Robot Testing from a machine learning perspective

Vinod Yadav[1*]
Faculty of Applied Informatics
Tomas Bata University in Zlin
Nam T.G. Masaryka
5555, 760 01 Zlin,
Czech Republic
vyadav@utb.cz

Raphael Kwaku Botchway[2]
Faculty of Applied Informatics
Tomas Bata University in Zlin
Nam T.G. Masaryka
5555, 760 01 Zlin,
Czech Republic
botchway@utb.cz

Roman Senkerik[3]
Faculty of Applied Informatics
Tomas Bata University in Zlin
Nam T.G. Masaryka
5555, 760 01 Zlin,
Czech Republic
senkerik@utb.cz

Zuzana Oplatkova Kominkova[4]
Faculty of Applied Informatics
Tomas Bata University in Zlin
Nam T.G. Masaryka
5555, 760 01 Zlin,
Czech Republic
oplatkova@utb.cz

2021 International Conference on Electrical, Computer and Energy Technologies (ICECET) | 978-1-6654-4231-2/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICECET52533.2021.9698727

***Abstract*—The need to scale software test automation while managing the test automation process within a reasonable time frame remains a crucial challenge for software development teams (DevOps). Unlike hardware, the software cannot wear out but can fail to satisfy the functional requirements it is supposed to meet due to the defects observed during system operation. In this era of big data, DevOps teams can deliver better and efficient code by utilizing machine learning (ML) to scan their new codes and identify test coverage gaps. This study introduces robot testing and machine learning to manage the test automation process to guarantee software reliability and quality within a reasonable timeframe.**

Keywords—automation, big data, machine learning, robotic testing, test automation

I. INTRODUCTION

Software testing is considered a critical activity that must be performed to guarantee the reliability of every software. This testing process enables the software to be executed under clearly defined conditions to detect errors and facilitate software evaluation. Although a higher number of test cases negatively influences the testing process in terms of cost, minimizing the quantity of these test cases while maintaining software reliability remains a key challenge during software testing. This paper focuses on machine learning (ML)-based automated testing (robot automated testing) [2].

Whenever software developers encounter problems while working on their machines, they create a new branch, fix the problem, commit, push the code with the newly created unit and finally merge the updated code into a master branch for deployment into production. After that, the newly formed branch gets deleted. As developers work on their local machines independently, managing their work becomes difficult. To resolve this challenge, we used a cloud-native application framework called OpenShift. We created dynamic test cases using a robot framework and ML to provide more efficient tests. This is because we can test only a few test cases statically; however, using robot framework and ML, we can have multiple dynamic generated test cases. We can also run automation or robot testing for message protocols, SSH, HTTP Rest endpoints, JMS, SOAPUI API, FTP, XML, JSON data, etc…[6].In the test scenario, messages that are sent and received are predefined. The message flow is specifically defined by a tester for use cases. Automation Testing refers to using testing tools that avoid manual intervention to discover defects in software. For software testing, several testing frameworks are available, with the most common being Data-Driven Testing Framework (DDTF) and the Test-Driven Framework Development (TDFD)[4]. The tester defines a message flow the use cases will use. The literature review is next to this introduction, followed by a snapshot of robot testing combined with machine learning and our methodology. The paper comes to a close with a brief discussion and conclusion.

A. Robotic Testing Framework (RTF)

RTF [5] is a free-to-use online automation testing framework for acquiring, testing and obtaining test-driven based development. It uses many types of tests: keyword, behaviour, and data-driven for writing test cases. Robot Framework (RF) offers automation and database tests via its external libraries.

```
*** Settings ***
Library         String
Library         JSONLibrary
Library         DatabaseLibrary

*** Variable ***
${result}=  ok
&{headers}=                    Content-Type=application/json
Authorization=Basic ABCDEF==

*** Test Cases ***
Sample Database Test
    [Documentation]    Student Details Test.
    connect To Database Using Custom Params  pymysql
database='student',  user='student',  password='student',
host='localhost', port=3305

${db_StudentId} =    Query    SELECT id FROM student
WHERE department_id = 101;
```

