Supervised SPR Prediction

Li Zhengyuan, 2021.6.9

To Reproduce

To migrate the origin algorithm to a supervised setting, we need the model to retain two capabilities: predicting the future and ensure a meaningful latent space.To the second end, I retained Q-learning loss(one-step TD) .

So our loss is:

$$L_{total} = L_{reward} + L_{spr} + L_{rl} + L_{reconstruction}$$

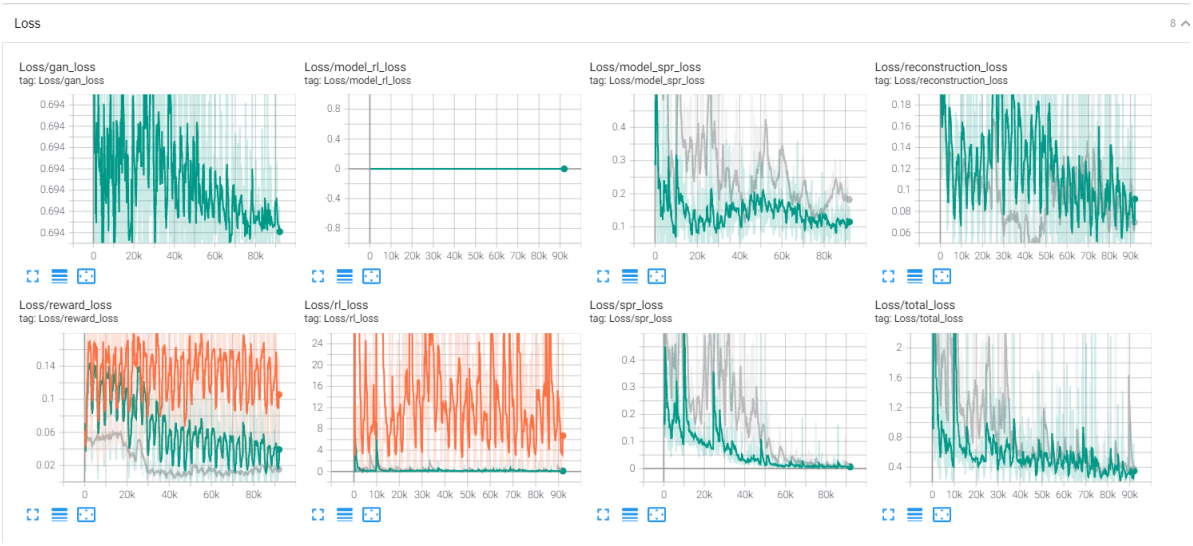

Figure 1:Training Curves, where the green curves represent training loss, the grey curves represent validation loss and the orange curves represent 'fake' loss.

Ignore the first two sub-images.We can see that the spr loss and reconstruction loss decreased steadily. To test whether the Q-prediction and reward prediction are meaningful, we add a 'fake' predcitor that always predicts zero-reward and Q value between 1 and 2.We find the our reward loss is lower than 'fake' reward loss and RL-loss is much lower, which indicates that the reward dynamics is learned.

To Understand

Static Reconstruction

Reconstruction can test whether the learned latents are meaningful.First, I use pure reconstruction loss to measure the upper-bound performance of reconstruction model.

However, I find that precisely reconstruct the image is extremely difficult, which is similar with the results of Dreamer-v2, "We hypothesize that the reconstruction loss of the world model does not encourage learning a meaningful latent representation because the most important object in the game, the ball, occupies only a single pixel".

Here's an example:

Figure 2

Everything is fine, but the ball is missing and the paddle is a little bit blurred.

I also tried using GAN-style generator, i.e. ,training a discriminator to judge whether the image is generated or original, but that doesn't work.

Sequence Reconstruction

The failure of reconstructing the ball makes understanding harder. I managed to find out some evidence.

