CONFORMIQ

Automate the Automation

A CONFORMIQ WHITEPAPER

ARTIFICIAL INTELLIGENCE

THE NEW WAVE TRANSFORMING TESTING LANDSCAPE

AUTHOR: KALYAN TALLURI

EXECUTIVE SUMMARY

In this paper, we shall walk through the high-level core of Artificial Intelligence (AI) in software testing perspective adopted by various organizations and the application of AI in Conformiq's Creator software. With AI, the machine comes up with testing alternatives and automatically optimizes the test case creation from a application/system functional model.

The model is based on the application's GUI, API or any back-end actions that can be constructed through various reverse engineering approaches. With AI, Conformiq generates test automation code automatically and takes care of the impact of changes made to the model for whole end-to-end testing processes. AI already plays an integral role in software testing but will gain more attention in the future. However, there are still some fundamental ideas about testing and quality that will remain for the foreseeable future since that can only be understood by humans.

CONFORMIQ

Automate the Automation

DEFINING AI, ML, NLP, DEEP LEARNING

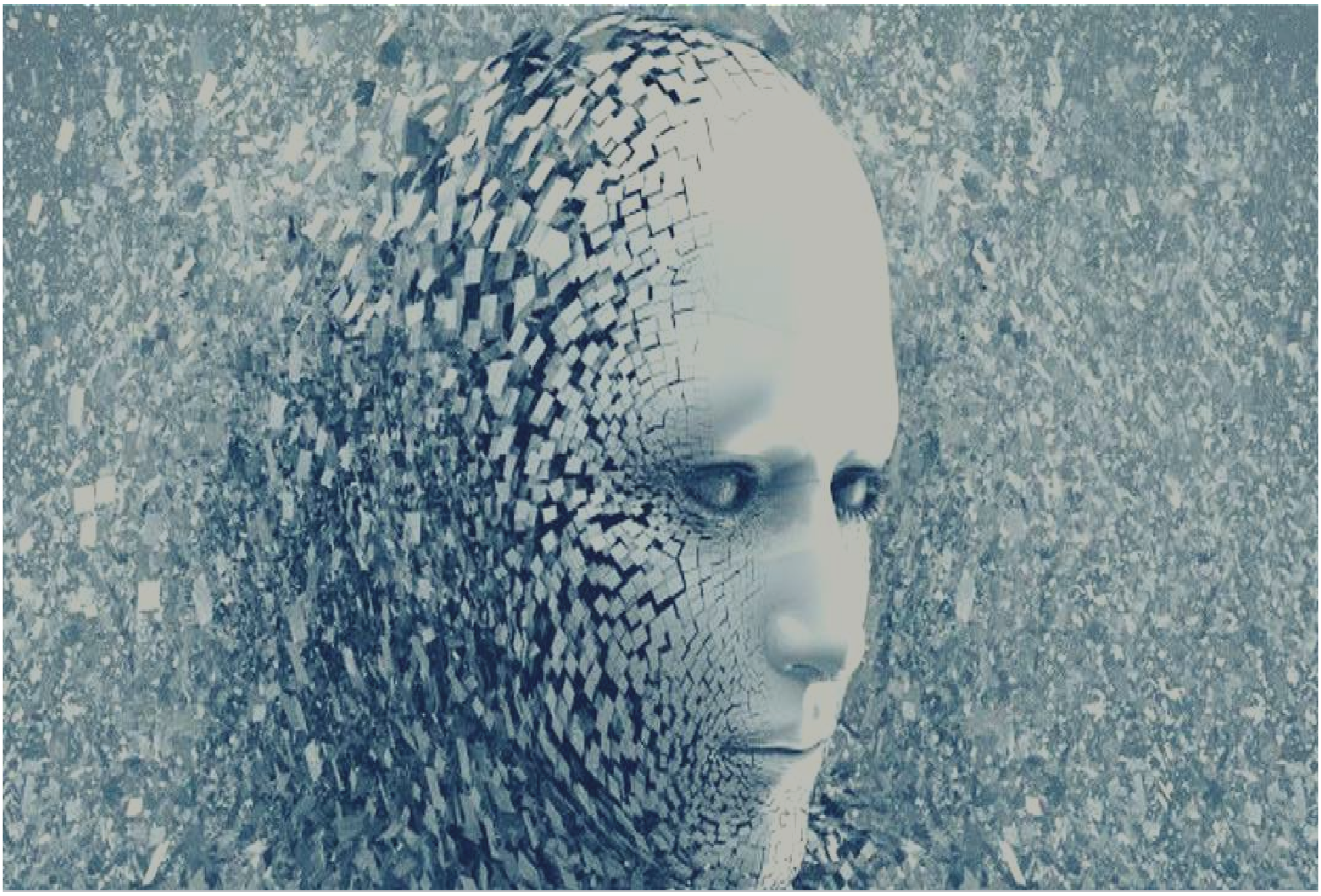
Artificial Intelligence or AI is human intelligence exhibited by a machine. The term AI is applied when a machine mimics a cognitive function such as learning and problem solving.

