Web Application Testing

Clare Davies, Lunara Nurgaliyeva, Mahmoodreza Afzali, Malik Shaheryar

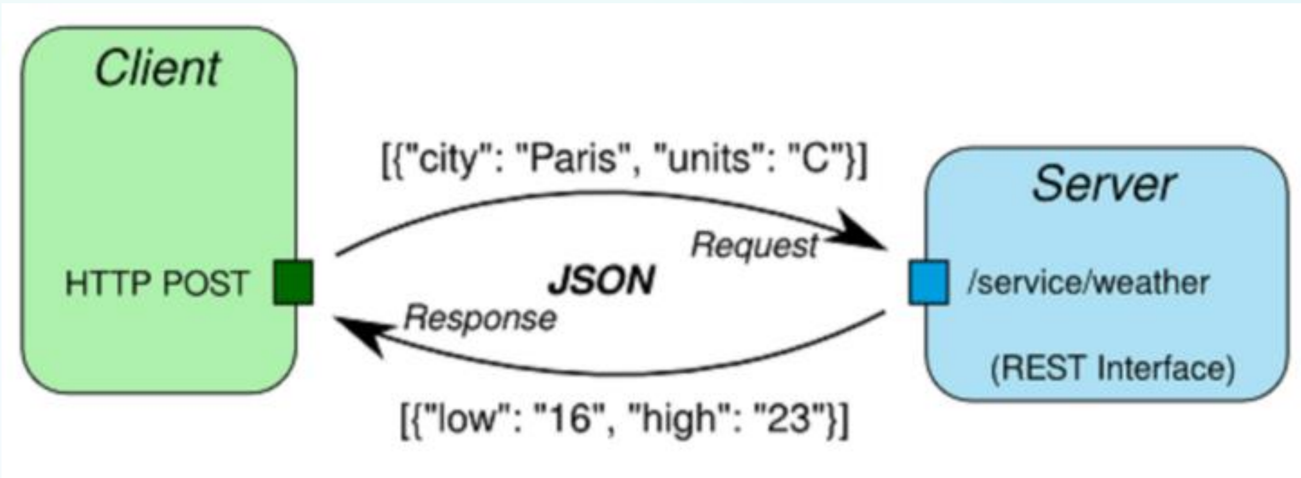
Testify
http://testifytech.ml

Åbo Akademi

Introduction

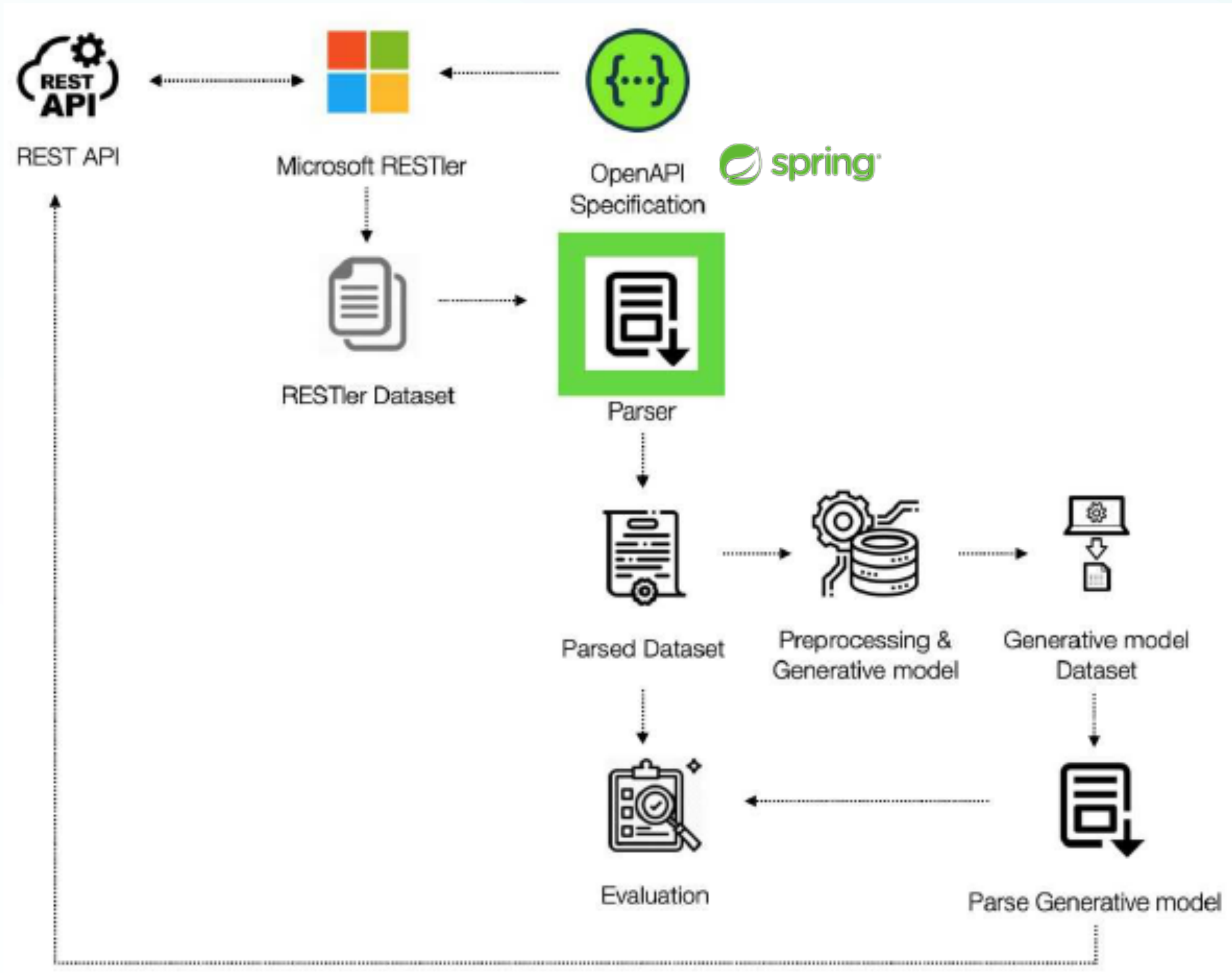
With the ever-increasing demand for web applications, the demand for Rest APIs has gone up exponential, which has led to a huge demand for efficient Rest API testing. This is currently handled with Rest API automation testing, but this has challenges such as requires handling planned failed test scenarios, comparing responses, and sequencing the API calls.