1. Sequence without a hitting

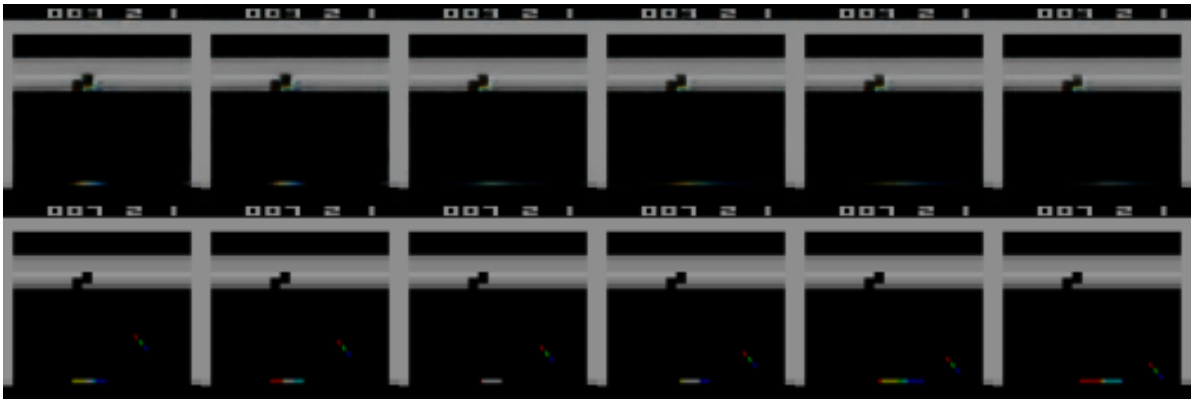

Figure 3

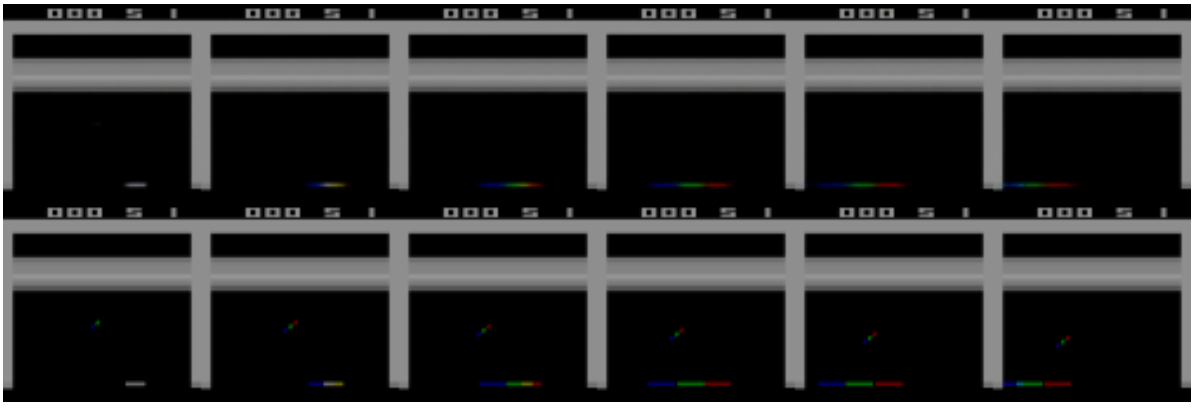

Figure 4:a Succuessful Paddle Prediction

For images without the ball hitting a block, it succeeds to reconstruct the blocks, the scores.It also tried to predict movement of the paddle.

2. Sequence with a hitting

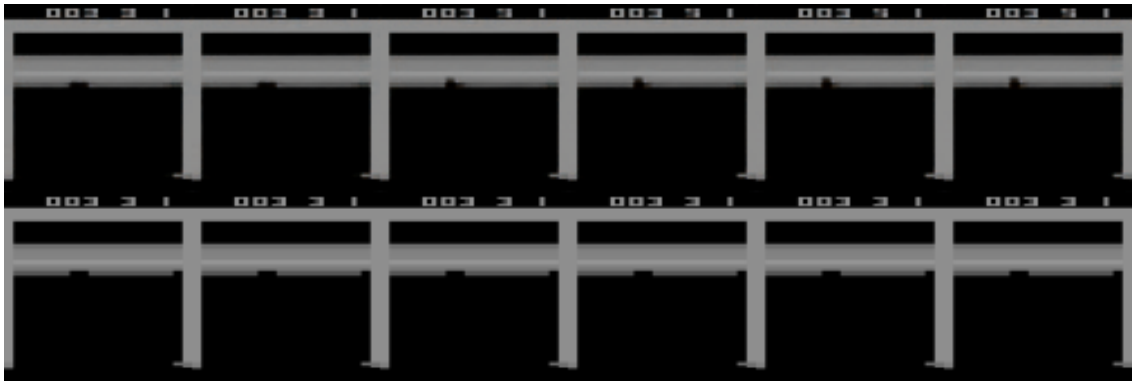

Figure 5:It showed its knowledge of the existence of hitting, but failed to predict the hitting point

Figure 6:Though the initial prediction is not good,it successfully predicted the hitting point

Figure 5 and Figure 6 show that it has some knowledge of hitting, but the predciction task is hard.

