Measurement & Unit Testing

AI Engineering - Recitation 3

Outline

Measurement

1. Measurement: The Big Idea
2. Measuring Quality Attributes in ML Systems
3. Activity - Malicious Android Applications
4. Data Collection in Production

Unit Testing

1. Unit Testing Overview
2. Activity - unittest
3. Unit Testing: Importance
4. Writing Unit Tests
5. Limitations

Measurement

Measurement

1. Measurement: The Big Idea
2. Measuring Quality Attributes in ML Systems
3. Activity - Malicious Android Applications
4. Data Collection in Production

Measurement: The Big Idea

- Measurement is the numerical quantification of the attributes of an object or event, which can be used to compare with other objects or events. (Source: Wikipedia)

- How to measure intangible quantities?
  - Determine what is observable around it, and then measure that.
  - Often not a single quantity
  - Example: Quality of recommendations measured from user feedback.

- Everything is measurable provided you care enough and are willing put in the effort

Measuring Quality Attributes in ML Systems

- Measuring quality attributes is a hard problem to solve
  - Quality attributes in ML systems are mostly intangibles
  - Multiple attributes are required for accurately measurement
- Not all qualities are relevant to every ML system
  - Identify relevant qualities, and then measure them
  - Every quality attribute has a different measure
- Operationalization of a measure is to turn these abstract quality attributes into something observable and measurable.

Activity - Malicious Android Applications

Scenario:

You are part of the team at Google that reviews Play Store apps to determine if they are malicious. With over 3 million apps, around 4000 new apps are added and 70,000 apps are updated every day. You are analyzing an ML system that scans these applications and automatically flag them for a manual review.

Some Qualities to Consider:

- Accuracy
- Training Cost
- Amount of data needed
- Scalability with the number of features considered
- Effort for data cleaning and feature engineering
- Inference Cost
- Cost of updating model with new data
- Model Size
- Robustness
- Interpretability

Reference: SEAI - Fall 2020 - I2: Tradeoffs

Activity - Malicious Android Applications

Quality	Measure	Operationalization
Accuracy	F1-score	Record precision and recall metrics on production data
Inference Cost	Time taken per sample and compute cost	Record time taken for inference in production, and get the cost analysis from cloud provider
Model Size	Space required for model on disk / memory	Record size of model after every round of training
Robustness	Relative difference between unseen data and training data	Treat production data as unseen data
Cost of updating model	A scale based on complexity of code changes in the model training process	Have human experts determine the complexity of code changes

Data Collection in Production

- Operationalization of any measure is dependent on how well the observation data is collected
- A lot of effort is put on telemetry systems in production
  - Gathering user feedback
  - Monitoring model quality
  - Performance monitoring
  - Detecting drifts and feedback loops
- In production ML systems, the model is probably not the most important component.

Unit Testing

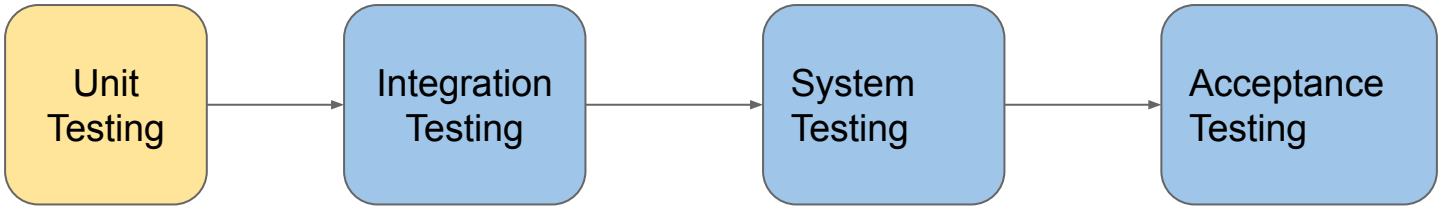
Unit Testing

1. Unit Testing Overview
2. In class Activity
3. Unit Testing: Importance
4. Writing Unit Tests
5. Limitations

Unit Tests

Unit Tests:

- Main types: manual and automated (preferred)
- Tools Examples: JUnit, NUnit, JMockit



Levels of Testing

Unit Testing Activity

Unit Testing: Importance

- Better understanding of goal trying to achieve
- Protect against regression failure
- Documentation
- Find software bugs early
- Code Coverage

Writing Unit Tests

- Descriptive Test Case Name
- 3 A's (Arrange, Act, Assert)
- Appropriate Test Error Message when test fails
- Organize Test Code
- Each test case should be independent of other test cases
- Decompose Code into testable units

Limitations

Hard task if units have many dependencies

Failures may go undetected because of missed scenarios

Do not uncover integration problems

Unreliable Tests

Resources

- Unittest- https://docs.python.org/3/library/unittest.html
- Getting Started with Testing in Python- https://realpython.com/python-testing/
- Mocking objects - https://realpython.com/python-mock-library/#what-is-mocking
- Assert methods - https://docs.python.org/3/library/unittest.html#assert-methods