Machine Learning or ML is a sub-field of AI that concerns itself with the use of data to train algorithms to perform challenging tasks where the "rules" are unclear or difficult to fully specify.

Natural Language processing or NLP is a sub-field of AI that studies the use of algorithms to understand natural language. ML is heavily used in NLP.

Deep Learning is a class of machine learning algorithm that model high-level abstractions in data by using multiple processing layers with complex structures.

Table of Contents



TESTING AND AI - AN INTRODUCTION

With software development lifecycles becoming more complicated by the day and delivery time spans reducing, testers need to impart feedback and evaluations instantly to development teams.

Given the breakneck pace of new software and product launches, there is no other choice than to test smarter and faster, not harder. Thus, it is quite evident that the key to streamlining software testing and making it more smart and efficient is artificial intelligence.

Today, AI and machine learning are centered on training software to understand input data versus output. This is very similar to the testing activities performed manually today. We type an input into a field, and we look for an expected output.

With AI, the machine comes up with testing possibilities and automatically optimizes the test case creation for your testing process. It even handles changes to the automation code and test cases that previously had to be made by QA professionals. Some examples of this are adding fields or changing the inputs and anticipating the outputs.

There are many activities in software testing that can be automated with AI technology, and, by doing so, it enables organizations to transition to fully automated testing in an Agile / DevOps process in a complete software development lifecycle.

CONFORMIQ

CURRENT TESTING CHALLENGES

Software teams are under constant pressure to deliver better quality products in ever-shorter timeframes. To do that, testing has shifted both left and right, and the automation of tests has become critical. However, in the meantime, traditional test automation has become a bottleneck.