The most common type of API is REST (Representational State Transfer). In REST API when a client sends a request the response is sent back in a standardized way.



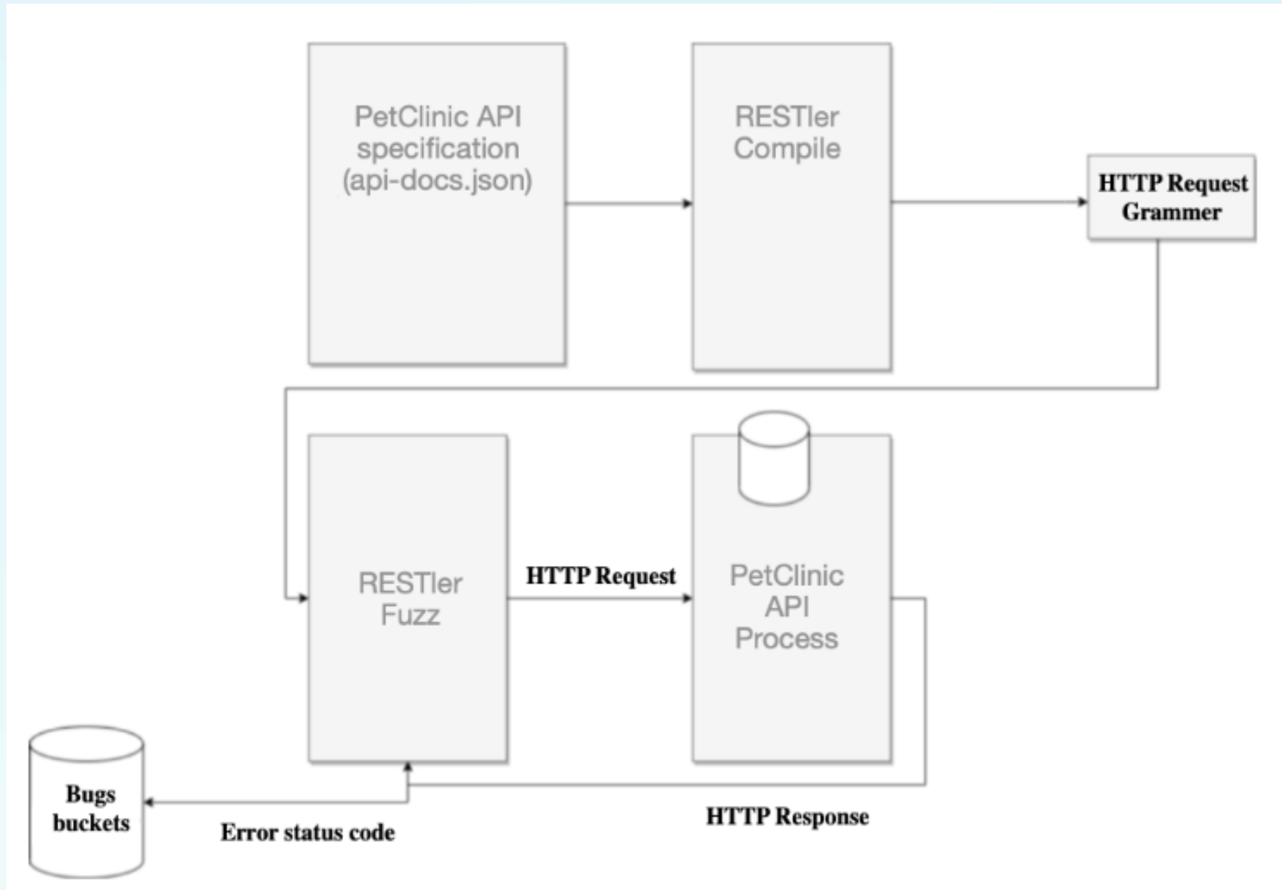
This project aims to look into further automating the Rest API testing and combating the issues highlighted above in the current automation testing by automatically generating the test cases feeding after feeding in the information from the Rest APIs.