Funding Agency: IGA/CebiaTech/2021/001

- It is considered a generic open-source (including libraries and tools in the ecosystem) automation framework released under Apache License 2.0. RF performs test automation and robotic process automation (RPA). It is dynamically supported, with many industry-ruling companies using it in their software development.
- RF is open and extensible, can be merged with any tool to build robust and adaptable automation solutions. Being open-source also means it is free to use with no licensing costs. It has several external libraries such as database, collections, JSON, string, etc.… which contains utilities meant for the Robot Framework's usage.
- It allows us to fire all types of SQL queries in our database library after initiating an action to verify the results. RF-based applications are independent of the operating system. It is run using both JPython (JVM) and Iron Python (.Net) with Python as the main framework.
- First, we need to install all of the essential libraries for robot automation and database testing; after that, we must add them to the robot's setting components and begin the database connection. After connecting the database, we would have access to write all SQL queries in test cases as needed and then develop scripts for machine learning testing.

B. *Machine Learning*

Machine learning (ML) is a sub-field of Artificial Intelligence (AI) with the sole aim of emulating data learning activities[8]. It provides methods of identifying current and acquiring future knowledge to improve and realize self-perfection. Before diving into how ML can give both of these phases of the robot test automation process, it is crucial to understand the base problem of the robot test automation process. It is essential to understand why test automation is unstable in the absence of ML. For example, non-ML test data are static. They cannot automatically adapt to changes that are test data-dependent. With ML, test data have the opportunity to add the following features: automatically scan the new code, analyze security issues and identify test coverage gaps. ML algorithms used for tasks such as decision-making automatically validate and compare the latest datasets based on predefined earlier datasets. The section below demonstrates some real-time applications of robot framework (RF). Machine learning(ML) is a core artificial intelligence research area that mimics human learning activities[8].

It investigates ways to identify current and new information and strategies for enhancing traits to achieve self-perfection.

In our test scenarios, software engineers often run their test cases from the beginning every time a build is finalized, which, of course, takes time and effort. As a result of adopting this trend, the system will become intelligent. If a developer conducts a test, the results and scenarios will be transferred to other nodes, allowing them to benefit and learn from the outcomes.

II. LITERATURE REVIEW

In [1], the authors proposed a software testing framework with Generative Adversarial Networks (GAN), which generates test cases for the software under test to enhance test coverage. GANs use two neural networks, pitting one against the other to learn the underlying distribution of the training data to generate new data instances that resemble training data. GAN consists of a generator and a discriminator. The generator takes random noise and generates fake data. The discriminator checks whether the generated data is real or false. GANs estimate generative models using an adversarial scheme. Their main idea was to utilize the test inputs and the corresponding execution path as inputs of GAN. Satish in [3] implemented an automation testing framework for testing web applications via the Selenium WebDriver tool where they argued that their tool helped in dynamically changing web applications.

Fig 1 Protocols supported by SOAPUI

SOAP-> Simple Object Access Protocol.

WSDL-> Web Service Definition

REST-> Representational State Transfer.

JDBC-> Java Database Connectivity

A. *SOAPUI*

SoapUI is an API testing tool pair of accessible, open-source, and cross-platform [7]. A calm-to-use graphical user interface and enterprise-class qualities allow you to quickly and rapidly build and execute automated practical, regression, compliance, and load tests. It is a functional API testing tool that runs non-functional testing such as performance and security tests. Since the Robot Automation test is different, Robot testing provides

functionalities such as SOAPUI API test and database test. It can improve the quality of code (clean code) and bug-free application if the requirements are properly defined and managed and the right tool for the correct application matches the application's needs. Whenever we encounter any problem in our test case, our robot test directly points out and displays the location and error message on our HTML page (see figure 4).

Example

```
*** Settings ***
Library  SoapLibrary
Library  Collections
Library  Operating System
Library  XML
*** Variables ***
${ClientUrl}  http://endpoint.com/ sampletest.asmx?wsdl
*** Test Cases ***
Create Soap Client  ${ClientUrl}
${response}  Call SOAP Method With XML
    request.xml
${result}  Get Data From XML By Tag ${response}
    result
Log ${ result }
```

B. Rest Endpoints

Rest endpoints testing involves evaluating all rest APIs and requests such as Get, Post, Put, and Delete with the proper response status and response body.