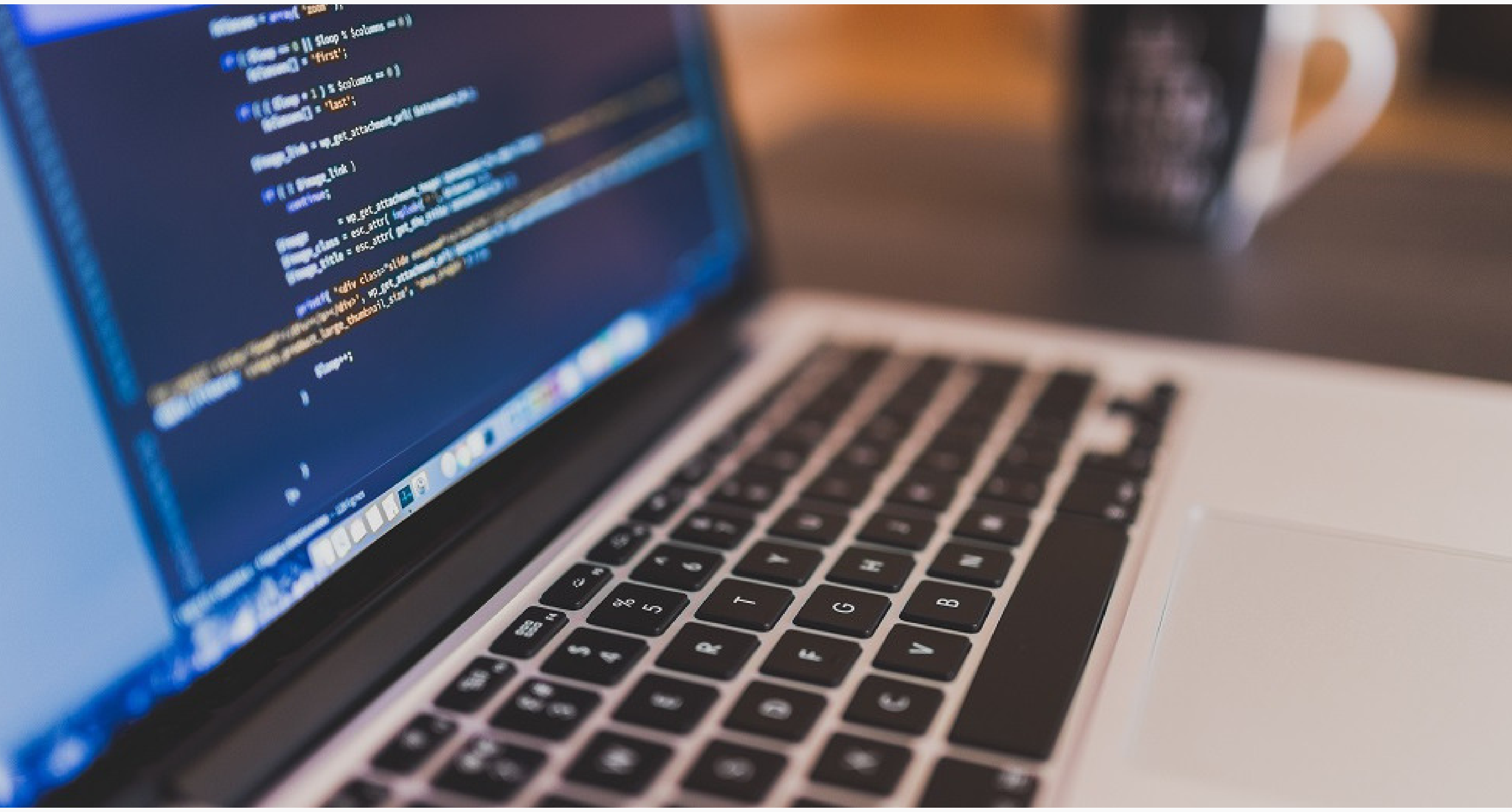
Humans will be unable to carry out an exhaustive testing of an application due to the paucity of time and resources.

Manual test case design, execution and automation development for repetitive testing for a large project scope completely inflates the cost & effort. Thus, affecting go-to-market and overall quality of testing adversely.

Hand-crafted testing not only requires extensive human hours but is prone to inaccuracies and inconsistencies. AI would ensure more foolproof results.

Manual testing faces scalability issues, requiring the management of running several machines.
Manual testers can’t give agile developers immediate assessment on how their constant inputs and alterations to the application have impacted the existing user experience. Help is not available in generating and optimizing test cases, prioritizing testing and automation, enhancing UI testing and reducing tedious analysis tasks.

Testing activity needs someone with domain knowledge. There is a need for folks who not only know how to automate but can also do more headless-based automation – along with analyzing and understanding complex data structures, statistics, and algorithms.



AI USE CASES IN SOFTWARE TESTING

Test engineers can use AI in many places like creating test data, defect analysis, application testing, etc. A part of the testing that is repetitive like regression testing can leverage AI for quicker results.

