What you see in the agent’s behavior. Does it eventually make it to the target location?

The Basic Agent is an agent with completely random moves which does not obey the traffic rules. This agent, as expected, behaved erratically and rarely reached to its destination.

Justify why you picked these set of states, and how they model the agent and its environment.

For modeling an agent which can behave in a more intelligent manner, a set of states must be defined. Ideally, these states should contain all of the necessary information about the environment that is needed for choosing the correct action that eventually learn the environment and reaches its destination without acid end and on time. The states are as follow:

1. The state of the traffic light (either 'Green' or 'Red') - Green = NS open, Red = EW open
2. The state of oncoming traffic.
3. The state of traffic om the right.
4. The state of traffic om the left.
5. The direction recommended by the route planner
6. The deadline

The first four states give enough information to the smart cab to learn the traffic laws. The cab learns If the light is green or red, and if it is green, the cab can freely go straight through the intersection or make a right turn. However, the left turn on green depends on the state of the on-coming cars. If the light is red, the cab can make a right turn, if there are no cars on the left, on the right, and on-coming cars.

Adding the next\_waypoint state allows the smart cab to make progress towards the destination.

The deadline state allows the smartcab to learn to get to the destination on time.

For example, one of the states my agent can experience is ('green', None, None, 'forward', 24, 'forward').

What changes do you notice in the agent’s behavior?

After implementing a simple version of Q-Learning (and an action selection method that gradually becomes less explorative as the number of iterations approaches 100), the behavior of the cab greatly improved:

* It started to obey the traffic lights
* Followed the directions provided by the route planner

Report what changes you made to your basic implementation of Q-Learning to achieve the final version of the agent. How well does it perform?

Applied large number of parameter combinations for Q-learning. The parameters were alpha, gamma, and epsilon.  With initial value of 0.

hyper\_params={"init\_values":[0],"epsilons":[.1,.2,.4],"alphas":[.3,.7,.9],"gammas":[.4,.7,.9]}

The program generates 27 combination of parameters. Ran the program with this combination and achieved the best performance with the following parameters:

init\_value = 0, epsilon=0.2, alpha=0.9, gamma=0.4. The smartcab reached the destination with these parameters 80-90 times in 100 trials.

The full list of parameter results can be found in *Smartcab Perfrmance Report.csv* file*.*

Does your agent get close to finding an optimal policy, i.e. reach the destination in the minimum possible time, and not incur any penalties?

Yes, by adding the deadline state and adjusting the Q-learning parameters, it seems the agent gets closer to optimal policy in 100 trials.