Example

```
*** Settings ***
Library  RequestsLibrary
Library  JSONLibrary
Library  String
Library  DatabaseLibrary
*** Variables ***
${headers}=  Content-Type=application/json
Authorization=Basic ABCDEF==
*** Test Cases ***
${response}=  GET  http://endpoint.com/sampletest
expected_status=200  ${headers}
Log  SampleTest Body is: ${response.text}
```

C. Http and Https

RequestsLibrary is a Robot Framework module that wraps the well-known Python Requests Library to enable HTTP and HTTPS API testing functionality.

Example

```
*** Settings ***
Library  RequestsLibrary
*** Variables ***
${headers}=  Content-Type=application/json
Authorization=Basic ABCDEF==
*** Test Cases ***
${http_response}=  GET  http://endpoint.com/httptest
expected_status=200  ${headers}
${https_responses}=  GET  https://endpoint.com/httpstest
expected_status=200  ${headers}
Log  SampleTest Body is: ${ http_response.text}
Log  SampleTest Body is: ${ https_response.text}
```

III. Methodology

When we start the robot automated software testing, it is crucial to create a list of requirements of test cases. This research discusses two primary tools, namely: Robot Framework, Machine Learning. Recently, several companies have been utilizing robot automated testing tools for various applications because of their advantages compared to other testing tools. This study segment presents five test automation tools used by other researchers in the last few years. Figure1 shows the possible types of tests done using the SOAPUI API in our work.

A. Team Foundation Server (TFS)

TFS is a product of Microsoft Soft, now called Azure DevOps Server. The lightweight operations actualized by the tools and environment can authorize development teams to achieve the target without affecting productivity. It is somehow similar to GitLab, where we create as many repositories as we want and integrate them with production into OpenShift.

B. Jenkins Server

Jenkins, written in Java is a free-to-use online automation tool with plugins designed for integration purposes[9]. Jenkins obtained a build for a single exchange made in the TFS repository. Once the code is ready, we immediately deploy and test it on the testing server. Accordingly, development teams are given updates on build and test results.

C. Openshift

Red Hat develops OpenShift. It is a family of the containerization software product. The OpenShift has developer-oriented views, and developer views are oriented around working with application resources within a namespace also provides a CLI that supports a superset of the actions that the Kubernetes CLI provides.

Funding Agency: IGA/CebiaTech/2021/001

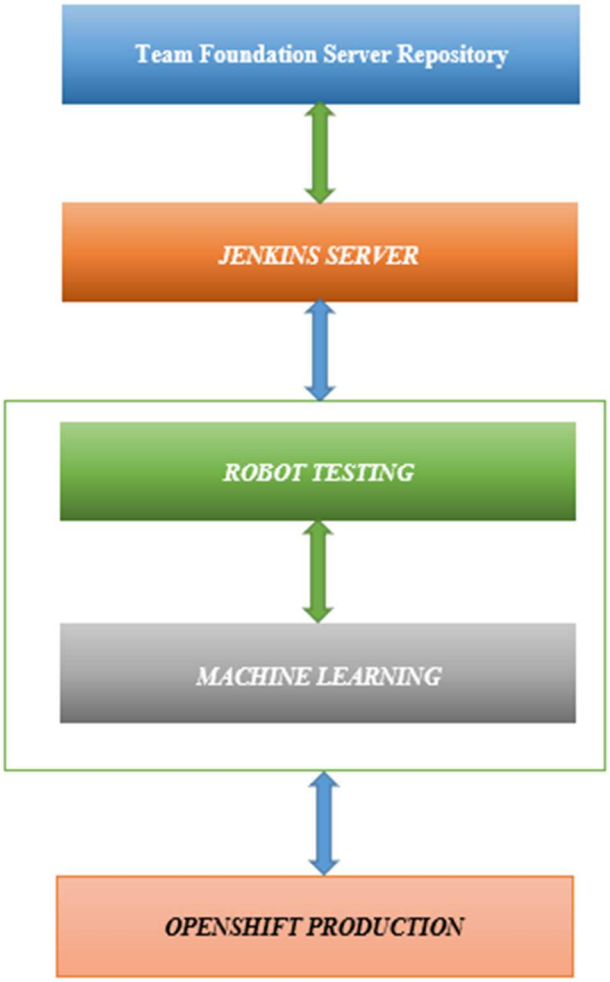
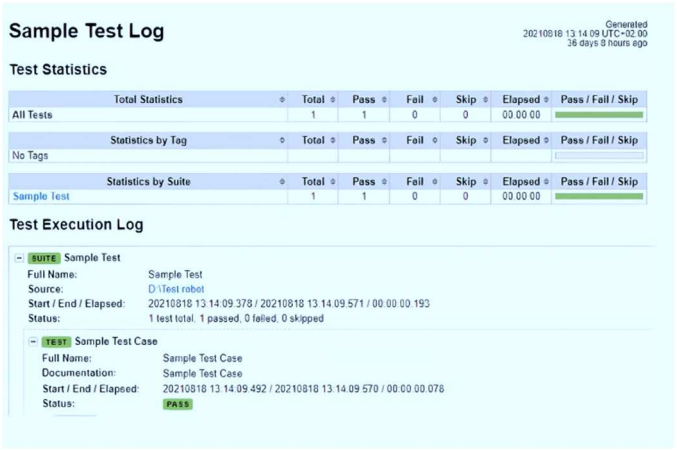