3. The length of sequence

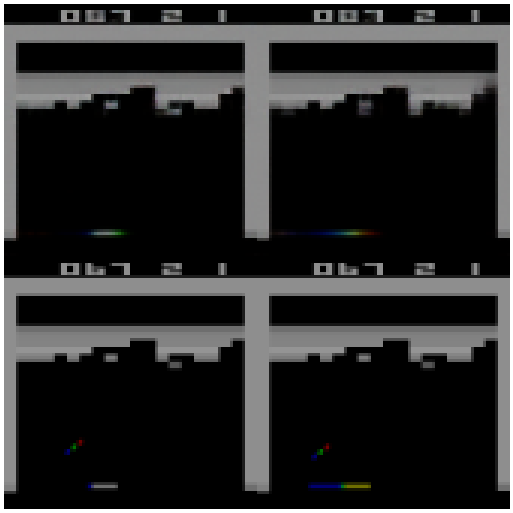

Figure 7:One-step Prediction

Figure 8:Three-step Prediction

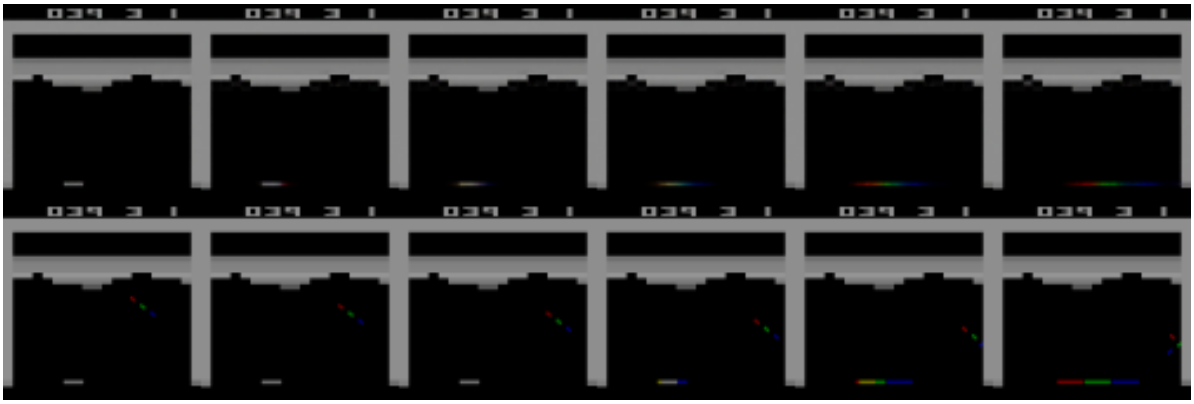

Figure 9:Five-step Prediction

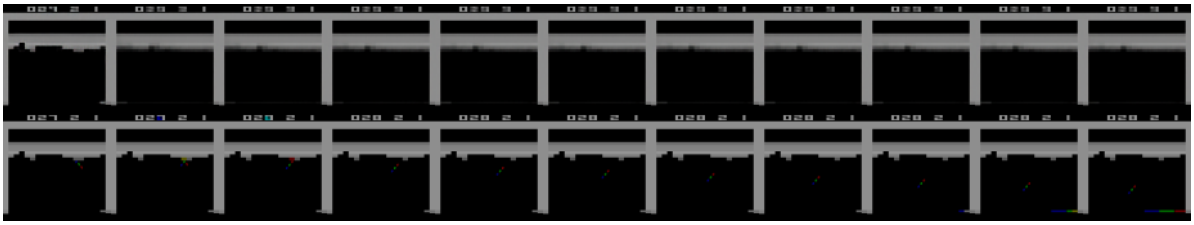

Figure 10:Ten-step Prediction

For different prediction horizons, I find that 1-step,3-step and 5-step prediction succeed in retaining the landscape of disappeared blocks, but 10 step-prediction fails.I hypothesize that's because long-term prediction makes the task more complex, and may require more training time or more sophisticated networks.

Implement Details

- Dataset: Each checkpoint contains 1 million samples, we have 250 checkpoints in total, so the full dataset requires $1e6 * 84 * 84 * 250 = 1764GB$ memory.So I downsample by choosing the first in every 20 checkpoints.That's reasonable since the checkponts themselve overlap with their neighbours.
- FrameWork: For simplicity , I do not use data augmentation.
- Training: I found that the training is slow because of the sheer volume of data, the results above are the ability of our network after go over the dataset about 3 times(depending on the length of sequence).

Future Works

Due to the limited time and resources, the experiments are not exhausted.There are few things I would do if given more time:

- Test whether the reward loss and DQN-loss meaningful: the meaning of DQN-loss is suspicous since the policy is always changing.Reconstuction itself is enough for avoiding the feature collapse, so do we still need reward loss?
- Try more representation loss to improve the performance.
- Try GAN-style loss to enable reconstructing the ball， the problem can be stated as:Given image A and its latent feature B, we want to reconstruct image A with latent feature B.That's not the same as the original GAN, which tried to reconstruct images from noise.I believe there will be some CV techniques to handle that problem.

Reference

[1] Schwarzer M. Data-efficient reinforcement learning with self-predictive representations[J]. 2021.

[2] Hafner D, Lillicrap T, Norouzi M, et al. Mastering atari with discrete world models[J]. arXiv preprint arXiv:2010.02193, 2020.

[3] Goodfellow I J, Pouget-Abadie J, Mirza M, et al. Generative adversarial networks[J]. arXiv preprint arXiv:1406.2661, 2014.