

환경 복잡도에 따른 시계열 데이터 처리 시 LSTM 과 프레임 스택킹 방법의 성능 비교

박영주, 박도희, 서보승

한국항공대학교

andy1andy@kau.kr, lahee0803@kau.kr, sbs7696@kau.kr

A Comparative Study on the Performance of LSTM and Frame Stacking Methods in Time Series Data Processing Based on Environmental Complexity

Youngju Park, Dohee Park, Boseung Seo
Korea Aerospace Univ.

요 약

본 논문에서는 환경 복잡도에 따른 'Frame Stacking 을 적용한 PPO' (이하 P-Frame Stacking)와 'LSTM 을 적용한 PPO' (이하 P-LSTM) 두 모델간 성능을 비교한다. OpenAI Gym 의 Pendulum 환경에서 관측 난도를 달리하여 3 가지 환경(완전 관측; Fully Observable, 부분 관측; Partially Observable, 이미지 입력; Image Input)으로 변형하였고, 세 환경에 대하여 두 모델을 각각 학습하고 성능을 비교한다. 실험 결과, Pendulum 관측 환경의 난도가 증가함에 따라 P-LSTM 이 더 적합함을 확인한다.

I. 서 론

강화학습에서 시계열 데이터를 처리하는 대표적인 방법으로 Frame Stacking 과 순환 신경망(RNN)이 존재한다. 두 방법의 성능 차이는 문제의 특성 및 환경적 요인에 따라 달라질 수 있다. 본 논문에서는 OpenAI Gym 의 Pendulum 환경에서 관측 난도를 조정하여 세 가지 서로 다른 환경을 구성하고, P-Frame Stacking 기법과 P-LSTM 기법의 성능을 비교한다. 이를 통해 Pendulum 의 관측 환경 난도에 따른 두 모델의 적합성을 평가한다. 또한 기존의 벡터 환경 학습 연구를 보완하고자, 본 연구에서는 이미지 프레임 환경으로 새롭게 정의된 학습을 통해 모델 성능을 평가한다.

II. 실험 환경 및 모델 정의

2.1 Pendulum 제어

진자의 운동 방정식은 물리 법칙에 의해 정의되며, 아래와 같이 표현된다. 에이전트는 이 방정식을 바탕으로 얻어진 현재 상태(관측값)를 입력으로 받아 제어 입력인 토크를 계산하여, 진자가 불안정한 평형점인 상방향에서 안정적으로 균형을 유지하도록 한다.

$$I\ddot{\theta} + b\dot{\theta} + mgl \sin \theta = T \quad [1]$$

2.2 Observation 정의

환경 1	완전 관측 환경 (벡터 입력 환경) (Fully Observable Environment)
	관측값 = [Pendulum x 좌표, y 좌표, 각 속도] Observation 예시: [0.9986 0.0523 -0.1877]
환경 2	부분 관측 환경 (벡터 입력 환경) (Partially Observable Environment)
	관측값 = [Pendulum x 좌표, y 좌표] Observation 예시: [0.9986 0.0523]

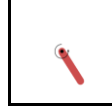
환경 3	이미지 입력 환경 (Image Input Environment)
	관측값 = 500 x 500 Pixel, RGB Image Input Observation 예시 : 

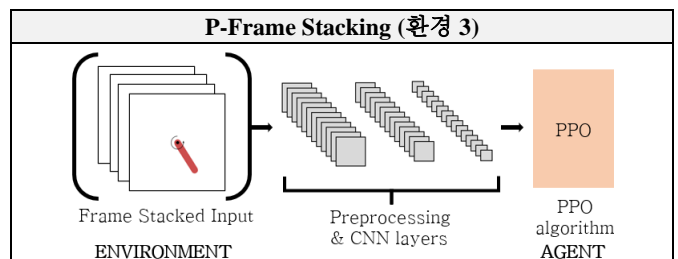
표 1. 난도에 따른 3 가지 환경

2.3 모델 정의

P-Frame Stacking. PPO 알고리즘에 Frame Stacking 을 추가한 모델이다. Frame Stacking 은 각 스텝마다 최근 4 개의 관측값(벡터 또는 이미지)을 쌓아 하나의 새로운 관측값을 구성하는 것으로 정의한다.

P-LSTM. PPO 알고리즘에 LSTM 층을 추가한 모델이다. 각 스텝에서 단일 관측값(벡터 또는 이미지)을 활용해 학습이 이뤄지며, 대표적인 순환 신경망으로 알려진 LSTM 을 활용한다.

두 모델 모두 환경 1 과 2 에서 벡터 데이터를 MLP 층을 통해 학습한다. 아울러, 환경 3 에서는 CNN 층을 추가해 이미지에서 특징 벡터를 추출한 후 MLP 층을 통해 학습을 진행한다.



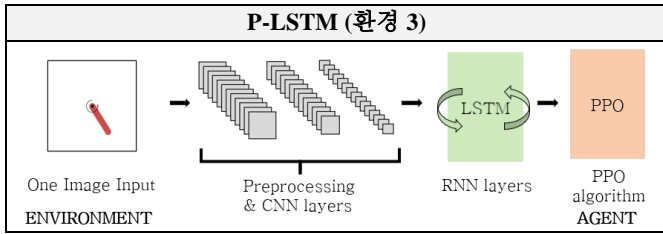


표 2. P-Frame Stacking, P-LSTM의 환경 3 적용 예시

III. 실험 결과

학습이 완료된 두 모델을 100 번의 에피소드에 대하여 테스트한 결과(표 3), 환경 1에서는 P-Frame Stacking이 P-LSTM의 성능을 월등히 뛰어 넘었으며, 학습 속도 또한 P-Frame Stacking이 P-LSTM보다 빨랐다.

