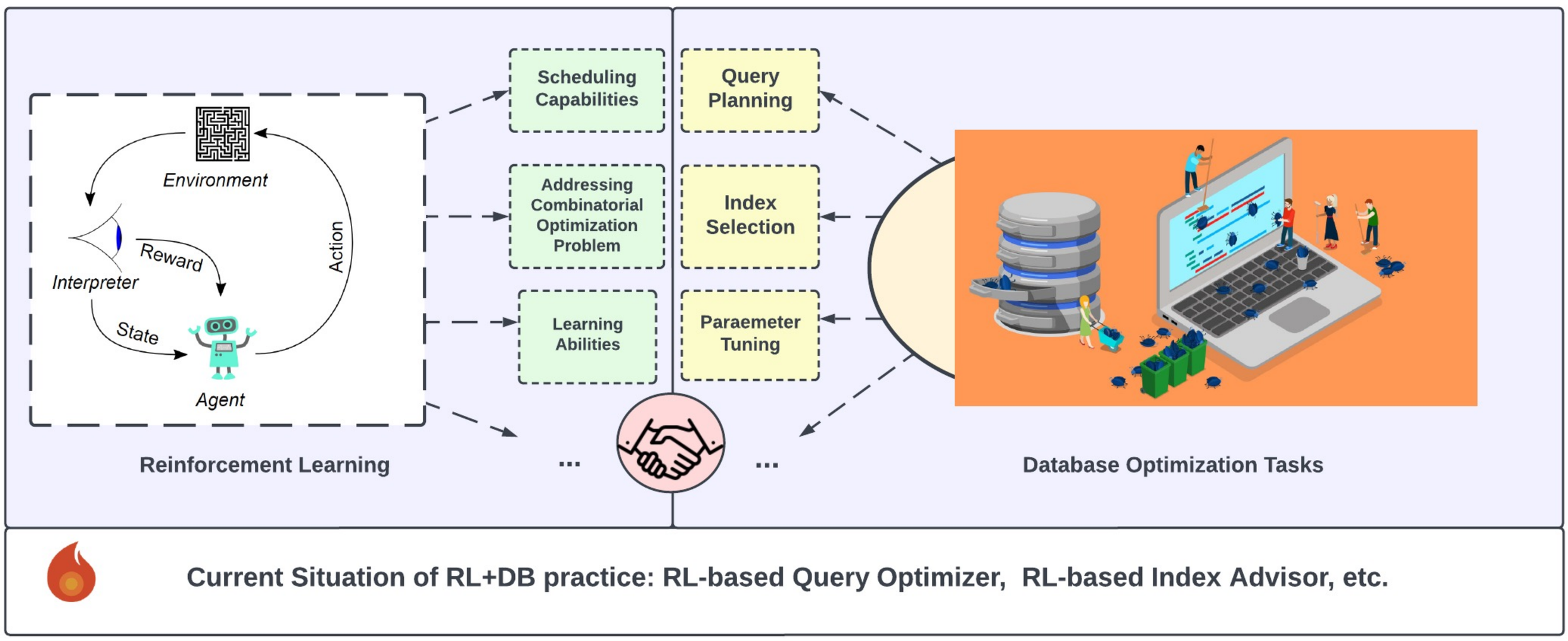
Enhancing Generalization through Task Vector Fusion in Deep Reinforcement Learning for Database Optimization

Taiyi Wang, Eiko Yoneki

Department of Computer Science and Technology, University of Cambridge

(E-mail: Taiyi.Wang@cl.cam.ac.uk)

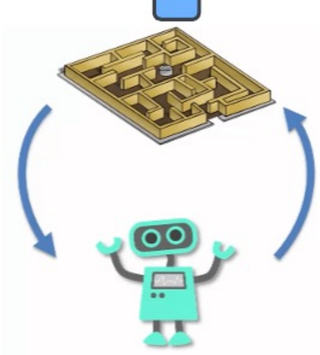


Current Situation of RL+DB practice: RL-based Query Optimizer, RL-based Index Advisor, etc.

