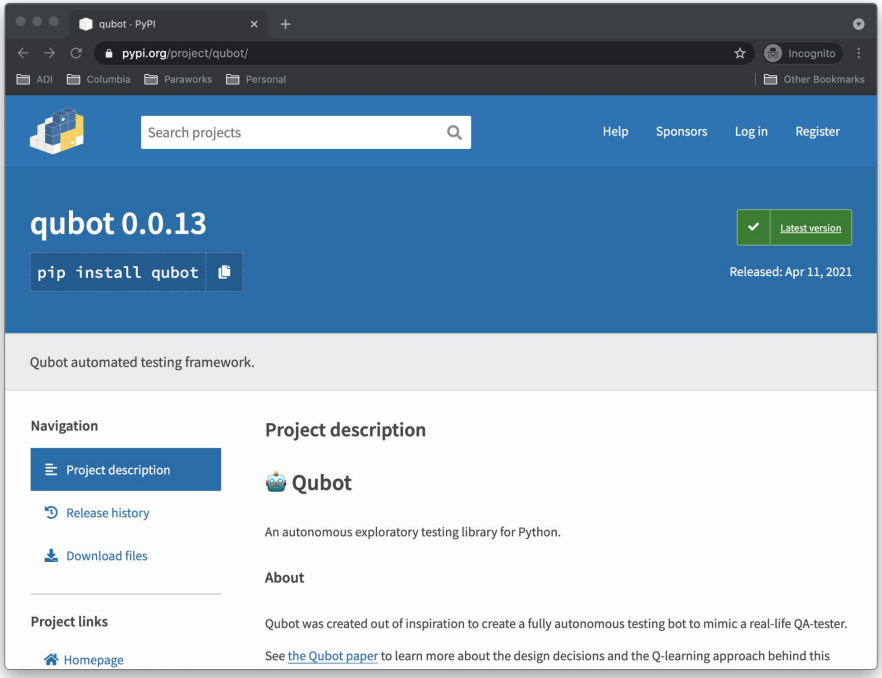
Qubot

An Autonomous Exploratory Testing Library for Python

Anthony Krivonos and
Kenneth Chuen
(Team Gren Maju)



https://github.com/anthonykrivonos/qubot
https://pypi.org/project/qubot

BACKGROUND

Designing the Framework

- **Qubot: Designing a Bot to Perform Autonomous Black Box Testing** (Anthony's Midterm Paper)
  - **SAT Architecture** from *Automated Softwa Testing Framework for Web Applications*
  - **Q-Learning Technique** from *DRIFT: Dee Reinforcement Learning for Functional Software Testing*

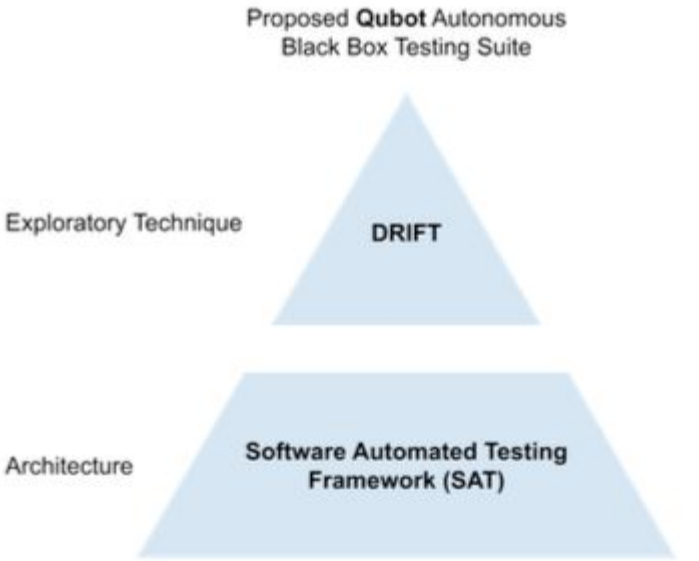


FIG. 17: Proposed Qubot Framework

Qubot An Autonomous Exploratory Testing Library for Python

https://github.com/anthonykrivonos/qubot
https://pypi.org/project/qubot

DEVELOPMENT

Development and Testing Flow

- Python project
  - → Selenium Webdriver **crawls the website** under test (WUT)
  - → DOM converted into a **custom UITree**
  - → Q-learning agent **traverses UITree**
  - → Agent **finds terminal** DOM element(s)

```
NAVIGATE: <html id="" class="">
    NAVIGATE: <body id="" class="">
        LEFT_CLICK: <a id="" class="my-website">
        NAVIGATE: <div id="about-me" class="content">
...
```

FIG. 2: A Sample UITree Printed to Console

https://www.selenium.dev/images/selenium_logo_square_green.png

Qubot An Autonomous Exploratory Testing Library for Python

https://github.com/anthonykrivonos/qubot
https://pypi.org/project/qubot

USAGE

3 Techniques to Begin a Test

```
qb = Qubot(
    url_to_test="google.com",
    terminal_info_testing=...,
    terminal_info_training=...,
    driver_params=...,
    model_params=...,
    reward_func=...,
    input_values=...,
)
qb.run()
print(qb.get_stats())
```

