utm_oi=629977854172205056>

<https://zhuanlan.zhihu.com/p/104475016?utm_source=wechat_session&utm_medium=social&utm_oi=629977854172205056>

<https://zhuanlan.zhihu.com/p/29665107?utm_source=wechat_session&utm_medium=social&utm_oi=618131782479122432>

<https://zhuanlan.zhihu.com/p/29665107?utm_source=wechat_session&utm_medium=social&utm_oi=618131782479122432>



to compare the different performance of different conditions ,tiantian does many steps comparison