Fast Evolving Environment

- Frequent Tuning Needs
- Changing Workloads
- Changing Data Distributions

Dilemma of RL practice for DB



RL agents are usaully trained within Static Environment, e.g. fixed workloads, static data, through repeated decisions and training steps

Current Solutions

Smart training and pre-training strategies

1. Meta Training
2. Pre-trained + Fine Tuning

Hard to Control and Monitor, too sophiscated,e.g. how to tell the training is sufficient and won't be overfitted?

Representative States

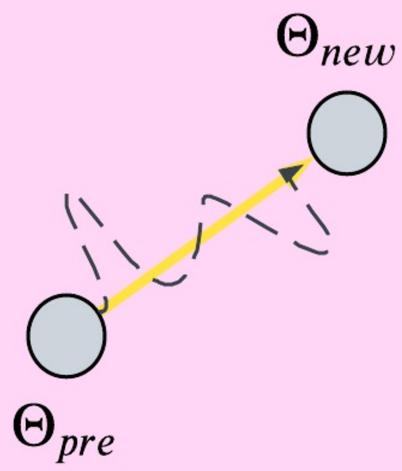
Comprehensive Encoded Workload Model

Downstreaming Task- Specific Designs are needed. Low sample efficiency on complex states representations.

How can we leverage pre-existing trained models to streamline and enhance task performance on unseen workloads?

Task Vectors

Learned Models ≈ Vectors of NN Parameters



Task Vector is defined by

$$V = \Theta_{new} - \Theta_{pre}$$

Proposed Solutions

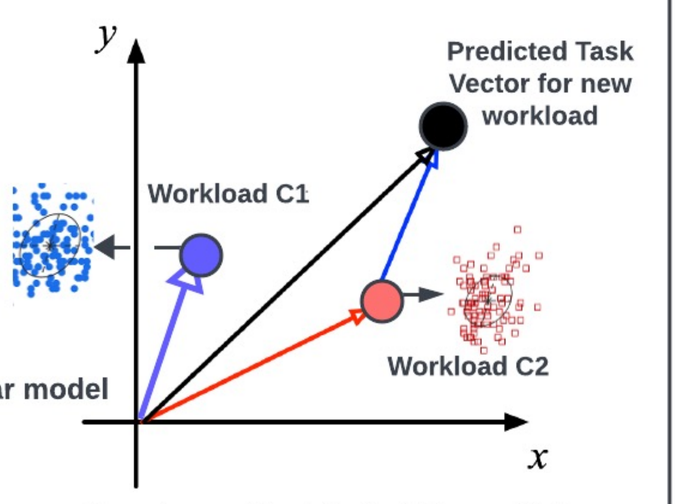
Task Vector Fusion from known workloads creates the most cost-effective predictions for unseen workloads.

$$\min_f L\left(\Theta_{\text{original}} + f(\mathbf{V}_{\text{pre}})\right)$$

where:

- L is the loss/Evaluation function.
- Θ_{original} are the original neural network parameters.
- f is the mapping function from the set of pre-trained task vectors $\mathbf{V}_{\text{pre}} = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\}$ to the predicted task vector.
- The goal is to find the optimal f that minimizes the loss L.

e.g. Fusion Model = Linear model



- **Clustering workloads by Task Vector affinity.**
- **Employ Central Vectors for Cluster Representation.**
- **Assuming pre-trained models exist for Workload Cluster 1 and Workload Cluster 2.**

$$\overrightarrow{V_{pred}} = \overrightarrow{V_{W_{cluster1}}} + \overrightarrow{V_{W_{cluster2}}}$$

Challenges

A. Pre-trained Task Vector Sets:
Acquiring a comprehensive and diverse collection of pre-trained task vectors that adequately represent the spectrum of potential database workloads and scenarios.

B. Fusion Model Evaluation:
Establishing robust metrics and methodologies to assess the effectiveness and efficiency of the fused models in accurately predicting and adapting to new tasks.