This was done by creating test cases using Restler (fuzzy grammer approach) and then feeding the resulting test set into a machine learning generative model. The resulting test sets are then evaluated and compared..



Methodology

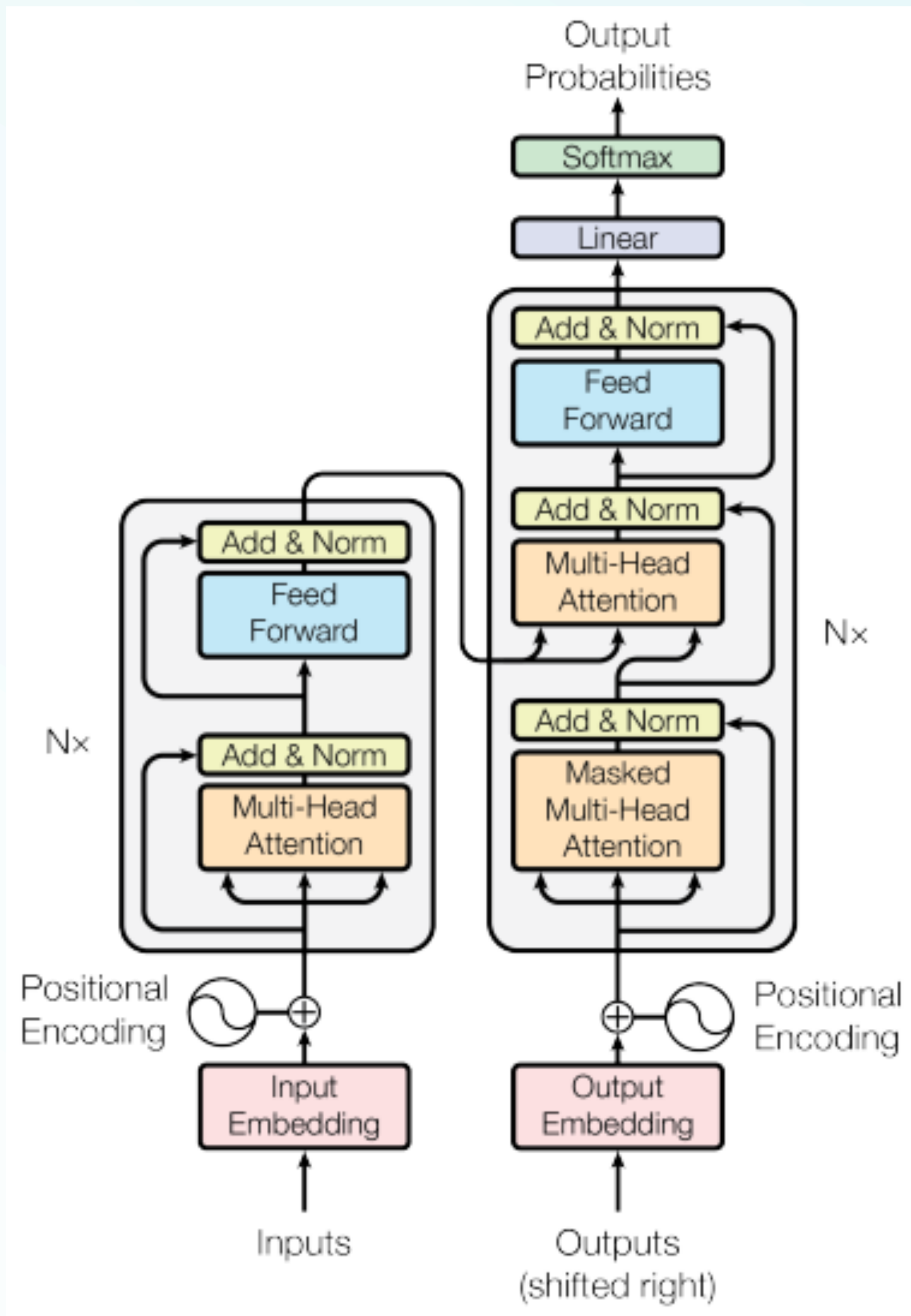
Dataset generation with RESTler