Some other areas where AI can be leveraged are:

Test Generation

- Automatically generate tests using many test design algorithms, automation code for various test ecosystems and test data

Test Optimization

- Optimize legacy test assets and convert them to automation; propose a minimum set of test cases based on business criticality
- Analyze customer and production data to understand important features and potential automation areas

Test Execution

- Identify screens and elements dynamically in any software and automatically drive application to execute test cases

Test Analysis

- Compares test assets and applications and suggests what changes are meaningful/noticeable. Analyze both code and the tests that run against it, and informs you exactly what your tests are covering and what they're not
- Continuously compare test results to test history to quickly detect changes and regressions, resulting in more stable releases

APPLICATION OF AI IN TESTING – CURRENT MARKET LANDSCAPE

With continuous and increasing demand of embracing AI into software development lifecycle, software providers have no choice but to keep evolving their engineering practices and deliver more innovative solution that leverage AI into software space. Specifically, towards quality assurance efforts, test automation is one such area that is continuously assessing the option of implanting AI bots to deliver top quality products to the market.

Let's look at how IT product vendors and solutions providers have been leveraging AI in their products/solutions:

PRODUCT VENDORS:

Most of the product based organizations tend to have large code base (in some case 100 million lines) which results in large test code base too. This means test execution & associated cost growing exponentially too. Such problems have forced the organizations to develop programs to analyze and determine which tests to be keep and which ones to remove based on what changes have been made to the requirements.

A large social media platform provider launched their own version of AI solution that is used to identify bugs before mobile code is shipped. Based on mathematical logic to perform symbolic reasoning about program execution, the program scans through all apps, for Andriod and iOS reporting any bugs or issues. This saves many human hours allowing users to move fast on mobile.

TESTING SERVICES PROVIDERS:

Global SIs and Indian providers have been made headway delivering AI solutions that focus on:

- Accuracy: Gives more time to focus on complicated process rather than redundant test processes
- Automation: Once created automated tests can be executed again and again without manual intervention
- Saving time & money: Automated test execution reduces the testing timespan from days to hours resulting in faster go to market

Some examples of the solution platform:
AI based platform of a large Indian IT services provider analyzes data gathered from several sources across the IT software development life-cycle and automates the process. The tool is based on the rationale that a defect in an application in production can cause a series of events, which can impact production monitoring systems or cause an end-user interaction to the service desk. Similarly, another provider claims that their solution significantly reduces the cost of maintenance for both physical and digital assets providing organizations with enterprise-grade artificial intelligence and lowered cost of change.

Along the lines of self-aware and self-monitoring platform, a global SI's analytics-driven platform "learns" the resources it uses in addition to automatic discovery and correction of known faults. The platform is claimed to be designed around three key entities that can swiftly adjust, evolve and make decisions with a testing system: A Smart Asset (can be a test case, test environment or test data and each asset can evolve, self-monitor and self-correct based on environmental factors), Smart integrations (allow various tools, components, frameworks and each of the smart assets to speak to each other within the correct context) and an analytics engine (includes cognitive capabilities powered by smart techniques and insights which test architects can use to prioritize what needs to be tested, optimize testing efforts, and identify areas of improvement).

All these platforms allow testing team to easily integrate tools from leading test automation tool providers like Conformiq, Micro Focus, Sauce Labs, Tricentis, Worksoft, etc. This integration also compliments end-to-end test automation and DevOps reducing deployment time from weeks to days with increased software quality.
With significant development in analytics, artificial intelligence and data visualization and availability of the solution around these areas, it's now possible to not only find these areas of inefficiencies and eliminate them, but also to uncover business risks and opportunities that weren't visible before.

APPLICATION OF AI IN CONFORMIQ

Conformiq introduced Artificial Intelligence with the idea that everything in test lifecycle should be automated and works without human intervention using integrated tools & systems, delivering better testing than current application test teams.

