Assessing RLGraph prototyping capability

Indigo Orton
RLGraph

- Framework for building, executing, and testing reinforcement learning models

- Better than RLlib
Motivation - Reinforcement learning standardization

• RL tasks are challenging to implement, execute, and test
• Less support than other forms of deep learning
• Lack of standardization leads to customization
• Customization increases barrier for new work
• Existing frameworks operate on distinct aspects of RL
RLGraph

• Common framework for RL tasks
• Separation of concerns
  • Logical component composition
  • Backend graph definition
  • Distributed execution
• Support for multiple backends (e.g. TensorFlow, PyTorch)
• Efficient for prototyping
• Ease of prototyping agent
• Accessibility and applicability for non-RL research and industry
• Comparison with existing RL agents
Evaluation - 2048

- Well defined
- Easy to simulate and execute actions
- Clear reward signal
- Reasonable size for training interesting network

Image from [2]
Conclusion

- Evaluating RLGraph
- Prototyping capability and performance
- Use in non-RL focused domains
- 2048 target
- Progress – negligible