Kenneth Chuen and Anthony Krivonos March 20th, 2021

Prof. Gail Kaiser COMSE6156

Final Project First Progress Report

Team Name: Gren Maju

Team Members: Kenneth Chuen (kc3334) and Anthony Krivonos (ak4483)

Team Roles:

- **Anthony Krivonos (ak4483)**: Framework Developer

  Responsible for architecting, developing, and pushing into production (on PyPi) the Qubot framework. Uses OpenAI Gym to create a custom RL environment for the model, as well as Selenium Webdriver to navigate the websites under test.

- **Kenneth Chuen (kc3334)**: Framework Tester and User Study Facilitator

  Responsible for extending the Qubot GitHub repository with code and documentation on the studies to be performed. In the study of Qubot on 10 different pieces of software, the framework will be pulled from PyPi as if it were an external library, and then used to exploratorily test the websites. Responses from several developers in the user studies will be collected and documented in the repository as well.

Novelty

Qubot will be one of the few, if not the only, fully autonomous exploratory testing framework published for Python. It will be able to crash test and simulate page flows for a wide variety of websites that do not require hyper-specific input generation nor difficult login mechanisms. Additionally, testers will have the options to specify their entire tests in a .qu JSON file, largely removing the need for coding knowledge.

Value to User Community

Qubot will be most valuable to small developer teams working on websites that do not have the resources to hire QA testers, as well as QA testers in larger companies who do not have a lot of technical expertise, but are tasked with exploratorily testing large portions of code. Each of these user demographics will be able to perform tests and accumulate various analytics on the software they are testing, and Qubot may be used as both a command-line tool and a Python library.

Dataset

Qubot learns on the fly by scraping interactive HTML elements off websites and converting user interactions into a UITree. Then, it uses the UITree that it just learned to navigate to the right HTML element or webpage. Hence, it will not be using any external datasets. That being said, the ground truth we will be comparing Qubot against is manual interaction with the website under test. For instance, if it takes 3 button clicks to get from Page A to Page B on a website, we expect Qubot to be able to learn to do this consistently within a reasonable amount of epochs. If it fails to achieve such a goal, then we will mark the test as having failed.

Comparison Subjects

Given the meager amount of source code for Q-Learning testing agents that is currently available online, we will be unable to compare Qubot to techniques written in papers such as DRIFT and Fastbot. That being said, we found a project on GitHub, called Selenium-AI, that we will try to make some comparisons against during our experiments. This work will simply be obtained by pulling the project from GitHub and spinning up a Docker instance with the image from this repository.

Project Delivery

All source code and documentation will be available on this public GitHub repository. The link to the Qubot PyPi project will be posted in the README.md file when it is published.

Open Source Software Used

OpenAI Gym: used for custom RL agent environment

Selenium Webdriver: used for webscraping