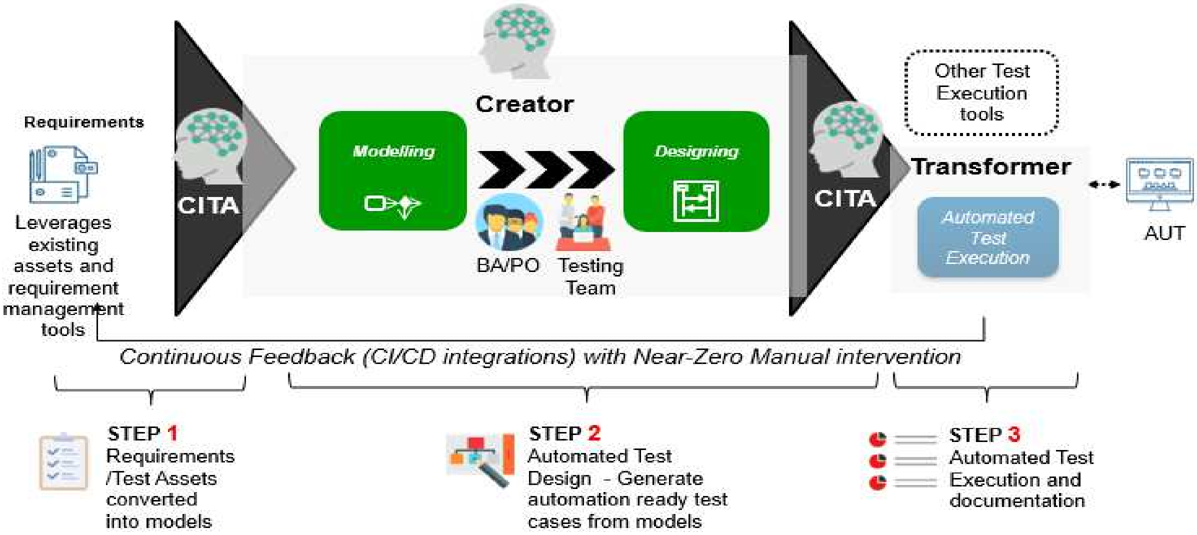

Requirements
Leverages existing assets and requirement management tools
CITA
Creator
Modelling
Designing
BA/PO
Testing Team
CITA
Other Test Execution tools
Transformer
Automated Test Execution
AUT
Continuous Feedback (CI/CD integrations) with Near-Zero Manual intervention
STEP 1 Requirements /Test Assets converted into models
STEP 2 Automated Test Design - Generate automation ready test cases from models
STEP 3 Automated Test Execution and documentation

Figure 1. Conformiq AI architecture

Conformiq leverages AI in three aspects:
1) Automated model construction
2) Automated test case generation
3) Automated test automation script generation

01 AUTOMATED MODEL CONSTRUCTION

Conformiq is applying numerous AI techniques including Natural Language Processing (NLP) for automatic graphical model creation. This is a capability to "reverse engineer" a formal existing testing asset such as manually written test cases. Models are automatically constructed from existing business process models available in BPMN/UML standards, existing tests cases written in a natural language either in Gherkin or traditional step-expected results formats. From the application scanners/spy/recorders, graphical models can also be automatically constructed. While constructing the model, all the input assets are processed for redundancy analysis and optimization. If the assets are manual/automation tests, while constructing the graphs/models, respective conditions/constraints will be added automatically. Test suite attributes and test data can further boost the quality of the models.

With this capability, duplicate tests and subtests are automatically eliminated from the import to generate an optimized model. It is a highly valuable and important piece of technology allowing legacy test assets to be maintained when deploying Conformiq technology but also allowing the user to jump start their modeling efforts, including for additions to brownfield projects, further improving their productivity and making the whole process less error prone.

