안녕하세요,

BakingSoDA팀 입니다.

인퍼런스 하실 때 필요한 모델들을 첨부하겠습니다.

이 모델들을 제출한 코드 밑에 부분에서 아래와 같이 입력 해주시면 됩니다.

parser.add\_argument("--check\_model\_checkpoint", type=int, default=1000)

parser.add\_argument("--blk\_1\_process\_model\_checkpoint", type=int, default=1000)

parser.add\_argument("--blk\_2\_process\_model\_checkpoint", type=int, default=1000)

parser.add\_argument("--blk\_3\_process\_model\_checkpoint", type=int, default=1000)

parser.add\_argument("--blk\_4\_process\_model\_checkpoint", type=int, default=1000)

또한, 맨 마지막 줄에서,

run(train\_mode=-1)

하시면 됩니다.

그리고, line 611 부터 나와있는 함수들을

**def** step(self, obs, eval=False):

with torch.no\_grad():

pi = self.pi.\_distribution(obs)

if eval == False:

a = pi.sample()

else:

a = torch.argmax(pi.probs)

logp\_a = self.pi.\_log\_prob\_from\_distribution(pi, a)

v = self.v(obs)

return a.numpy(), v.numpy(), logp\_a.numpy()

**def** act(self, obs):

return self.step(obs, True)[0]

이렇게 바꾸시면 90.5 정도의 점수가 나옵니다.

저번에 제출한 코드에서는 그냥 a = pi.sample()를 사용하기 때문에 90점정도가 나옵니다.

감사합니다.

최규원 드림