Reinforcement Learning For Adaptive Interview Coaching

Author
Debargha Nath (22CS01070)

Table of content

1. System Definition: Agent and Environment

1.1. State Space Definition (S)

Instead of just qualitative labels, we quantify the student's state at any day t.

The State Vector St is defined as:

$S_t = \{M_t, C_t, B_t, T_t\}$

- **Mt (Mastery):** Current skill level (0 to 100).
- **Ct (Confidence):** Self-belief metric (0 to 100).
- **Bt (Burnout):** Fatigue/Stress level (0 to 100).
- **Tt (Time):** Days remaining (30-t).

1.2. Action Space Definition (A)

We refine the actions to have specific, probabilistic intents. Let's define the set of actions A:

Detailed Explanation of each action's effect on the state of a student is given later.

1. **A1: Increase Difficulty**
    - *Intent:* High risk, high reward. Attempts to spike Mastery.
2. **A2: Offer Quick Revision**
    - *Intent:* Stabilize Confidence, maintain Mastery, lower Burnout slightly.
3. **A3: Give Encouragement**

    - *Intent:* purely psychological. Boosts Confidence, drastically lowers Burnout.
4. **A4: Switch Topic**
    - *Intent:* Break monotony. Prevents Burnout spikes without stopping learning.

2. Agent: The Placement Preparation Guide

2.1. Core Functionality

The Agent acts as an adaptive supervisor for a student preparing for placements. Unlike a static timetable, it operates as a **dynamic control system** over a finite horizon (30 days). It continuously monitors the student's psychological and academic state to optimize the daily learning strategy.

2.2. The Objective

The Agent aims to simultaneously maximize growth and minimize risk:

- **Maximize Mastery:** Push technical/aptitude skills to the highest possible limit.
- **Maximize Confidence:** Ensure the student believes in their ability to perform.
- **Minimize Burnout:** Prevent fatigue-induced crashes or quitting.
- Achieve all this goal in minimum time. As only 30 days are available for preparation

2.3. State Space

The Agent observes the following variables at the start of each day to make decisions. Note the specific scales defined:

Parameter	Range	Description
Mastery (Mt)	**0 – 20**	Current technical/aptitude preparedness.

Parameter	Range	Description
Confidence (Ct)	**0 – 20**	Self-belief regarding interview/test performance.
Burnout Probability (Bt)	**0.0 – 1.0**	Likelihood of mental exhaustion (0% to 100%).

2.4. Action Space

The Agent chooses one of four distinct strategies listed in section 1.2. daily:

2.5. Interaction Loop

- **Observe:** Agent reads the student's current Mastery, Confidence, and Burnout.
- **Act:** Agent selects the best action (A1–A4) for that specific day.
- **Update:** The Agent's representation of the Student (Environment) reacts to the action, resulting in new State values for the next day.

3. Student Learning Environment Model

3.1. The Environment / Student

The **student** represents the learning environment in which the agent operates.
The student's **state** is defined by three evolving attributes:

Attribute	Description
Mastery	Reflects current technical understanding and preparedness level.

Confidence	Indicates self-belief and readiness to face placement challenges.
Burnout Probability	Measures emotional strain, fatigue, or risk of losing motivation.

As the student progresses through the 30-day plan, these values change based on the recommended actions, forming a **feedback loop** that the agent learns from.

3.2. Action Impact Summary

Each action affects the student's state differently.

Action	Name	Primary Effects on Student State
0	**Increase Difficulty**	• Success: Mastery (+2), Confidence (+2), Burnout (+0.1) • Failure: Confidence (-2), Burnout (+0.15)
1	**Switch Topic**	• Burnout (-0.2), Confidence (+1)
2	**Give Encouragement**	• Burnout (-0.4), Confidence (+3)
3	**Offer Quick Revision**	• Mastery (+1), Burnout (-0.1)

3.3. Reward System

The reward evaluates how effective each action is in improving the student's state.