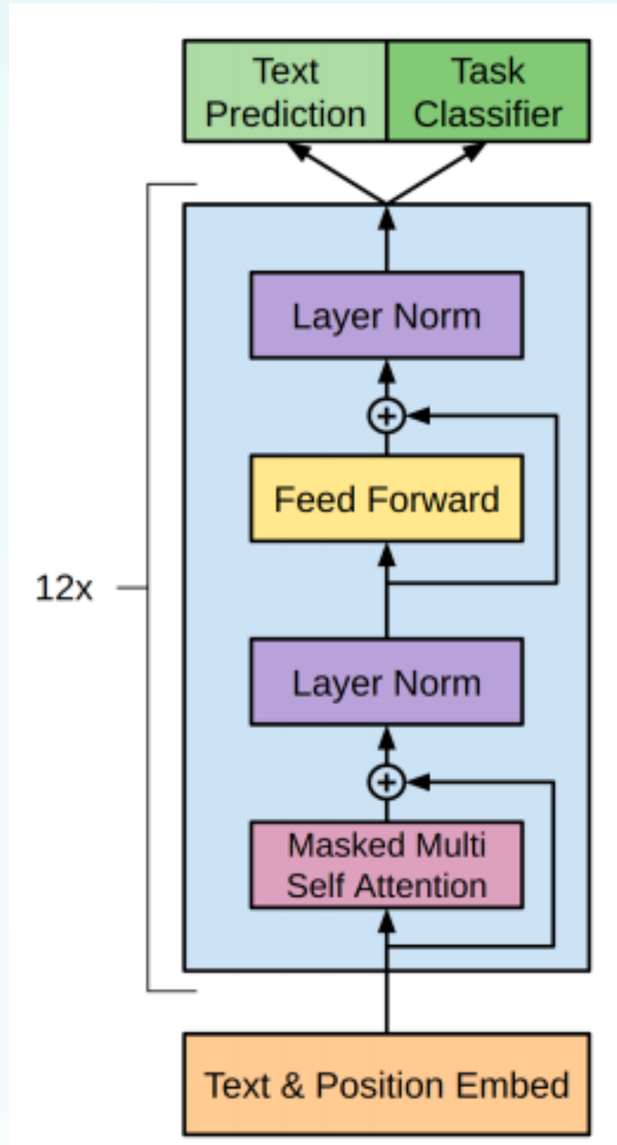
RESTler has two main components, a compiler and an engine. The compiler takes an API specification and produces the fuzzing grammar, which has all the schema information about the requests. The RESTler test engine takes the fuzzing grammar and some other settings in order to run tests and find bugs.

