Adaptive Hybrid Greedy and Reinforcement Learning for Distributed Submodular Maximization (AHRL-Greedy)

Presenter: William Zhong
Date: 26 Nov 2024

Introduction

- Problem Statement:
  - Selecting a representative subset from massive datasets is crucial in large-scale machine learning tasks like clustering and kernel methods.
  - This selection can be modeled as maximizing a submodular objective function.

- Challenges:
  - Traditional centralized approaches are impractical for large-scale problems due to the need for centralized data access.
  - Existing methods like GREEDI, while computationally efficient, may not always achieve optimal solutions.

Motivation

- Why It's Important:
  - Submodular maximization has applications in machine learning, data mining, and sensor networks.
  - Scalable algorithms are essential for handling growing and distributed data volumes.

- Why It's Challenging:
  - Submodular maximization is typically NP-hard, especially with constraints.
  - Distributed settings add complexity due to coordination and communication challenges among nodes.

Related Work

- Greedy Algorithms:
  - Foundational in submodular optimization, offering simple yet effective approximations.
- Parallel Greedy Algorithms:
  - Extend greedy approaches to distributed settings, allowing simultaneous selections by multiple nodes.
- Randomized Greedy Algorithms:
  - Introduce randomness to improve solution diversity and explore a larger solution space.
- Hybrid Approaches:
  - Combine greedy algorithms with learning mechanisms to address the limitations of purely greedy methods.
- Reinforcement Learning (RL):
  - Applied to combinatorial optimization problems, enabling adaptive and learned decision-making.

Problem Formulation

- Objective:
  - Maximize a submodular function $f: 2^v \rightarrow R_{\geq 0}$ over a dataset V, subject to a cardinality constraint k.

- Distributed Environment:
  - Data is partitioned across N nodes, each holding a subset $V_i \subseteq V$
  - Each node selects a local subset $S_i \subseteq V_i$ aiming to contribute to a global solution S^*.

Proposed Method: AHRL-Greedy

- Hybrid Greedy-RL Selection:
  - Each node operates as an RL agent, selecting elements based on local marginal gains and learned Q-values.
  - The selection criterion is $v_i^* = argmax_{v \in V_i}(\Delta f(S_i, v) + Q_i(S_i, v))$.

- Reward Function:
  - Combines immediate submodular gain, diversity, and coverage:
    - $r_i(S, v) = \Delta f(S, v) + \lambda \cdot diversity(S, v) + \mu \cdot coverage(S, v)$
    - $\Delta f(S, v)$: Marginal gain from adding v to S.
    - λ,μ: Hyperparameters balancing diversity and coverage

Experience Replay and Learning

- Experience Replay Buffer:
  - Each node maintains a buffer BiB_iBi storing past experiences (S,v,r,S').

- Q-Value Updates:
  - Parameters θi\theta_iθi are updated using temporal difference learning:
  - $\theta_i \leftarrow \theta_i + \alpha(r_i + \gamma max_{v'} Q_i(S', v') - Q_i(S, v))\nabla\theta_i Q_i(S, v)$
  - α: Learning rate.
  - γ: Discount factor.

Consensus Mechanism

- After local selections, a consensus mechanism combines $S_1, S_2, \ldots, S_N$ to form the global solution S^* .

- Ensures representativeness and diversity across the entire dataset.

Algorithm Overview

Algorithm 1: AHRL-Greedy Algorithm

Input: Set V, number of nodes N, cardinality constraint k, learning rate α, discount factor γ.

Output: Final global solution S^*.

Initialize local subsets $S_i = \emptyset$ for each node $i \in \mathcal{N}$;

Initialize Q-value parameters θ_i and experience replay buffer $\mathcal{B}_i$ for each node;

for *each time step* $t = 1, 2, \ldots, T$ **do**

  for *each node* $i \in \mathcal{N}$ *in parallel* **do**

    Select element $v_i^* = \arg\max_{v \in V_i} \left(\Delta f(S_i, v) + Q_i(S_i, v) \right)$;

    Add element to local subset: $S_i \leftarrow S_i \cup \{v_i^*\}$;

    Store experience (S_i, v_i^*, r_i, S_i') in $\mathcal{B}_i$;

    Update Q-value parameters θ_i using experience replay;

Apply consensus mechanism to generate final global solution $S^* = \text{consensus}(S_1, S_2, \ldots, S_N)$;

Experiments

- Objective:
  - Evaluate the performance of AHRL-Greedy compared to existing methods like GREEDI.
- Datasets:
  - Utilized standard large-scale datasets commonly used in submodular maximization tasks.
- Metrics:
  - Measured solution quality based on the value of the submodular objective function achieved.
  - Assessed computational efficiency in terms of runtime and scalability across distributed nodes.
- Results:
  - AHRL-Greedy consistently matched or outperformed GREEDI in solution quality.
  - Demonstrated efficient scalability with increasing data sizes and number of nodes.

Experiments (Con’t)

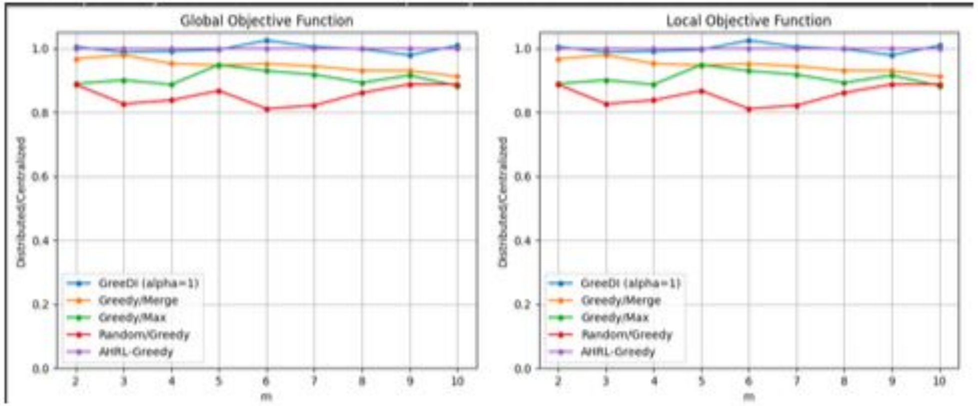

Figure 1: Experiment of AHRL-Greedy and comparison with different method.

Conclusion

- Summary:
  - Introduced AHRL-Greedy, a novel algorithm combining greedy heuristics with reinforcement learning for distributed submodular maximization.
  - Addressed challenges of scalability and coordination in distributed environments.
- Key Takeaways:
  - AHRL-Greedy effectively balances immediate gains with long-term benefits like diversity and coverage.
  - The integration of reinforcement learning enables adaptive decision-making in dynamic data scenarios.
- Future Work:
  - Explore enhancements to the consensus mechanism for improved global solution quality.
  - Investigate the application of AHRL-Greedy to other combinatorial optimization problems beyond submodular maximization.