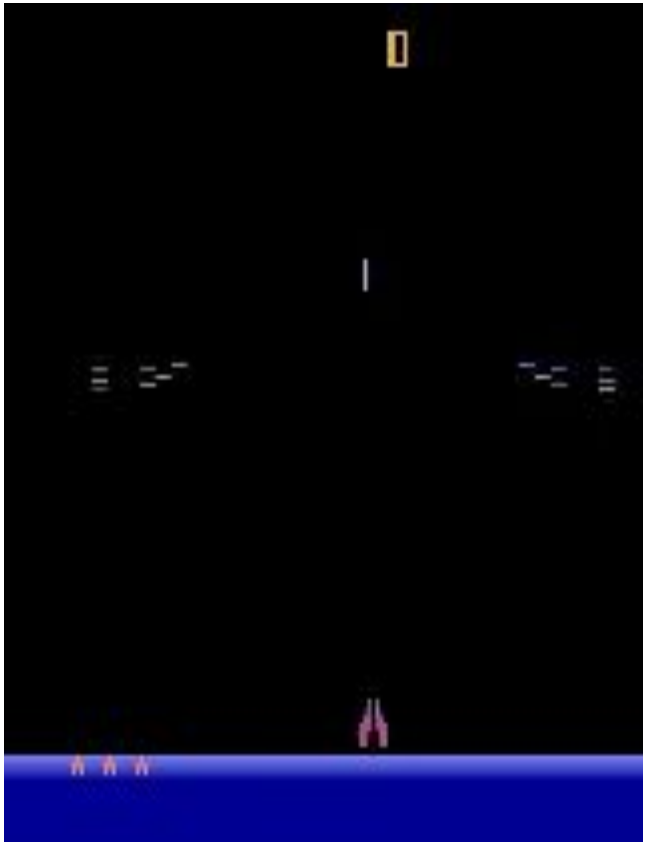
Research on the usage of target networks in deep reinforcement learning (Machine Learning 2022 Course)

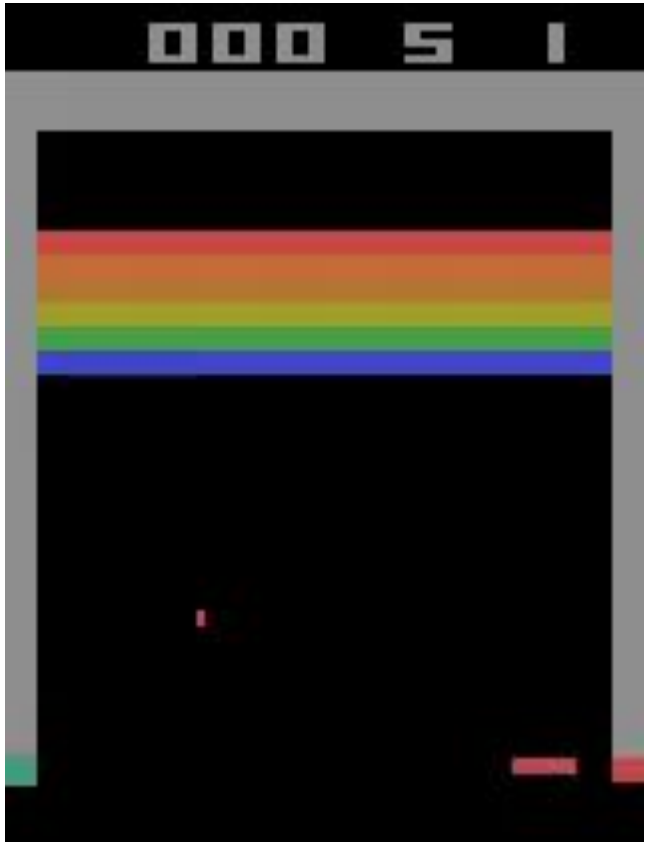
Andrey Spiridonov, Vladislav Trifonov,
Yerassyl Balkybek, Nikolay Sheyko, Dmitrii Gromyko

Hypothesis on the usage of the target networks:

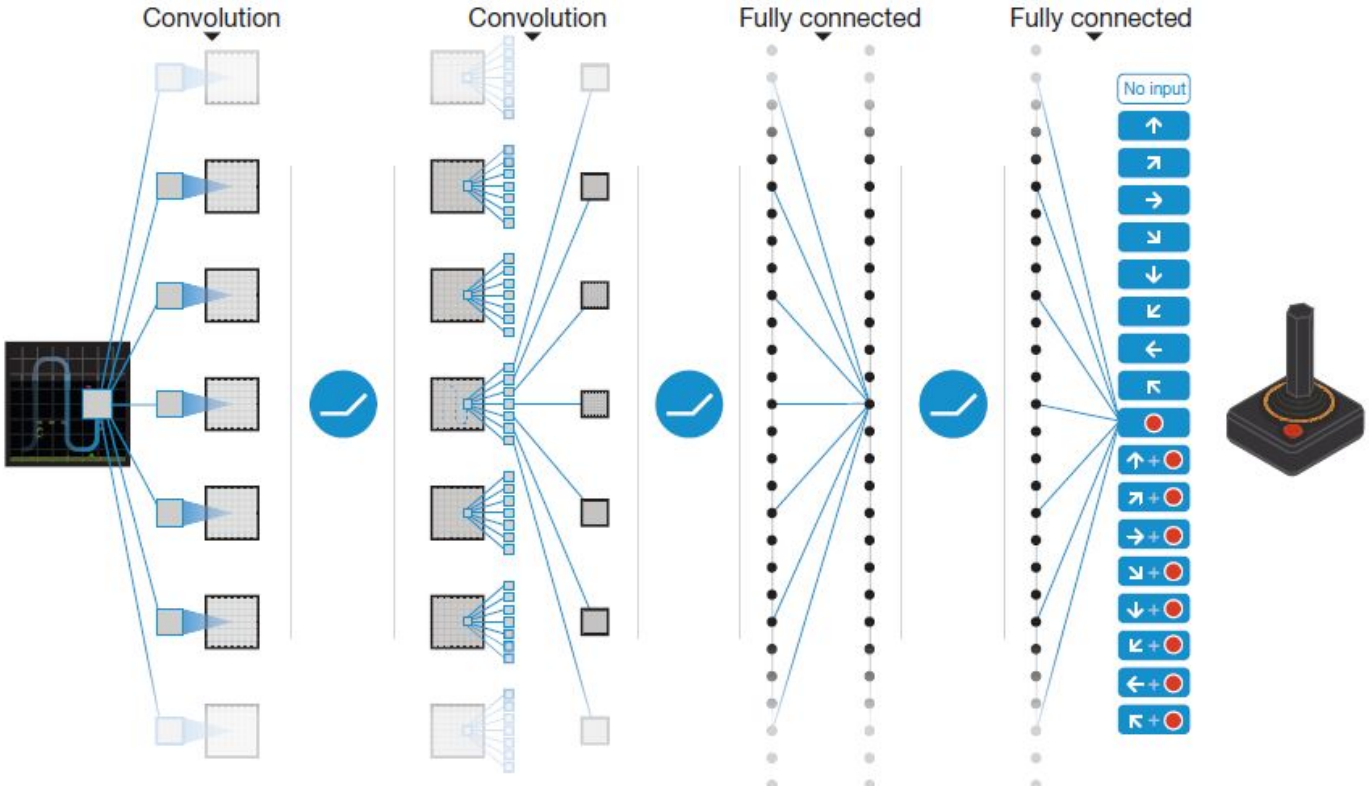
Larger sizes of neural networks positively impact the stability of the algorithm,
so that at some point using the target network will be pointless.

Our trained agents in two games:
Demon Attack and Breakout





Schematic illustration of the convolutional neural network.



Mnih, V., Kavukcuoglu, K., Silver, D. *et al.* Human-level control through deep reinforcement learning. *Nature* 518, 529–533 (2015). https://doi.org/10.1038/nature14236

Deep Q-network

Action-value function:

$$Q^*(s,a) = \max_{\pi} \mathbb{E}\left[r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \ldots \mid s_t = s, a_t = a, \pi\right]$$

Bellman equation:

$$Q_{i+1}(s,a) = \mathbb{E}_{s'}\left[r + \gamma \max_{a'} Q_i\left(s', a'\right) \mid s, a\right], Q_i \to Q^* \text{ as } i \to \infty$$

Approximation with convolutional neural network: $Q(s,a;\theta_i) \approx Q^*(s,a)$

Loss function with $y = r + \gamma \max_{a'} Q\left(s', a'; \theta_i^-\right)$:

$$L_i\left(\theta_i\right) = \mathbb{E}_{s,a,r}\left[\left(\mathbb{E}_{s'}[y \mid s,a] - Q\left(s,a;\theta_i\right)\right)^2\right] = \mathbb{E}_{s,a,r,s'}\left[\left(y - Q\left(s,a;\theta_i\right)\right)^2\right] + \mathbb{E}_{s,a,r}\left[\mathbb{V}_{s'}[y]\right]$$

Double DQN

$$y_i^{DQN} = r + \gamma \max_{a'} Q\left(s', a'; \theta^-\right) \rightarrow y_i^{DDQN} = r + \gamma Q\left(s', \max_{a'} Q\left(s', a'; \theta_i\right); \theta^-\right)$$

Duelling DQN

Two sequences or streams of fully connected layers provide a value function $V^\pi(s)$, and state-dependent action function $A^s(s, a)$:

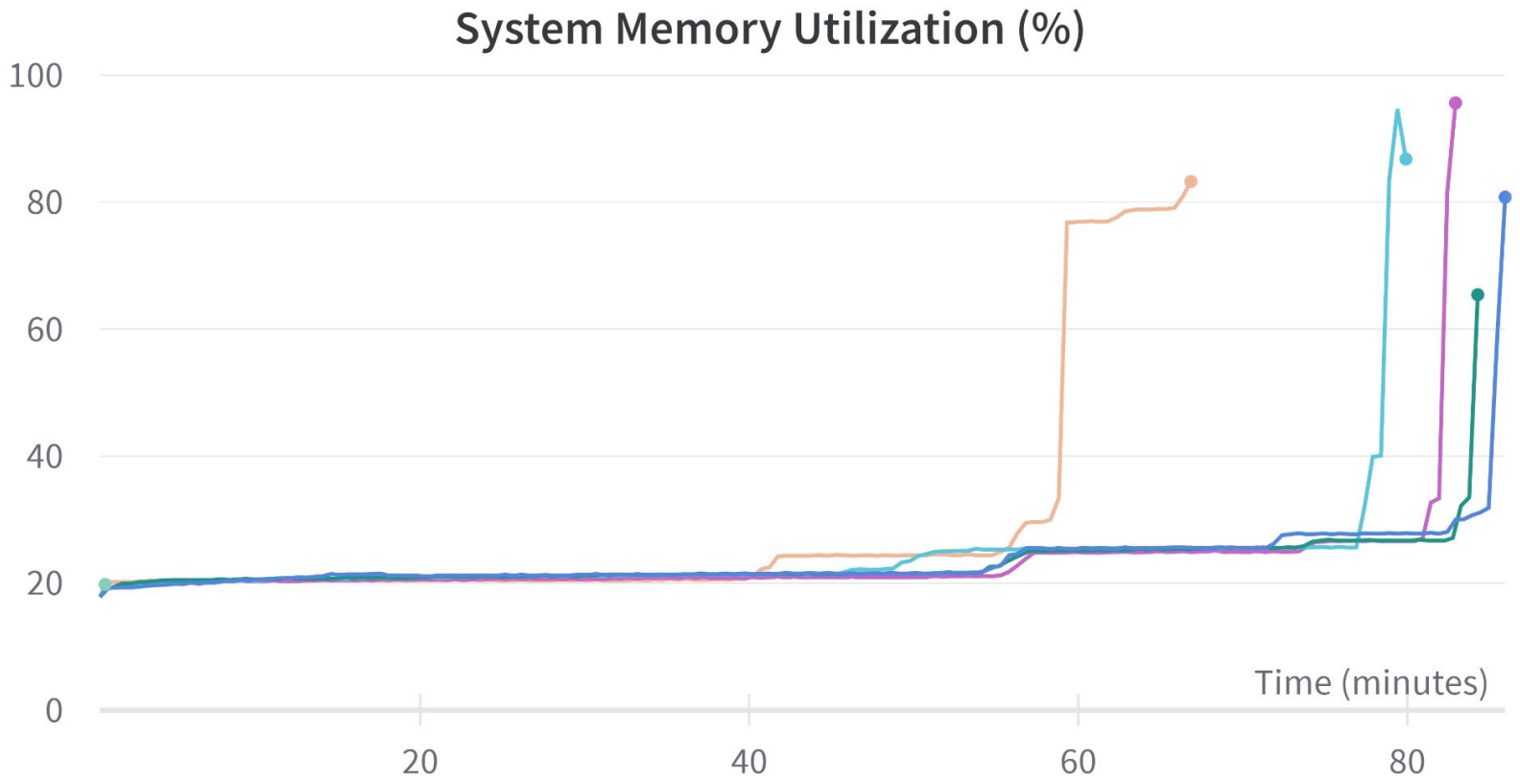
$$V^\pi(s) = \mathbb{E}_{a \sim \pi(s)}\left[Q^\pi(s, a)\right]$$

$$A^\pi(s, a) = Q^\pi(s, a) - V^\pi(s)$$

Action-value function:

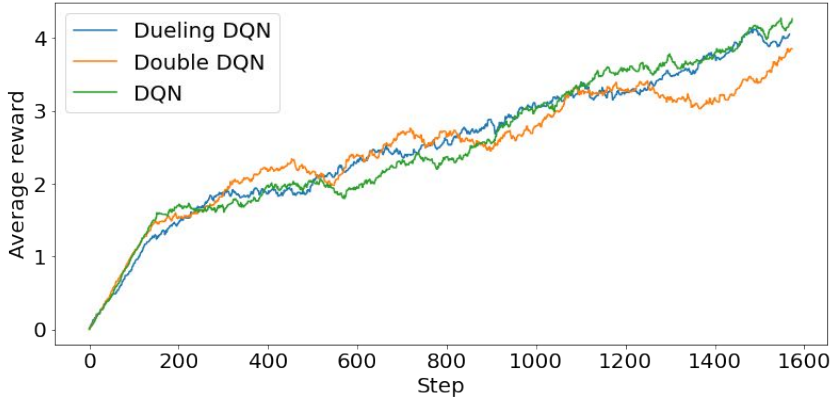
$$Q(s, a; \theta, \alpha, \beta) = V(s; \theta, \beta) + \left(A(s, a; \theta, \alpha) - \frac{1}{|A|} \sum_{a'} A(s, a'; \theta, \alpha)\right)$$

