

Rlgraph: Modular Computation Graphs For Deep Reinforcement Learning

Paper by Michael Schaarschmidt, Sven Mika, Kai Fricke, Eiko Yoneki

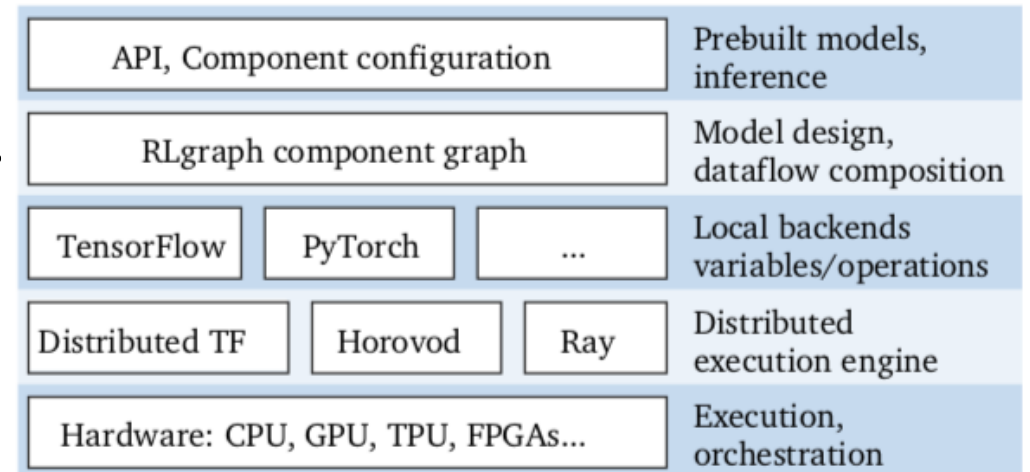
Proposal by Matthew Guest

Rlgraph Overview

- This paper outlines a new, unifying Framework with the aim to improve:
- Incremental Building Testing
 - To improve the speed of prototyping and robustness of production models.
- Extensibility
 - By separating “logical component composition, backend graph definition and distributed execution,” components are interchangeable and well defined.

RLgraph Components

- The RLgraph framework is primarily a Component graph.
- A Component class can encapsulate arbitrary computations.
- A Component contains internal methods, API methods, variables, and associated sub-components.
- This graph structure is an abstraction that can be executed across platforms.



Core Proposal

- Demonstrate Core Contributions of Rlgraph:
 - Incremental testing
 - Modular (interchangeable) components
 - Execution environment independence
 - Robustness to forgetfulness
- Implementing simple RL model for some common task using Rlgraph:
 - Interchanging components
 - Changing execution environments

Possible exploration

- Do Rlgraph computation graphs have exploitable structure?
- Answer could be no...
- If yes:
 - REGAL: computation graph compiler construction?
 - Relaxed Graph Substitution

Project Questions

- Motivation
- Plan

Thank you for listening

Q&A