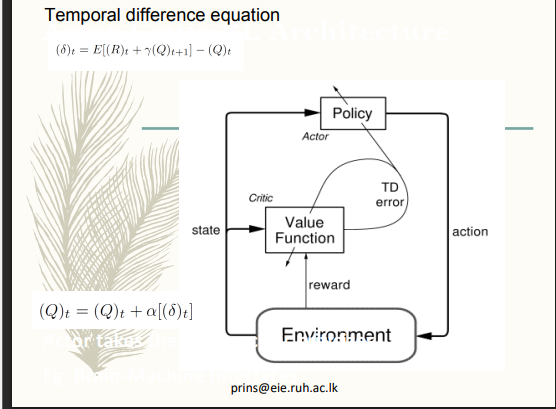


1. Actor-Critic Architecture (Bellman Expectation Equation)

* The **Actor-Critic** model is a type of RL algorithm that combines:
  + **Actor (Policy Improvement)**: Determines the best action to take in a given state.
  + **Critic (Policy Evaluation)**: Evaluates the taken action and provides feedback.
* The **Bellman expectation equation** is used to estimate the value of a state based on expected rewards and future state values: Q(st)=E[Rt+γQ(st+1)]Q(s\_t) = E[R\_t + \gamma Q(s\_{t+1})]Q(st​)=E[Rt​+γQ(st+1​)] where:
  + Q(st)Q(s\_t)Q(st​) is the expected return from state sts\_tst​.
  + RtR\_tRt​ is the immediate reward.
  + γ\gammaγ is the discount factor for future rewards.
  + st+1s\_{t+1}st+1​ is the next state.



**Temporal Difference (TD) Learning**

* The **Temporal Difference (TD) equation** is a fundamental concept in RL that updates the value of a state iteratively: (Q)t=(Q)t+α[δt](Q)\_t = (Q)\_{t} + \alpha [\delta\_t](Q)t​=(Q)t​+α[δt​] where:
  + α\alphaα is the learning rate.
  + δt\delta\_tδt​ is the **TD error**, which represents the difference between expected and received rewards.
* The diagram shows that the **policy (Actor) updates the actions**, while the **value function (Critic) updates the TD error**.

