7)

- Good performance on test data doesn't necessarily correlate to good performance in practice. We observed that the predictions made on test data had very high accuracy. However, while using the model to predict the agent's action (which way the agent needs to move) in a new grid world, the model frequently ran out of the grid world or kept getting stuck in loops going left and right repeatedly. One possible explanation for this could be the reason that the model didn't understand the context of when to perform an action but simply performs it since it matches what it has already seen. Another possible explanation for this could be that there isn't enough data for the model to train on edge cases (when the agent is in (0,0), first row, first column, last row, last column, (dim,dim)), since most of the training data comprise of the agent being in the middle of the maze (as there are more cells in the middle of the maze compared to on the edges), with paths along with all 4 cardinal directions. This idea was only reinforced when it worsened while traversing through a higher dimensional matrix (meaning more central nodes and lesser edge cases).

- Simulating this in a new gridworld required hardcoding a lot of edge cases where the model either ran out of the maze or got stuck looping over and over again. The model certainly couldn't reach the goal node just based on the prediction it made, it needed someone looking over its actions so that it doesn't end up going out of bounds or continuously loop over and over.

8)

(Deduced from friend's report)

- The model took roughly 530 steps to get to the goal node when the agent did it with a Trajectory Length = 500. The standard deviation was around 20-25. It was just a traj length of agent vs model plot with a few points (if you wanted to add made-up data to the report).
- The ML agents were certainly a lot more time-intensive to train and as mentioned above missed the context of when to move where and constantly ran out of the grid and kept looping.