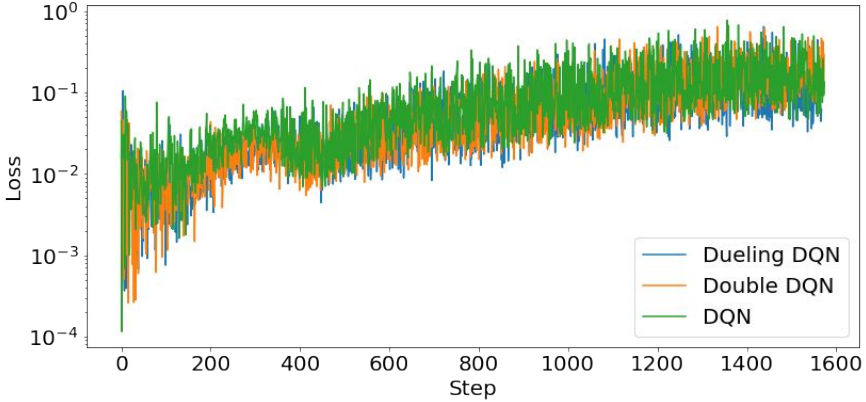
All runs terminated due to the Colab memory collapses



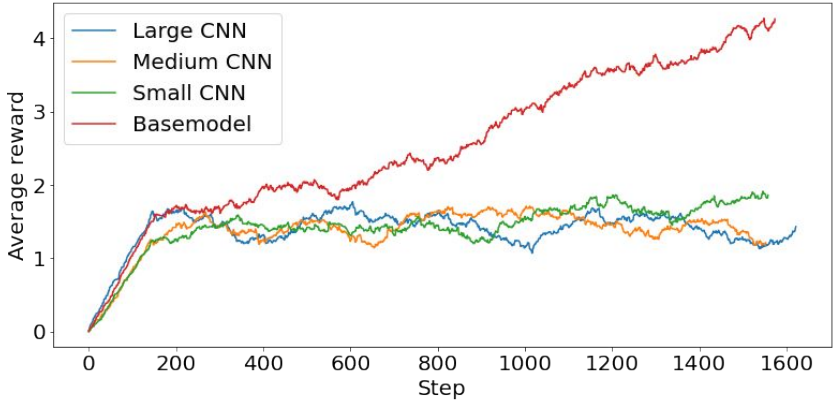
Environment: **Breakout**

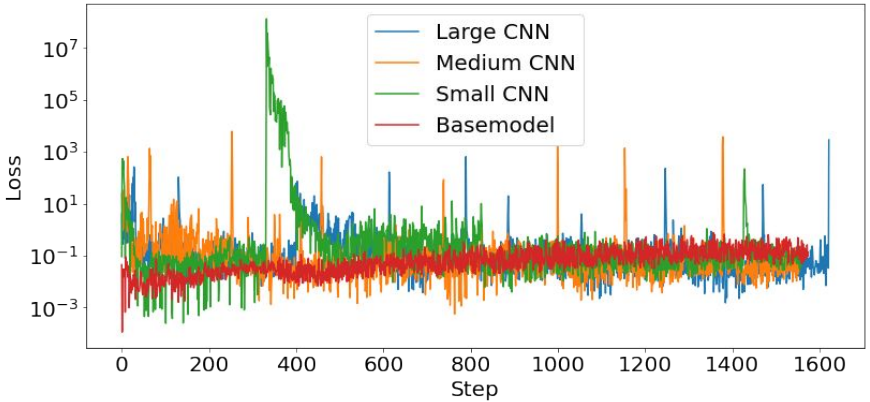
Comparison of models





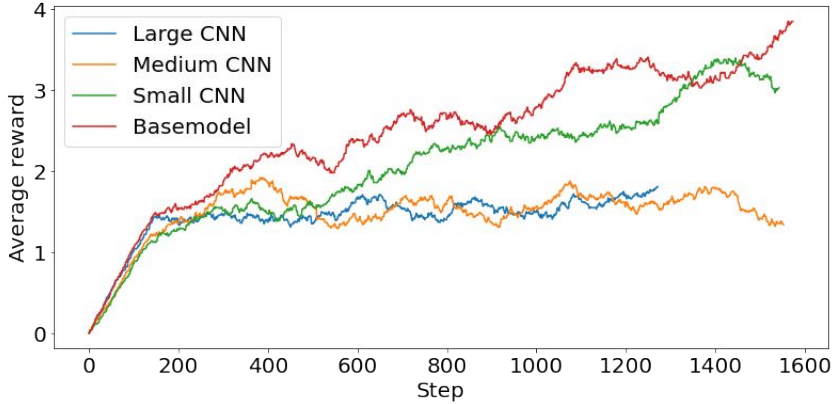
Deep Q-Network