환경 2에서는 환경 1과 마찬가지로 양상을 보였지만, P-Frame Stacking의 성능이 환경 1과 비교했을 경우 비교적 감소한 반면, P-LSTM의 성능은 미세하게 개선된 것을 알 수 있다.

마지막으로, 환경 3에서는 벡터가 아닌 이미지 입력 환경을 통한 학습으로, 환경난도가 크게 증가해 두 모델 모두 낮은 성능을 보인다. 한편, P-Frame Stacking은 그림 3에서 보이듯 학습이 전혀 진행되지 않지만, P-LSTM은 시간이 지남에 따라 학습이 진행되어 P-Frame Stacking의 성능을 현저히 뛰어 넘은 것을 알 수 있다.

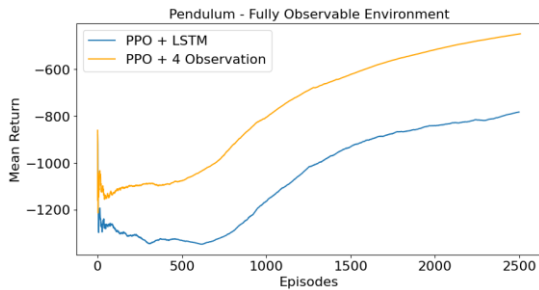


그림 1. 환경 1에서의 두 모델의 평균 보상

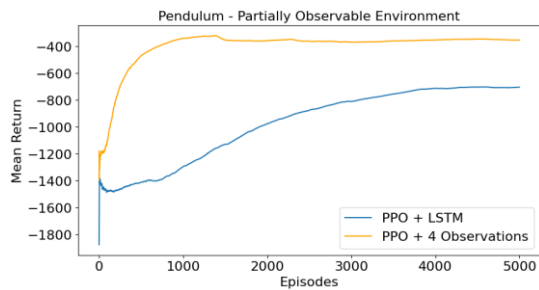


그림 2. 환경 2에서의 두 모델의 평균 보상

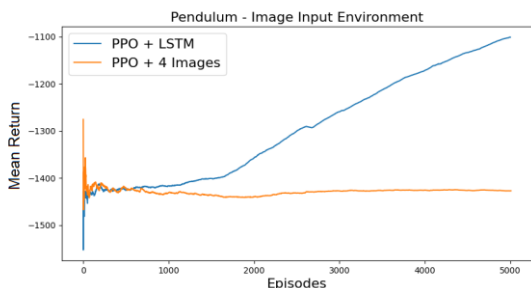


그림 3. 환경 3에서의 두 모델의 평균 보상

	P-Frame Stacking	P-LSTM
환경 1	-175.92 ± 114.44	-592.84 ± 79.61
환경 2	-354.71 ± 234.05	-584.38 ± 75.91
환경 3	-1386.18 ± 224.57	-727.30 ± 139.20

표 3. 환경에 따른 두 모델의 테스트 결과

IV. 결론

본 논문에서는 고전적 제어 환경인 Pendulum 환경에 변화를 주어 observation 난도에 따른 P-Frame Stacking와 P-LSTM의 적합도를 비교한다. 그 결과, 환경 1의 경우 P-Frame Stacking이 P-LSTM보다 우수한 성능을 보이지만, 환경이 복잡해질수록 P-LSTM의 성능이 P-Frame Stacking보다 상대적으로 높아지는 것을 알 수 있으며, 이는 그림 1, 2, 3과 표 3를 통해 명확히 나타나는 것을 알 수 있다. 이러한 결과를 바탕으로, 관측 환경의 난도가 증가할수록 P-LSTM의 적합도가 증가한다는 결론에 이른다.

다만, 본 연구는 Pendulum이라는 단일 환경에서만 실험을 진행한 한계가 존재하므로, 후속 연구에서는 더욱 다양한 환경을 통해 두 모델 간의 성능 차이를 일반화할 필요가 있다. 특히 본 연구는 기존 연구들과 달리 이미지 프레임을 도입했다는 차별점이 존재하며, 환경 3에서 P-LSTM이 P-Frame Stacking에 비하여 뛰어난 성능을 보인 것에 주목할 만하다. 이러한 이미지 프레임 환경과 CNN 층이 포함된 순환신경망 모델을 활용한다면, Gymnasium과 같은 가상 환경의 제어뿐만 아니라 카메라와 같은 장비를 활용하여 현실 세계에서보다 범용적인 적용 가능성을 기대할 수 있을 것이다.

참 고 문 헌

- [1] OpenAI, "Pendulum Environment," (https://gymnasium.farama.org/environments/classic_control/pendulum/).
- [2] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal Policy Optimization Algorithms," *arXiv preprint arXiv:1707.06347*, vol. 17, no. 7, pp. 1-12, Aug. 2017.
- [3] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997.
- [4] W&B, "PPO vs RecurrentPPO (aka PPO LSTM) on environments with masked velocity [SB3 Contrib]," 2023, ([https://wandb.ai/sb3/no-vel-envs/reports/PPO-vs-RecurrentPPO-aka-PPO-LSTM-on-environments-with-masked-velocity-SB3-Contrib---VmldzoxOTI4NjE4#ppo-lstm-vs-ppo-\(no-framestack\)\)](https://wandb.ai/sb3/no-vel-envs/reports/PPO-vs-RecurrentPPO-aka-PPO-LSTM-on-environments-with-masked-velocity-SB3-Contrib---VmldzoxOTI4NjE4#ppo-lstm-vs-ppo-(no-framestack)))).
- [5] Stable-Baselines3 Contributors, "Stable Baselines3," 2023, (<https://github.com/DLR-RM/stable-baselines3>).