1. Define Qubot Configuration Programmatically

qu_config.json

```
{
  url_to_test=”google.com”,
  terminal_info_testing: {...},
  terminal_info_training: {...},
  driver_params: {...},
  model_params: {...},
  reward_func: 1,
  input_values: {“text”: “abc”}
}
```

```
from qubot import Qubot

qb = Qubot.from_file(
        './qu_config.json')
qb.run()
print(qb.get_stats())
```

2. Define Qubot Configuration in JSON, Execute in Code

qu_config.json

```
{
  url_to_test=”google.com”,
  terminal_info_testing: {...},
  terminal_info_training: {...},
  driver_params: {...},
  model_params: {...},
  reward_func: 1,
  input_values: {“text”: “abc”}
}
```

```
~ qubot ./qu_config.json
```

3. Define Qubot Configuration in JSON, Execute in Command Line

Qubot An Autonomous Exploratory Testing Library for Python

https://github.com/anthonykrivonos/qubot
https://pypi.org/project/qubot

OUTPUT

Sample Test Output

- Qubot **records statistics** while running tests
- Can be **JSON-outputted**
- Statistics class exposed, so testers can **add their own measurements**

output_stats.json


```
{
    "construct_ui_tree_time": {
        "avg_millis": 258399.23095703125,
        "max_millis": 258399.23095703125,
        "min_millis": 258399.23095703125,
        "times": [
            258399.23095703125
        ]
    },
    ...
    "elements_encountered": {
        "count": 157,
        "events": [
            ...
            "<meta id=\"\" class=\"\">
(fc6beb15-e9f0-9d43-8122-7853dbc249d2)",
            "<body id=\"\" class=\"\">
(ec15c835-4133-364a-8274-acf437f63b92)"
            "<span id=\"\" class=\"ribbon-strip green\">
(a1ce8c3b-034c-1246-bff2-6b4cbab51f77)",
            ...
        ]
    },
    "elements_inputted": {
        "count": 2,
        "events": [
            "<input id=\"gi-EMAIL\" class=\"\">
(bf3cf488-2c96-1e44-b6b2-2ffb03dde5f8)",
            "<input id=\"gi-EMAIL\" class=\"\">
(c24baec5-5a49-0545-891d-83fd2333aeb6)"
        ]
    },
    "step_count": 2000,
    "reward_sum": -995
}
```


Qubot An Autonomous Exploratory Testing Library for Python

https://github.com/anthonykrivonos/qubot
https://pypi.org/project/qubot

TESTS

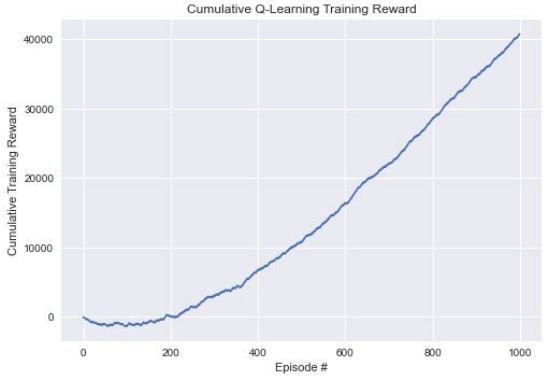
Field and Developer Experience Tests

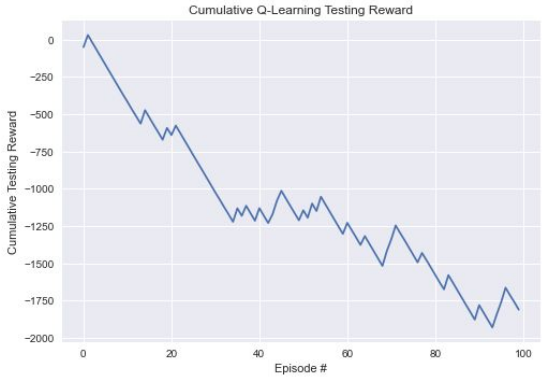
- Field Test: attempted to test 10 various webpages and record results
- Developer Experience Test: asked fellow peers to perform their own tests using Qubot
- *(Also tried to compare to Selenium-AI (small GitHub repo) but failed due to insufficient instructions and outdated Docker image)*

TESTS

Test Results

- Performs well on **single-page sites** but fails to generalize—**does not perform well on testing data**
- Developers' pros:
  - Config file is portable and CLI command is auto-installed
- Developers' con:
  - ML params are TOO configurable (i.e. "what the heck are α and γ?"





Qubot An Autonomous Exploratory Testing Library for Python

https://github.com/anthonykrivonos/qubot
https://pypi.org/project/qubot


(base) ~ Desktop touch test.json
(base) ~ Desktop vim test.json
(base) ~ Desktop qubot test.json

Qubot

QUESTIONS?