Reinforcement Learning Assignment

Name: Dhev Mugunddhan A(21011101040)

Class: AI & DS – A

Problem: A news value maximiser: Politically and commercially affiliated media companies are tasked with maximizing the views for certain articles more than others. Build a system that maximises the views for these “aligned” articles.

Solution:-

Problem Statement: We are seeing a dozen of news articles everyday at various sources, be it online websites, or your browser feed, even in apps. And we could also see they have been curated personally and gives you the best experience you can get in their platform. But, to look on the other side, they specially pick articles and display in a way that has higher viewer count and highly engaging content. They have their own selection algorithms and strategies that curates to your interest and also increase more viewers to their target articles such that their revenues are increased. Now we are going to see how to implement a RL algorithm that does the job of selecting articles and strategies that are aligned with the media company’s interest.

Definitions and parameters:

* Environment: Real world with the viewers and the media company
* Agent: The media company who chooses a strategy that maximizes views for the aligned articles.
* Actions: The set of strategies or display order /position of articles in their page.
* Rewards: 2 \* viewer\_count of aligned\_articles + (-1\*) viewer\_count of non-aligned\_articles (For a dynamic environment which are modelled through probabilistic random variables)
* Here, each arm represents the actions to take
* objective is to maximize the expected total reward over some time period per se 1000 or 2000 time steps.
* each of the k actions has an expected or mean reward known as the value of that action which is q\*(a) = E[Rt | At =a] where We denote the action selected on time step t as At, and the corresponding reward as Rt.
* We denote the estimated value of action a at time step t as Qt(a). We would like Qt(a) to be close to q\*(a).
* The viewer count for each aligned\_articles and non-aligned articles for each action is modelled by two normal distributions for simplicity. We then use this to compute our regards.
* We follow the E-greedy strategy for our model to learn the best action.
* We would like to keep E as 0.2 to allow 20% percent of the time steps to exploration and the rest to exploitation. This is for initial start and we will tune it such that our model learns the estimate Qt(A) as close to q\*(a).
* For simplicity of implementation we take only 5 different actions to be chosen at any time-step

**Assumptions:**

* We assume that we do not know the action values with certainty, hence the true action value is hidden to the agent. So, we go for the estimated action value.
* We also assume the environment is independent of the agent’s actions.