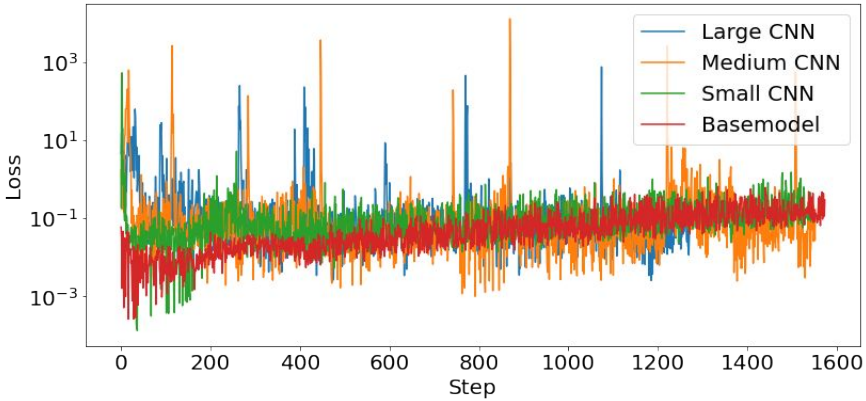




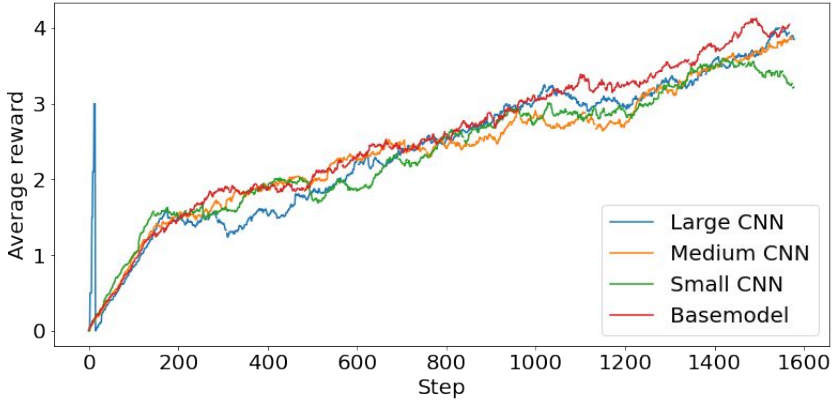
Environment: **Breakout**

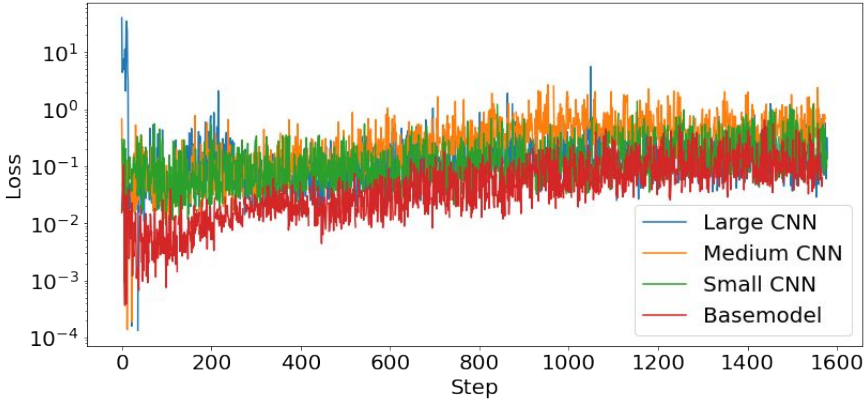
Double DQN





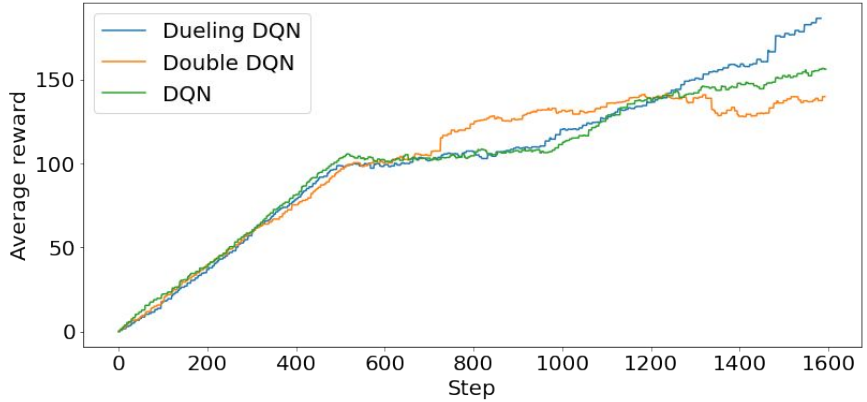
Dueling DQN

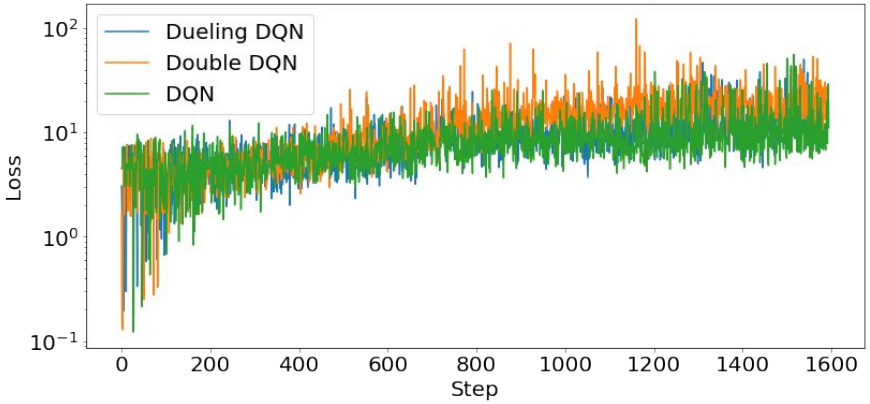




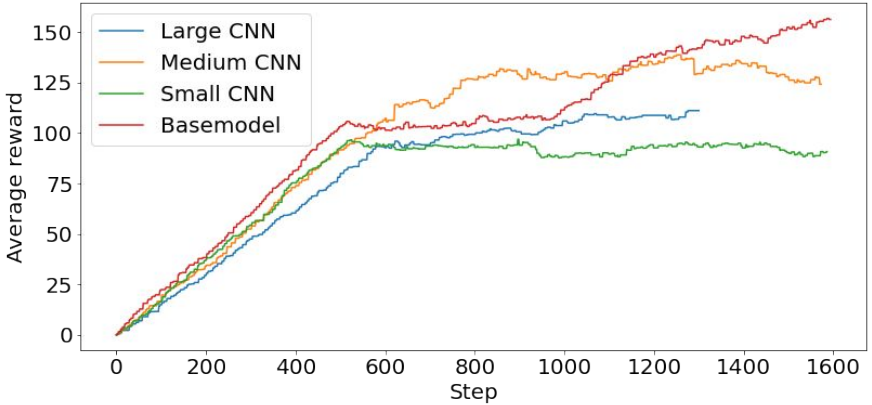
