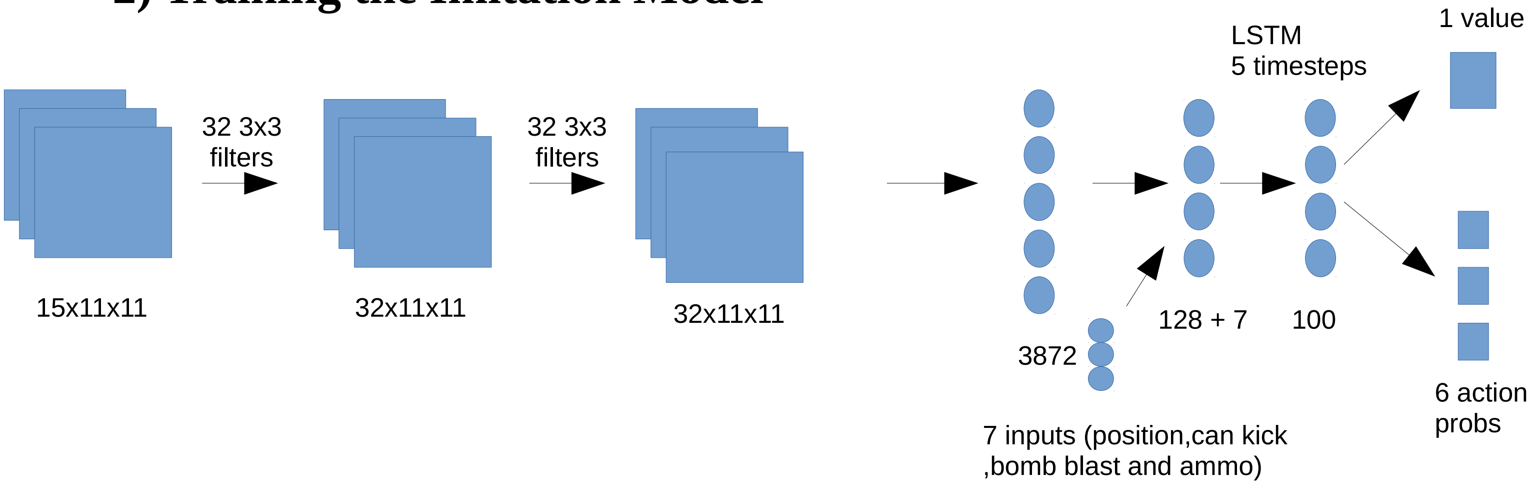
SAMARITAN - A Reinforcement Learning Agent for Pommerman

There are three steps :
1) Data Collection for Imitation Model
2) Training the Imitation Model
3) Training the Reinforcement Learning Model

1) Data Collection for Imitation Model

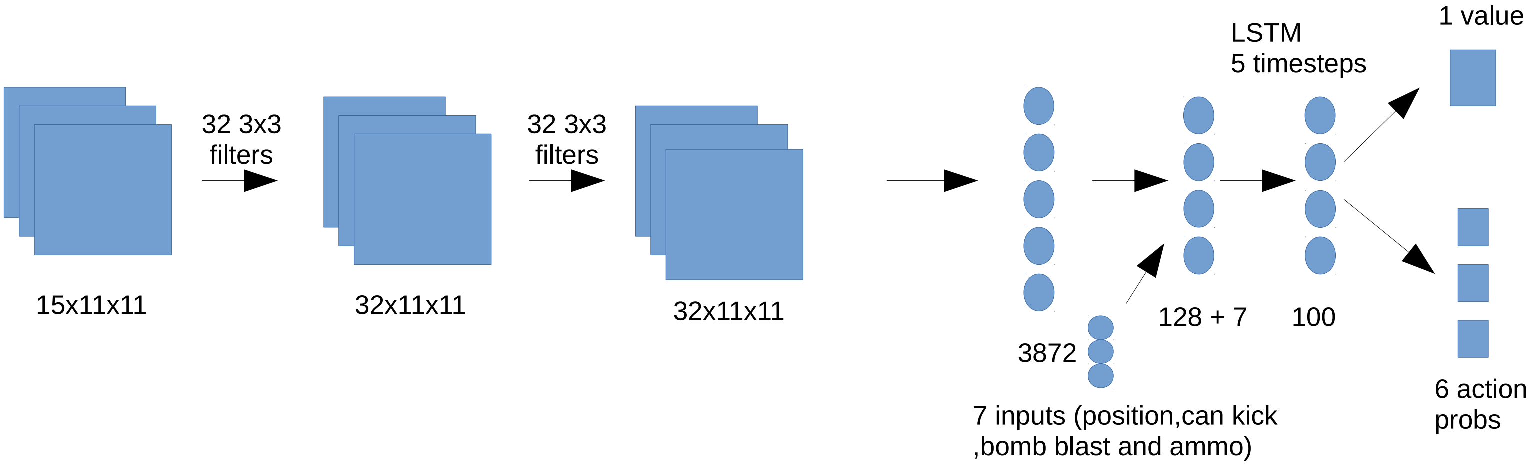
a) The environment is Pommerman Radio with 4 simple agents and we run 150 k episodes of this environment and from each episode we store one observation, reward and action of each agent from a random step in the episode so we would have a total of 600 k observations, rewards and actions. We also run another 15k episodes for test data.
b) We use a discount factor of 0.9 for the last reward of each of the agent.
c) We also design new middle rewards based on if we get power ups – kick, ammo and blast strength, put bombs and does not repeat a position in last 5 time steps.
d) The observations are pre-processed into 22 channels of 11x11 map with the first 15 channels working like a bitmap of different objects in the map and last 7 channels just having values of power ups, our position and team mates position in last step we got from message.

2) Training the Imitation Model



a) We use adam optimization with combination of action loss(cross entropy loss) and value loss (mean squared error loss) with epochs of 100 and batch size of 512.
b) In each epoch we will batch wise update the model for train and then run it on test.
c) If accuracy correct action taking in the state continuously reduces for three epochs in test data we stop

3) Training the Reinforcement Learning model



a) We use adam optimization with combination of action loss(cross entropy loss), value loss (mean squared error loss) and entropy loss with 10000 epochs
b) In every epoch we run 100 episodes of the pommerman game 10 of which is against team of static agents, 10 against random agents with no bombs, 30 against simple agents and 50 against super agents which are better than simple agents in the sense that they do not die by their own bombs from which we store preprocessed observations, actions, rewards, state-action probabilities and state values.
c)We randomly select 100 batches of stored data in each epoch.
d) The lstm consists of 5 timesteps which help in getting the middle rewards whereas discount factor is the one which helps in getting the final reward.