PROJECT PROPOSAL: COMPILER OPTIMIZATIONS USING REINFORCEMENT LEARNING

Akshat Garg
akshat061200@gmail.com

1 INTRODUCTION

The project deals with the application of reinforcement learning to compiler optimizations for CompilerGym

2 OBJECTIVE

Using Reinforcement learning to do compiler optimization with the help of Vowpalwabbit agents. Experimenting with several compiler optimization tasks.

3 PROBLEM DESCRIPTION

The project deals with the optimization of compilers whether in performance ,storage size or power consumption etc in every possible way using reinforcement learning techniques. We will use VowpalWabbit to create RL agents and experiment with various algorithms and optimization challenges. Compiler optimization is very much important as we need the best results in least amount of time.

4 METHODOLOGY

This problem can be solved using reinforcement learning. We will use VowpalWabbit as a external library to solve the compiler optimization problem.VowpalWabbit provides fast, efficient, and flexible online machine learning techniques for reinforcement learning.

5 PROJECT SCOPE

Doing compiler optimization with the help of Reinforcement Learning, the project will be using a vowpalwabbit agents and integrate their AI models in the compiler gym. Inscope:-

- Implement VW agents for CompilerGym
- Experiment with some compiler optimization tasks
- Experiment with other ML techniques
- Feature engineering over computer programs and computing the results

Out of Scope:-

- will work only on Compiler gym
- high performance gpu may be required

6 FEASIBILITY STUDY

We will be able to meet our project schedule

1. **Risks Involved**:- Time taken to solve problem is high, exact ML algo will be tough to find
2. **Resource Requirement**:- A good quality gpu may be required.

7 TOOLS/TECHNOLOGY

Python/C++ programming with vowpalwabbit's ML agents will be used.

8 MILESTONES

- Week 1:-
  TASK:-
  - Understanding machine learning basics and algorithms who are new to ML field.
  - Learn to code ML algorithms like Logistic regression, XGboost,Decision Tree, Linear Regression etc.
    CHALLENGES:-
  - Difficult to understand math and working behind different ML algorithms.
  - Understanding about the decision to which ML model will also be difficult.
  - difficult to code from scratch for beginners.
- Week 2:-
  TASK:-
  - Research about previous Compiler optimization techniques both using ML and other things.
  - Learn about future examples of Compiler optimization techniques, and understand with task of compiler optimization is in the most upcoming demand.
  - research papers from different websites and understand the working.

  CHALLENGES:-
  - It may be difficult to understand Too much conceptual and working of the compiler tasks.
- Week 3:-
  TASK:-
  - Learn about compiler and compiler processes.
  - Getting familiar with Compilergym and Vowpalwabbits environments.
  - start applying Vowpalwabbits ML algorithms.
  - Look for pre-existing methods of compiler optimization in compiler gym.
  - Decide on what compiler tasks we want to optimize.

  CHALLENGES
  - In-depth knowledge of compilergym and vowpalwabbit is extremely required.
- Week 4:-
  TASK:-
  - Start using the vowpalwabbit agents for compiler-gym.
  - Start making the record of each optimization technique result with different parameters

  CHALLENGES
  - Integration of compiler gym and vowpalwabbit it tricky.
  - Major RL and ML implementations take lot of time to run so it is a timetaking process.
- Week 5:-
  An extension of Week4 going in-depth in vowpalwabbit's ML agents and trying more rigorous techniques.
- Week 6:-
  TASK:-
  - Analyze the results of obtained optimization.
  - Think of ensembled techniques to solve the problem.

- Week 7:- Repeat Week5.
- Week 8:-
  TASK:-
  - Finalizing the reports of all the compiler optimization techniques.
  - Crosscheck all the results and start noting the results in a formal documentation.
- Week 9:-
  TASK:-
  - Start learning and implementing other ML techniques for stretch goals.
  - Learn about feature engineering for computer programs.
- Week 10:-
  TASK:-
  - Do feature engineering on programs and compute the results of different programs of similar kind.
  - Generate both supervised and unsupervised data.
  - Try out all the machine learning algorithms for compiler optimization.

  CHALLENGES
  - Correct concept of feature engineering is very difficult
- Week 11:-
  TASK:-
  - Compute the results of all feature engineered programs over the all machine learning optimization algorithms.
- Week 12:-
  TASK:-
  - Aggregate the final results.
  - Go through all the completed tasks and confirm their output.
  - Deliver the final Github repo with a project report.

9 WHY DO I WANT TO WORK ON THIS PROBLEM SPECIFICALLY?

First reason is that I am very much fond of machine learning. I am always looking forward AI/ML to solve real life and different computer science problems. As of now I am already working on application of AI/ML in fashion industry field and when I saw the projects list and got through this topic I was already thinking on the vast number of use cases of the problem and how can it benefit the programmers. I already have a very good knowledge about Machine Learning, Reinforcement Learning and this will provide me much better understanding of concepts and stand points.I will be very eager to work on this project and learn the vast use of machine learning in compiler optimization.

10 REFERENCES

https://docs.microsoft.com/en-us/archive/msdn-magazine/2015/february/compilers-what-every-programmer-should-know-about-compiler-optimizations