Environment: **Demon Attack**

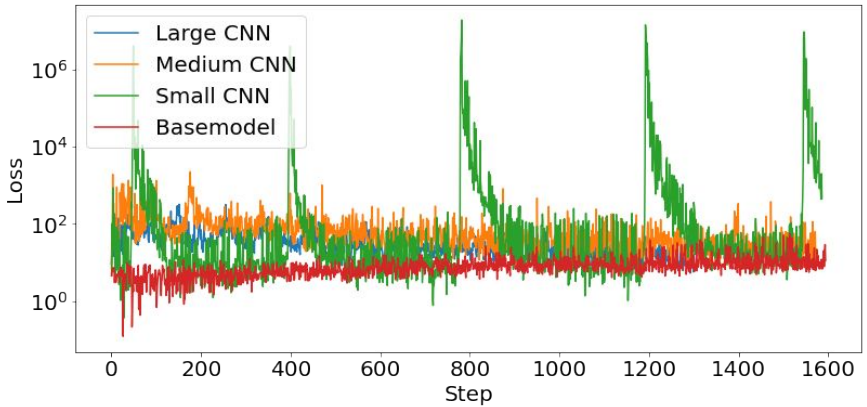
Comparison of models





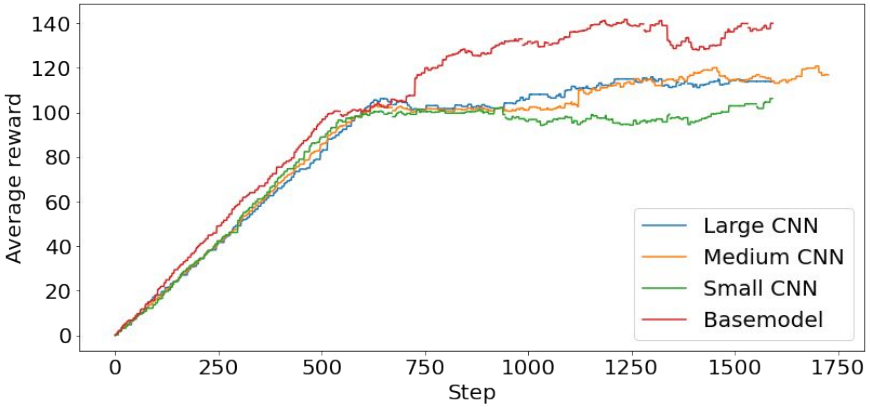
Deep Q-Network

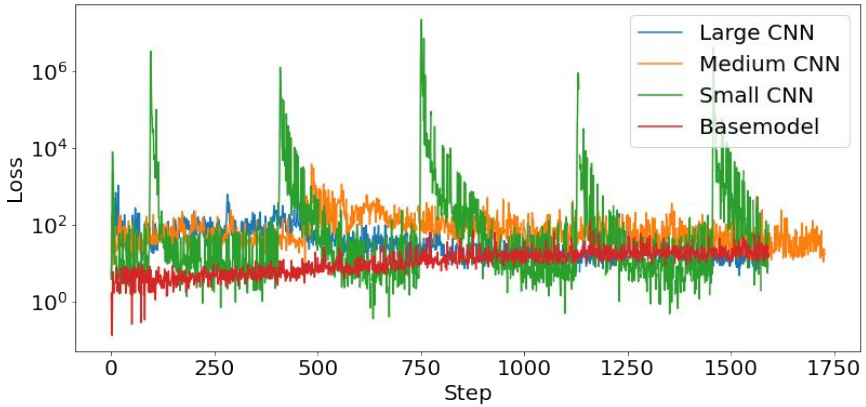




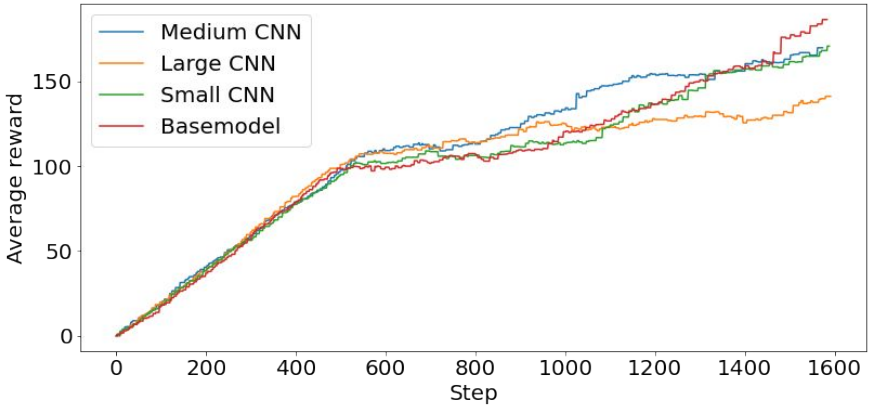
Environment: **Demon Attack**

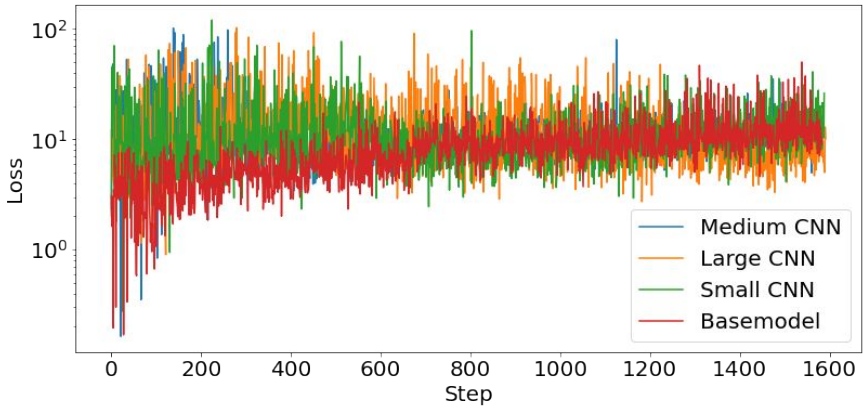
