Recitation 3

Testing for AI-Enabled Systems

Recap: Black-box Testing

- Evaluate the behavior of a system without knowing its internal structure
- You're concerned about the inputs and outputs
- Some types of black-box testing
  - Boundary Value Analysis (with or without robustness variant)
  - Partition Testing (Equivalence Classes)
  - ...

Source: Lecture, Analysis of Software Artifacts, Jan 2020

Recap: Partition Testing

- Partition the input domain into groups (based on domain knowledge)
- Choose **representative** equivalence classes of inputs
    - All inputs in an equivalence class will succeed or fail in the same way
- Partitions are complete (cover the input space), and disjoint (two partitions do not overlap)

Source: Lecture, Analysis of Software Artifacts, Jan 2020

Recap: Finding Equivalence Classes

- Cases in the specification
- Ranges of each input
- Invalid inputs
- Membership in a group
- Properties of inputs
- Possible outputs
- Risk-based (kinds of errors based on inputs)
- …

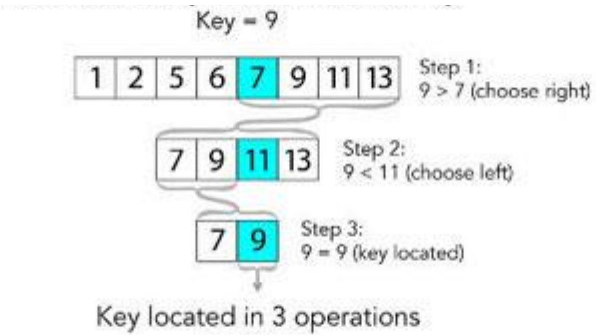
Source: Lecture, Analysis of Software Artifacts, Jan 2020

Recap: Example Equivalence Classes

Let's say we have a method that implements binary search.

public boolean search (int[] array, int elementToLookFor)

What are some partitions / equivalence classes we can test for?



Source: Lecture, Analysis of Software Artifacts, Jan 2020

Recap: Example Equivalence Classes

Let’s say we have a method that implements binary search.

public boolean search (int[] array, int elementToLookFor)

- Array:
  - **Length**: zero length, non-zero length - odd length, even length
  - **Sort Order**: sorted, not sorted

- elementToLookFor:
  - **Output**: not present, present - lowest index, highest index, in between
  - **Number of Occurences**: does not occur, occurs once, occurs 2 or more times

Source: Lecture, Analysis of Software Artifacts, Jan 2020

Challenges with Testing AI

- The oracle problem
- Evaluating the accuracy of the model using a held-out dataset is typically used
  - But… held-out datasets are often not comprehensive
- It's possible to overfit (or) underfit
- There are concerns about fairness, robustness, etc.
- A single overall metric makes it difficult to find out where the problems are
  - What does it mean when you have 90% accuracy for example? Is it sufficient?
  - How do you know where the model is making mistakes, and fix them?

Metamorphic Tests

- Invariance tests
  - Apply label-preserving perturbations to inputs
  - You expect the model predictions to remain the same
  - Example: Credit score should not change based on protected attributes (Fairness)
  - Example: Sentiment shouldn't change - Mark is a great instructor, Samantha is a great instructor
- Directional expectation tests
  - You expect the label to change in a certain way
  - Example: Housing price prediction
    - Increasing the number of rooms shouldn't decrease the price of the house
    - Decreasing the square footage shouldn't increase the price of the house
    - If this did happen, what do you think could possibly be the reason?