Model architecture

The deep neural network that is used in this work is Generative Pre-trained Transformer 2 (GPT2). It is a generative unsupervised model, which can generate new data similar to existing data during the pre-training step



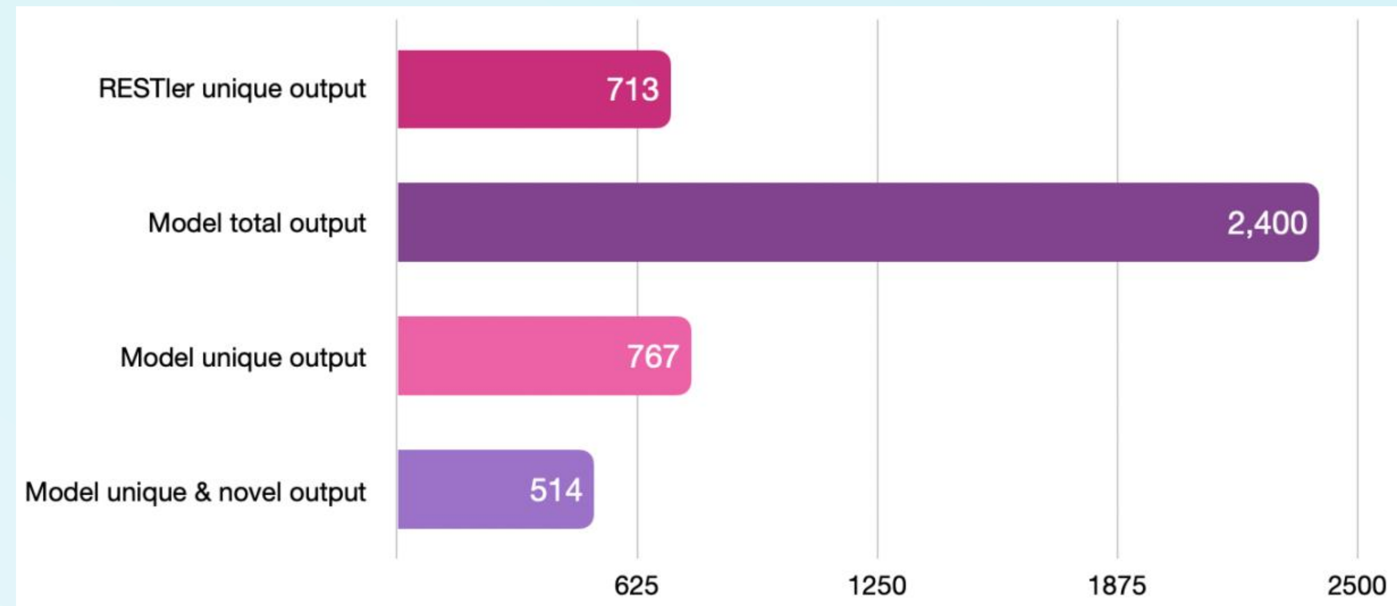
Transformer Model Architecture



Decoder-Only Architecture used by GPT-2

Results

Evaluation Metric	Restler	Generative Model
Total Number of test cases	45,099	2,365
Unique Test Cases	713	787
Number of test cases passed (Response code 200-299)	43,245	216
Number of test cases failed (Response code 400-599)	1,854	2149
Average Response Time	0.01347 seconds	0.00442 seconds
Uniqueness = Number of unique tests / Total tests generated	1.58%	33.28%
Valid Test Cases Percentage = Number of valid tests / Total number of tests	96%	9.13%
Invalid Test Cases Percentage = Number of invalid tests / Total number of tests	4%	90.86%



Evaluation of RESTtler and GPT Model

Anomaly Detection

Anomaly detection algorithms can be categorized into these groups :
1. Supervised: Used when the data set has labels identifying which transactions are an anomaly and which are normal. *(this is similar to a supervised classification problem).*
2. Unsupervised: Unsupervised means no labels, and a model is trained on the complete data and assumes that the majority of the instances are normal.
3. Semi-Supervised: A model is trained on normal data only *(without any anomalies)*. When the trained model is used on the new data points, it can predict whether the new data point is normal or not (based on the distribution of the data in the trained model).

Conclusion

Overall, the generative model successfully created an expanded dataset where there is a higher percentage of uniqueness and endpoints called. However, this test set could be further improved upon by expanding the range of error codes and producing more test cases. Moreover, in the future anomaly detection model also will be improved with provided evaluation metrics.