Positive Contributions

Component	Description	Impact on Reward

Mastery Gain	Increase in technical understanding	+5.0 × mastery_gain
Confidence Gain	Growth in self-belief	+1.5 × confidence_gain
Burnout Reduction	Lower stress or fatigue	+25.0 × burn_reduction

Special Conditions

Condition	Action Effect	Reward Outcome
Burnout high (> 0.7) and action = Encouragement (2)	Smart recovery step	+50.0 bonus
Burnout high (> 0.7) but action ≠ 2	Poor decision	-20.0 penalty
Burnout becomes extreme (> 0.9)	Risk of exhaustion	-30.0 penalty
Burnout reaches 1.0 or more	Critical failure	-1000.0 penalty

Time-Based Penalty

A small penalty discourages unnecessary actions:

reward -= 0.05 times(*) steps_used

End-of-Plan Bonus

Final Mastery	Outcome
≥ 16	+50.0 bonus

< 10	-50.0 penalty

4. Algorithm Used: Q Learning

4.1. Q-Learning Algorithm

Q-Learning is an **off-policy, model-free** temporal difference (TD) control algorithm.

- **Q-Table (Q):** This is the core data structure (self.q_table). It stores the learned **Q-values** (or quality values) for state-action pairs. Q(s,a) represents the expected cumulative discounted future reward for taking action a in state s.
- **Action Selection (ϵ-Greedy Policy):** The choose_action method uses the ϵ-greedy strategy for balancing **exploration** and **exploitation**.
  - With probability ϵ (self.epsilon), the agent **explores** by choosing a random action (np.random.randint).
  - With probability 1-ϵ, the agent **exploits** by choosing the action with the maximum Q-value for the current state (the greedy action: argmaxaQ(A,A)).
- **The Learning Rule (Q-Value Update):** The learn method implements the Q-Learning update rule, which is based on the **Bellman Equation** for optimality.
  The update is calculated as:

  $$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma a' \max Q(s',a') - Q(s,a)]$$

  where:

  - s: Current state (state)
  - a: Action taken (action)
  - r: Immediate reward (reward)

- s′: Next state (next_state)
- Q(s,a): Current Q-value (the value being updated)
- maxa′Q(s′,a′): The maximum expected future reward in the next state, s′ (used because Q-Learning is off-policy). This is the **target Q-value** from the best possible action in the next state.
- α: **Learning Rate** (self.alpha). It determines how much of the new information overrides the old information.
- γ: **Discount Factor** (self.gamma). It determines the importance of future rewards. A value closer to 1 makes the agent more long-sighted.
- [r+γmaxa′Q(s′,a′)−Q(s,a)]: The **Temporal Difference (TD) Error**, which is the difference between the target (estimated value of s,a) and the current value (Q(s,a)).

4.2. Hyperparameters and Auxiliary Techniques

- **Learning Rate (α):** self.alpha = 0.1 is the parameter controlling the magnitude of updates to the Q-values.
- **Discount Factor (γ):** self.gamma = 0.9 is the parameter that discounts future rewards.
- **Epsilon Decay:** The decay_epsilon method gradually reduces the value of ϵ (self.epsilon) over time using a decay factor (self.decay). This is crucial because it allows the agent to start with high **exploration** (high ϵ) and slowly transition to high **exploitation** (low ϵ) as it gains confidence in its learned Q-values.
- **Min Epsilon:** self.min_epsilon ensures that ϵ never drops completely to zero, guaranteeing that the agent continues to explore occasionally, preventing it from getting stuck in a suboptimal policy.
- **State Initialization:** The _ensure_state method handles unseen states by initializing their Q-values to zero, ensuring that every visited state is present in the q_table.