Double DQN



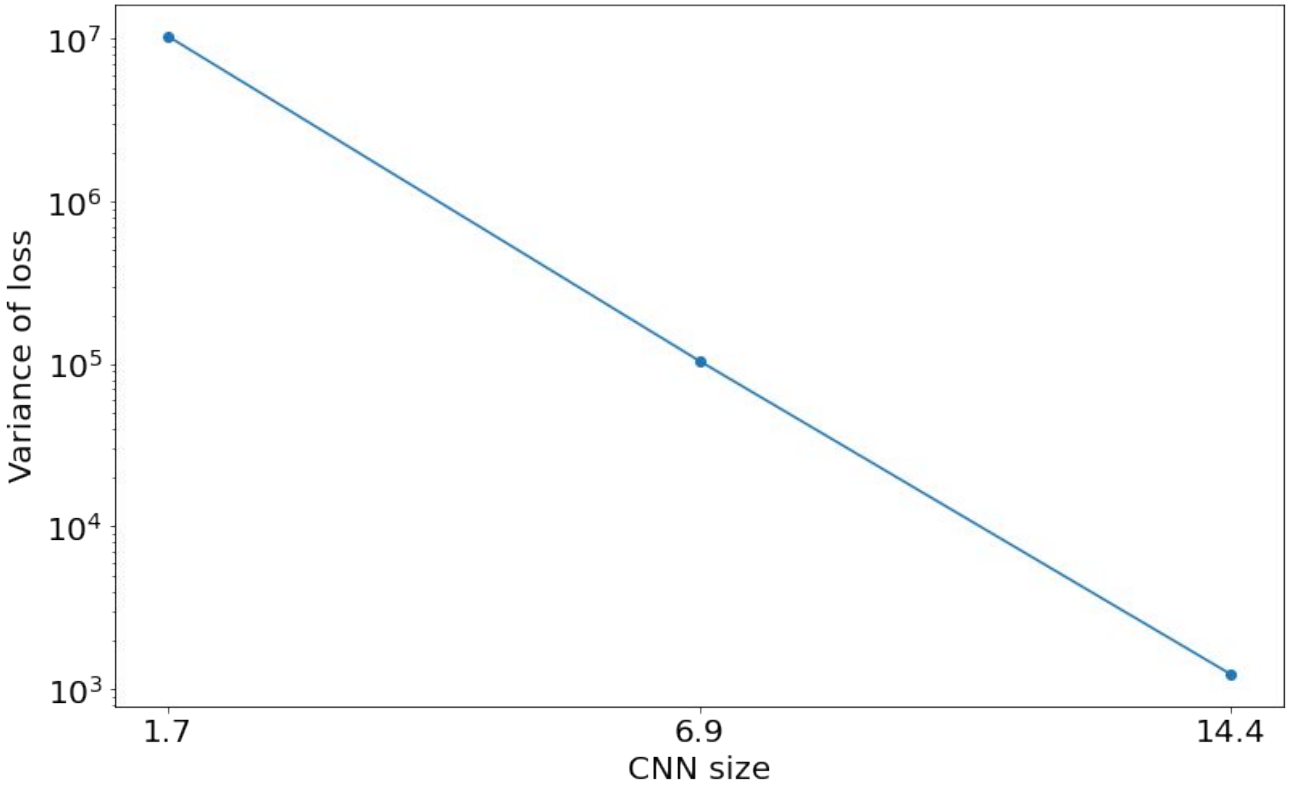


Dueling DQN



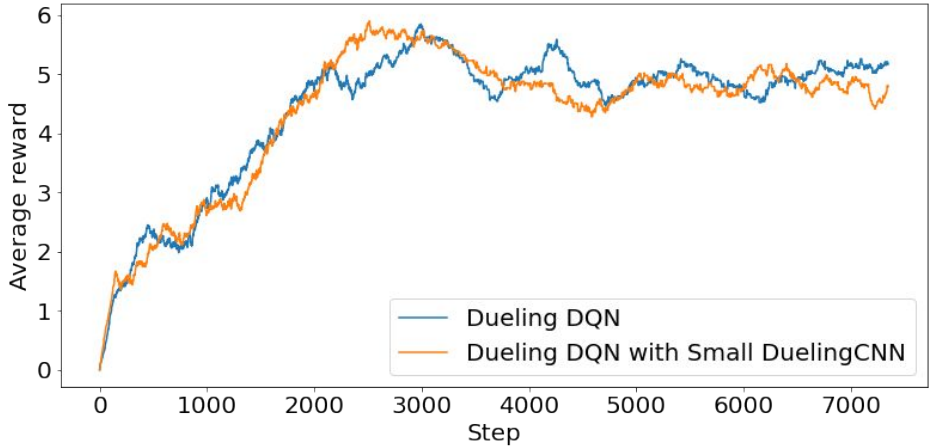


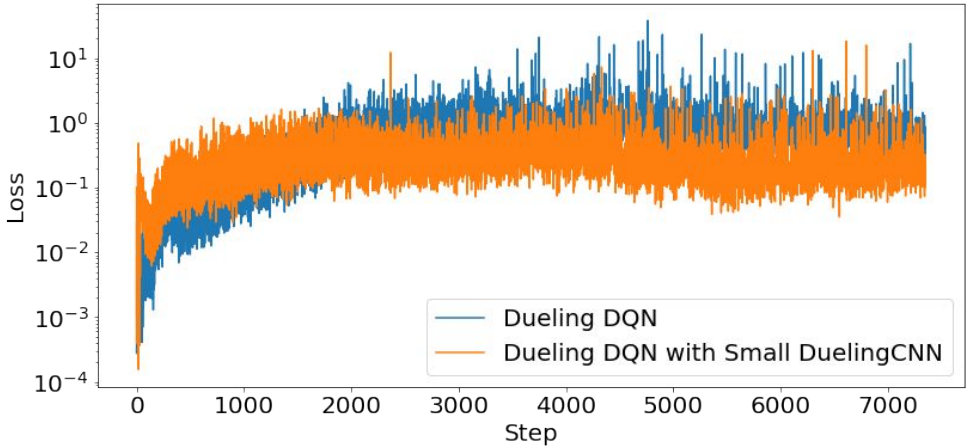
Stability of the algorithm vs. neural network size (Demon attack, Double DQN)



Average reward and loss for Dueling DQN for prolonged test

For the conducted number of steps there is no significant difference in average reward or loss





Conclusion

- For DQN and Double DQN target network is required
- Large size of action networks increases stability
- Target network in Dueling DQN may be excessive

Take a look at our WandB Report!