Fig. 2. Conceptual framework of the study

IV. DISCUSSION AND CONCLUSION

In this work, we presented a conceptual framework for RF and ML and performed functional testing of a web application developed with JAVA11, Machine Learning, and Robot Framework 4.0.2[10]. Robot Framework Automated Database Testing can help software developers evaluate database-based web applications, write all SQL queries as required in test cases, and reduce the effort to perform this task compared to manual or semi-automated tests. Additionally, ML provided the robot with automated tests cases, which generated similar test cases. The software testing tool can be selected based on the application needed to be tested, budget, usage, and efficiency. Jenkins compiles the program codes and prompts the developer when the violation occurs. In this paper, we have also shown how the adoption of database testing can help a company achieve better results, high performance, and reduce test execution time. Software companies must embrace robot automation testing-based ML to optimize tests execution time and effort by automating the whole software testing process, leading to an intelligent system. When we mention ML within the DevOps pipeline, it is essential to consider how ML can monitor ongoing Continuous Integration (CI) builds and point out trends within build-acceptance testing, API testing, and other testing areas. ML models can investigate the entire CI pipeline for failed builds. CI builds are often not appropriately reviewed and repeatedly die without attention. With ML entering this process, the immediate value is a shorter cycle and more stable builds, translating into faster feedback for developers and cost savings.



ACKNOWLEDGMENT

This work was supported by the Internal Grant Agency of the Tomas Bata University in Zlin, under the project number IGA/CebiaTech/2021/001. The resources of A.I. Lab further supported the work at the Faculty of Applied Informatics, Tomas Bata University in Zlin (ailab.fai.utb.cz).

REFERENCES

[1] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. Advances in neural information processing systems, 27.

[2] Guo, X. (2021, June). Towards Automated Software Testing with Generative Adversarial Networks. In 2021 51st Annual IEEE/IFIP International Conference on Dependable Systems and Networks-Supplemental Volume (DSN-S) (pp. 21-22). IEEE.

[3] Gojare, S., Joshi, R., & Gaigaware, D. (2015). Analysis and design of selenium webdriver automation testing framework. Procedia Computer Science, 50, 341-346.

[4] Beck, K. (2003). Test-driven development: by example. Addison-Wesley Professional.

[5] Bihlmaier, A., & Wörn, H. (2014, October). Robot unit testing. In International Conference on Simulation, Modeling, and Programming for Autonomous Robots (pp. 255-266). Springer, Cham.

[6] Stouky, A., Jaoujane, B., Daoudi, R., & Chaoui, H. (2017, November). Improving software automation testing using jenkins, and machine learning under big data. In International Conference on Big Data Technologies and Applications (pp. 87-96). Springer, Cham.

[7] Marale, P. S., & Chandavale, A. A. (2018). Implementation of REST API Automation for Interaction Center. In Intelligent Computing and Information and Communication (pp. 273-277). Springer, Singapore.

[8] Shi, Chunhe, et al. "Machine learning under big data." 6th International Conference on Electronic, Mechanical, Information and Management Society. Atlantis Press, 2016.