5. Example Cases

5.1 Case: Low Confidence, High Mastery, Low Burnout:

Input:

```
AI Coach ready!
Confidence (0–20): 5
Mastery (0–20): 10
Burnout (0.0–1.0): 0.2
```

Output:

```
Day | Action | Conf | Mastery | Burnout
-------------------------------------
1   | SWITCH_TOPIC          | 6    | 10      | 0.00
2   | INCREASE_DIFFICULTY   | 8    | 12      | 0.10
3   | INCREASE_DIFFICULTY   | 6    | 12      | 0.25
4   | SWITCH_TOPIC          | 7    | 12      | 0.05
5   | SWITCH_TOPIC          | 8    | 12      | 0.00
6   | SWITCH_TOPIC          | 9    | 12      | 0.00
7   | SWITCH_TOPIC          | 10   | 12      | 0.00
8   | SWITCH_TOPIC          | 11   | 12      | 0.00
9   | INCREASE_DIFFICULTY   | 9    | 12      | 0.15
10  | INCREASE_DIFFICULTY   | 11   | 14      | 0.25
11  | INCREASE_DIFFICULTY   | 13   | 16      | 0.35
12  | INCREASE_DIFFICULTY   | 15   | 18      | 0.45
13  | INCREASE_DIFFICULTY   | 17   | 20      | 0.55
14  | INCREASE_DIFFICULTY   | 15   | 20      | 0.70
15  | INCREASE_DIFFICULTY   | 17   | 20      | 0.80
16  | SWITCH_TOPIC          | 18   | 20      | 0.60
17  | INCREASE_DIFFICULTY   | 20   | 20      | 0.70

Student reached maximum mastery and confidence!
```

5.2. Case: Low Confidence, High Mastery, High Burnout

Input:

```
AI Coach ready!
Confidence (0–20): 2
Mastery (0–20): 10
Burnout (0.0–1.0): 0.7
```

Output:

```
Day | Action | Conf | Mastery | Burnout
-------------------------------------------
1   | SWITCH_TOPIC          | 3    | 10      | 0.50
2   | SWITCH_TOPIC          | 4    | 10      | 0.30
3   | SWITCH_TOPIC          | 5    | 10      | 0.10
4   | INCREASE_DIFFICULTY   | 3    | 10      | 0.25
5   | INCREASE_DIFFICULTY   | 1    | 10      | 0.40
6   | INCREASE_DIFFICULTY   | 3    | 12      | 0.50
7   | INCREASE_DIFFICULTY   | 5    | 14      | 0.60
8   | INCREASE_DIFFICULTY   | 3    | 14      | 0.75
9   | SWITCH_TOPIC          | 4    | 14      | 0.55
10  | INCREASE_DIFFICULTY   | 6    | 16      | 0.65
11  | INCREASE_DIFFICULTY   | 8    | 18      | 0.75
12  | SWITCH_TOPIC          | 9    | 18      | 0.55
13  | INCREASE_DIFFICULTY   | 11   | 20      | 0.65
14  | INCREASE_DIFFICULTY   | 13   | 20      | 0.75
15  | SWITCH_TOPIC          | 14   | 20      | 0.55
16  | INCREASE_DIFFICULTY   | 16   | 20      | 0.65
17  | INCREASE_DIFFICULTY   | 18   | 20      | 0.75
18  | INCREASE_DIFFICULTY   | 20   | 20      | 0.85

Student reached maximum mastery and confidence!
```

5.3. Case: Low Confidence, Low Mastery, High Burnout

Input:

```
AI Coach ready!
Confidence (0–20): 5
Mastery (0–20): 5
Burnout (0.0–1.0): 0.9
```

Output:

```
Day | Action | Conf | Mastery | Burnout
------------------------------------------
1   | SWITCH_TOPIC          | 6    | 5       | 0.70
2   | INCREASE_DIFFICULTY   | 8    | 7       | 0.80
3   | SWITCH_TOPIC          | 9    | 7       | 0.60
4   | INCREASE_DIFFICULTY   | 7    | 7       | 0.75
5   | INCREASE_DIFFICULTY   | 9    | 9       | 0.85
6   | INCREASE_DIFFICULTY   | 11   | 11      | 0.95
7   | INCREASE_DIFFICULTY   | 13   | 13      | 1.00
8   | GIVE_ENCOURAGEMENT    | 16   | 13      | 0.60
9   | SWITCH_TOPIC          | 17   | 13      | 0.40
10  | SWITCH_TOPIC          | 18   | 13      | 0.20
11  | SWITCH_TOPIC          | 19   | 13      | 0.00
12  | INCREASE_DIFFICULTY   | 20   | 15      | 0.10
13  | OFFER_QUICK_REVISION  | 20   | 16      | 0.00
14  | SWITCH_TOPIC          | 20   | 16      | 0.00
15  | INCREASE_DIFFICULTY   | 20   | 18      | 0.10
16  | INCREASE_DIFFICULTY   | 18   | 18      | 0.25
17  | INCREASE_DIFFICULTY   | 20   | 20      | 0.35

Student reached maximum mastery and confidence!
```

