Paper Critique

Shuvrajeet Das, DA24D402

Course: DA7400, Fall 2024, IITM
Paper: [Teacher-Student Curriculum Learning]
Date: [25-10-24]

Make sure your critique Address the following points:
1. The problem the paper is trying to address
2. Key contributions of the paper
3. Proposed algorithm/framework
4. How the proposed algorithm addressed the described problem
Note: Be concise with your explanations. Unnecessary verbosity will be penalized. Please don't exceed 2 pages.

---

1 The problem the paper is trying to address

The problem is to optimize curriculum learning by automatically selecting tasks for the **Student** model to learn, while minimizing training time and preventing forgetting. This is framed as a **Partially Observable Markov Decision Process (POMDP)** where a **Teacher** selects subtasks based on the Student's learning progress.

The goal is to maximize the cumulative reward and the Teacher must balance between **exploitation** (continuing on tasks with good progress) and **exploration** (selecting new tasks or preventing forgetting).

2 Key contributions of the paper

The key contributions of the paper are:

- Formalizing the **Teacher-Student Curriculum Learning (TSCL)** framework as a *Partially Observable Markov Decision Process (POMDP)*.
- Proposing a family of **Teacher algorithms** based on the concept of *learning progress*:

$$r_t = o_t - o_{t'}$$

- Addressing the *problem of forgetting* by prioritizing tasks where the Student's performance is declining.
- Demonstrating that **TSCL** matches or surpasses manually designed curricula in both *supervised learning* (e.g. decimal addition with LSTM) and *reinforcement learning* (e.g. Minecraft navigation).

3 Proposed algorithm/framework

The paper proposes the **Teacher-Student Curriculum Learning (TSCL)** framework, modeled as a *Partially Observable Markov Decision Process (POMDP)*. The framework is defined as follows:

- **State**:

$$s_t = \text{Student's learning state at time } t \quad \text{(neural network parameters, optimizer state)}$$

- **Action**: a_t = Teacher's choice of subtask for the Student at time t
- **Observation**: o_t = score (reward) obtained after Student trains on subtask a_t
- **Reward**: $r_t = o_t - o_{t'}$

Teacher's Policy: The Teacher selects tasks where the Student is making the most progress: To address *forgetting*, the Teacher also selects tasks where the Student's performance is worsening: Several **heuristic algorithms** are proposed for task selection, including:

- **Online Algorithm**: Exponentially weighted moving average to track expected reward:

$$Q_{t+1}(a_t) = \alpha r_t + (1 - \alpha) Q_t(a_t)$$

- **Naive Algorithm**: Linear regression over a fixed number of trials to estimate the slope of the learning curve.
- **Window Algorithm**: FIFO buffer of the last K scores and linear regression to estimate learning progress.
- **Sampling Algorithm**: Thompson sampling to naturally balance exploration and exploitation.

4 How the proposed algorithm addressed the problem

The proposed algorithm addresses the problem by leveraging the **Teacher-Student Curriculum Learning (TSCL)** framework to optimize the selection of training subtasks, which solves two major challenges:

1. **Efficient Task Selection**: The Teacher selects tasks based on *learning progress*, i.e., tasks where the Student is making the most improvement: This ensures that the Student focuses on tasks where learning is most effective, which speeds up training.
2. **Forgetting Prevention**: To counter forgetting, the Teacher selects tasks where the Student's performance is declining: This prevents the model from forgetting previously learned tasks and ensures consistent performance across all subtasks.

The algorithm uses various **heuristics** to efficiently estimate the learning progress and balance exploration and exploitation, such as:

- **Online Algorithm**: Tracks the expected reward $Q_t(a)$ using exponentially weighted averages:

$$Q_{t+1}(a_t) = \alpha r_t + (1 - \alpha) Q_t(a_t)$$

- **Naive and Window Algorithms**: Estimate the slope of the learning curve using linear regression, helping the Teacher identify where progress is fastest or where forgetting is occurring.
- **Sampling Algorithm**: Uses Thompson sampling to naturally balance between exploitation (tasks with high learning progress) and exploration (tasks with uncertain outcomes).