With Conformiq's solution, our clients are able to achieve:
o Transform manual test cases written in natural text or gherkin notations to test models
o Transform requirement models written in UML or Business models BPMN to test models
o Analyse the test assets for redundancy and eliminate if any duplicate/sub tests
o Automatically construct optimised models from the existing manual test or automated test asset

02

AUTOMATED TEST CASE GENERATION

The core of Conformiq test generation technology is a semantics driven, symbolic execution based, test generation algorithms. This algorithm traverses a part of the (usually infinite) state space of the system model and searches this part of the state space for testing goals and the number of testing goals in a model depends on the testing heuristics selected by the user. The algorithm then constructs tests from the explored part of the state space by selecting paths that lead to testing goals, then converting those paths to tests. Every input on an execution path (to the system model) becomes a test stimulus (to be sent to the real system), and every output of an execution path becomes an expected output (to be verified during test execution).

To keep things practical, the tool triggers yet another algorithm that selects an optimized subset of all the test cases that cover all the testing goals with minimum cost. This step eliminates redundancy in the generated test collection which makes sure that every test case has a reason to be in the test suite. The reason is documented from the generated requirements traceability and model coverage reports available to the user.

To generate good quality tests, Conformiq technology employs an internal mechanism to make predictions on ways the system might fail. It covers areas like:

o Deep state space analysis
o Symbolic data dependencies handled by constraint solving
o Generating full test oracles with test data inputs & expected results
o Model driven coverage and combinatorial optimization
o Automated test data generation

03

AUTOMATED TEST AUTOMATION SCRIPT GENERATION

Conformiq's test script generation technology is called CITA (Conformiq Intelligent Test Automation) and it leverages Artificial Intelligence (AI) techniques like NLP , Deep learning etc to automatically transform manual and generated test steps into a format suitable for automatic test execution. CITA can be trained for any target test automation framework or tool, deriving automation code/scripts automatically from existing test assets, automation assets, test strategies, test data, application user interfaces and automated test design tools.

With CITA, Conformiq users can customize, extend, and maintain generation of automation scripts actions by teaching CITA via excel (no programming required). This supervised learning of CITA helps non-automation engineers like manual testers, business analysts and even developers to accelerate the test automation. CITA's reader(s) normalizes input(s) by removing the external noise and identifies the automation data and objects automatically. CITA, then optimizes initial inputs before transforming tests to automation, leveraging NLP. It works with any input format, any target technologies, and any test automation framework!

CITA does not need to understand the details of a test execution environment and an output format, but instead CITA provides an API , which is then implemented by "exporters". The exporters, as the name suggests, are responsible of exporting the CITA generated unified output in a test execution environment specific fashion. In the case of a Selenium test execution environment, the exporter will take in CITA generated unified output and render the results in a format recognized and understood by the Selenium test environment.

It covers areas like:
o Automated automation code generation script less/scriptful
o Natural Language Processing (Textual analysis)
o Measuring quality, data/noise cleaning
o Test data externalization

CONFORMIQ

SUMMARY

Conformiq technology has greatly benefitted from ongoing AI research and over the years has increasingly incorporated more algorithmic advances into its core technology.

Conformiq AI helps its users:

- Shift left testing for faster time-to-market with improved test quality
- Effectively generating and optimizing test cases, prioritizing testing and automation, enhancing UI testing, and reducing tedious analysis tasks
- Effectively producing maximum automation code and integrating smoothly into any automation environment
- Building test assets faster and easier

ABOUT THE AUTHOR

KALYAN TALLURI

Global Technical Director

Kalyan Talluri is currently working as a Technical Director at Conformiq, product owner of Customer Solutions and focused in bringing automation product/solutions ideas to reality. Expert in Automation products development for Business Process, Functional & 360 degree End to End testing for IT & Embedded enterprises through leveraging advanced analytics and AI techniques. Prior to working at Conformiq, he has been working with Smartesting, Schneider Electric , Honeywell and Intergraph at various levels. Kalyan received his M.Tech from Indian Institute of Science and an MBA from Manipal University , he has been an author and co-author of few publications on testing & automation.