Comparison Between The Output of the Agent and fixed strategy

6. Comparison: Q-Learning Adaptive Strategy vs Fixed Schedule

6.1 Observations

Adaptive Strategy:

Day	Action	Conf	Mastery	Burnout
1	SWITCH_TOPIC	6	5	0.70
2	INCREASE_DIFFICULTY	8	7	0.80
3	SWITCH_TOPIC	9	7	0.60
4	INCREASE_DIFFICULTY	7	7	0.75
5	INCREASE_DIFFICULTY	9	9	0.85
6	INCREASE_DIFFICULTY	11	11	0.95
7	INCREASE_DIFFICULTY	13	13	1.00
8	GIVE_ENCOURAGEMENT	16	13	0.60
9	SWITCH_TOPIC	17	13	0.40
10	SWITCH_TOPIC	18	13	0.20
11	SWITCH_TOPIC	19	13	0.00
12	INCREASE_DIFFICULTY	20	15	0.10
13	OFFER_QUICK_REVISION	20	16	0.00
14	SWITCH_TOPIC	20	16	0.00
15	INCREASE_DIFFICULTY	20	18	0.10
16	INCREASE_DIFFICULTY	18	18	0.25
17	INCREASE_DIFFICULTY	20	20	0.35

Student reached maximum mastery and confidence!

Fixed Strategy Always Increase Difficulty:

Day	Action	Conf	Mastery	Burnout
1	Increase Difficulty	5	5	1.00
2	Increase Difficulty	5	5	1.00
3	Increase Difficulty	5	5	1.00
4	Increase Difficulty	5	5	1.00
5	Increase Difficulty	5	5	1.00
6	Increase Difficulty	5	5	1.00
7	Increase Difficulty	5	5	1.00
8	Increase Difficulty	5	5	1.00
9	Increase Difficulty	5	5	1.00
10	Increase Difficulty	5	5	1.00
11	Increase Difficulty	5	5	1.00
12	Increase Difficulty	5	5	1.00
13	Increase Difficulty	5	5	1.00
14	Increase Difficulty	5	5	1.00
15	Increase Difficulty	5	5	1.00
16	Increase Difficulty	5	5	1.00
17	Increase Difficulty	5	5	1.00
18	Increase Difficulty	5	5	1.00
19	Increase Difficulty	5	5	1.00
20	Increase Difficulty	5	5	1.00
21	Increase Difficulty	5	5	1.00
22	Increase Difficulty	5	5	1.00
23	Increase Difficulty	5	5	1.00
24	Increase Difficulty	5	5	1.00
25	Increase Difficulty	5	5	1.00
26	Increase Difficulty	5	5	1.00
27	Increase Difficulty	5	5	1.00
28	Increase Difficulty	5	5	1.00
29	Increase Difficulty	5	5	1.00
30	Increase Difficulty	5	5	1.00

6.2 Results and Conclusions:

6.2.1. Adaptive Strategy (Q-Learning):

- Dynamically chooses actions based on student state (Confidence, Mastery, Burnout).
- Encouragement is used when burnout is high to prevent critical failure.
- Results in smoother confidence growth and controlled burnout.

6.2.2. Fixed Schedule:

- Repeats a static timeline regardless of student state.
- May push burnout too high or fail to improve mastery efficiently.

6.2.3. Insights from Comparison:

- Q-Learning produces **higher total reward** by balancing mastery and confidence while avoiding burnout.
- Fixed schedules are vulnerable of pushing the student too much and making him burnout quickly