Source: Beyond Accuracy: Behavioral Testing of NLP Models with CHECKLIST

Minimum Functionality Tests

- Simple test cases that target a specific behavior
- Small, focused testing datasets
- Can we apply an approach similar to partition testing here?
    - Partition <-> Subpopulation
    - Test cases in each partition <-> Instances in each subpopulation
    - All instances in a subpopulation should ideally behave the same way
        - Might not strictly apply for ML, but this is the general idea
    - Need domain knowledge to come up with reasonable partitions

Source: Beyond Accuracy: Behavioral Testing of NLP Models with CHECKLIST

Sentiment Analysis on Tweets

- Labels: positive, negative, neutral
- Subpopulations
  - Different sentiments (positive, negative, neutral)
  - Different tenses (past, present, future)
  - Comparators, superlatives
  - Negations
  - Typos (tests robustness)
  - …
- Invariant test: Named entities don't affect sentiment
- Directional expectation test: More negative phrases don't make the sentence positive

Test case	Expected	Predicted	Pass?
A Testing **Negation** with ***MFT*** Labels: negative, positive, neutral			
`Template: I {NEGATION} {POS_VERB} the {THING}.`			
I can't say I recommend the food.	neg	pos	X
I didn't love the flight.	neg	neutral	X
...			
Failure rate = 76.4%			
B Testing **NER** with ***INV*** Same pred. (`inv`) after removals / additions			
@AmericanAir thank you we got on a different flight to [Chicago → Dallas].	`inv`	pos neutral	X
@VirginAmerica I can't lose my luggage, moving to [Brazil → Turkey] soon, ugh.	`inv`	neutral neg	X
...			
Failure rate = 20.8%			
C Testing **Vocabulary** with ***DIR*** Sentiment monotonic decreasing (↓)			
@AmericanAir service wasn't great. You are lame.	↓	neg neutral	X
@JetBlue why won't YOU help them?! Ugh. I dread you.	↓	neg neutral	X
...			
Failure rate = 34.6%			

Source: Beyond Accuracy: Behavioral Testing of NLP Models with CHECKLIST

Activity (10 - 15 mins)

- Pick a scenario
  - Driverless Cars - Detecting Stop Signs Based on Images
  - Detecting Unsafe Sidewalks For Wheelchair Users Based on Images
  - Speech Recognition Based on Audio
  - Cancer Detection Based on Images
- In your breakout room, think about subpopulations
  - How can the test dataset be split into smaller test sets (partitions) for the problem you picked
  - What are some critical subpopulations that need good accuracy?
- Update your points in the slides (a separate PPT will be shared in the chat)

Detecting Unsafe Sidewalks - Wheelchair Users

- Flat surfaces
- Type of surface (brick, cement, etc.)
- Presence / absence of vegetation
- Weather conditions
- Presence / absence of shadows
- Lighting
- Quality of the image
- ...

Source: PathVu, MSE Studio Project, 2019-2020

Driverless Cars - Detecting Stop Signs

- Environmental conditions
- Stop signs may be on poles, walls, etc.
- Lighting
- Held by a person
- Occlusions
- Stop signs with modifiers
- Arms of gates, barriers, etc.
- Stop signs at branching roads

Source: ScaledML Conference, AI for Full-Self Driving @ Tesla

Source: ScaledML Conference, AI for Full-Self Driving @ Tesla

Manually curate a set of “binary predicates” that must be satisfied to pass a unit test:

```
Catch-all stop sign detection: 3451/3468 (99.51%)
Heavy rain/snow:  …
Heavily occluded: …
Tilted stop signs: …
Digital stop signs: …
Person held stop signs: …
School bus stop signs: …
Construction stop signs: …
Hanging stop signs: …
Toll booth stop signs: …
Stop signs on gates/arms: …
Correct prediction of relevance: …
Correct prediction of time relevance: …
Stop sign modifier accuracy: …
```

Source: ScaledML Conference, AI for Full-Self Driving @ Tesla

Speech Recognition

- Accents
- Level of intensity
- Gender
- Pronunciations
- …

Cancer Detection

- Quality of the image
- Device used to scan the patient
- Age group
- Gender
- ...

Final Note

- Problems keep evolving, and data keeps changing
- That's why
  - Training and testing on just a single static dataset is not sufficient
  - Using a single aggregate metric for the whole dataset might not be sufficient
- There is a need to
  - Source new examples and augment the train/test dataset
  - Gather telemetry
  - Retrain the model
  - Look for data drift
- Think about subpopulations for movie streaming in your project groups

Thank You!