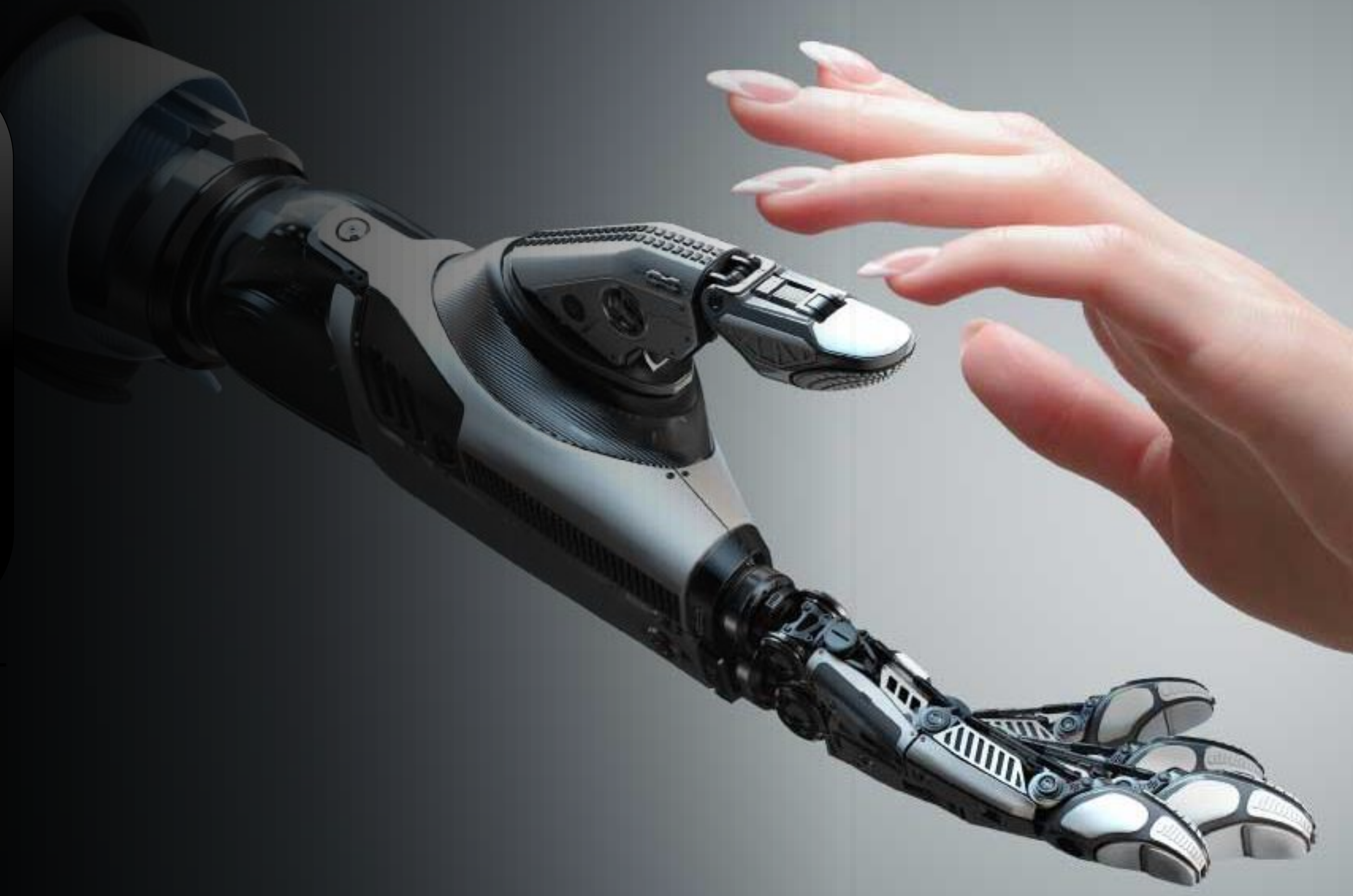
# Unveiling the Code Model Enigma: Interpreting the Gap Between AI and Human Intuition in Code Clone Detection



Presenter:

Shamsa Abid

Research Scientist, RISE Lab, SCIS, SMU

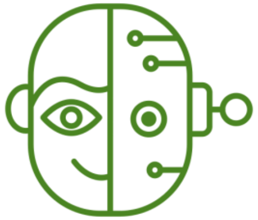
# Introduction

NEED TO UNDERSTAND MODEL'S DECISION MAKING

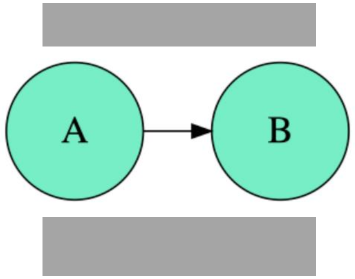
WHY WE NEED TO MOVE BEYOND ACCURACY

WHY WE NEED CAUSAL EXPLANATIONS AND HOW TO GET THEM

# Introduction



In this direction, our goal is to evaluate the performance of models in relation to human intuition.




Specifically, I will discuss how we apply counterfactual data mutations to get causal explanations

allows us to evaluate a model’s reliability and trustworthiness.

# Interpretability Overview

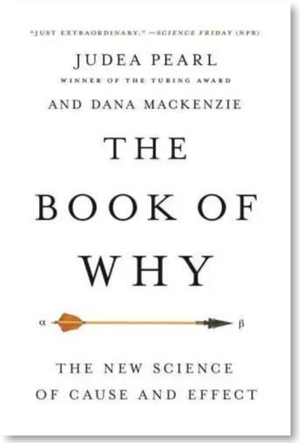
- *Real World Problems*
  - Criminal risk assessment tool, shows racial biases
  - European Union's "Right to Explanation"
- *Software Engineering Problems*
  - Explaining Predictions of Code Tasks
  - Semantic Code Clone Detection Task

| Model | Reported F1-Score | Observed F1-Score |
|---|---|---|
| CodeBERT | 94.0% | 71.11% ↓ |
| CodeGraph44CCDetector | 96.6% | 53.76% ↓ |
| CodeT5 | 97.2% | 65.9% ↓ |

# Interpretability Overview

- Interpretability
  - the degree to which a human can understand the cause of a decision
  - Interpretability also defined as a part of *explainability*.
- Explainable models
  - summarize the reasons for neural network behaviors
  - gain the trust of the users
  - generate insights into the causes of their decisions.
  - *"You were denied a loan because your annual income was £30,000. If your income had been £45,000, you would have been offered a loan."*

# Ladder of Causality: 3 levels of interpretability




**Counterfactuals: (Imagining)**
What would happen under different circumstances?

aims to answer "Why" questions.

**Interventions: (Doing)**
How can I make [...] happen?

"What if" questions.

**Associations: (Seeing)**
How are variables related?

"How would seeing x change my belief in y?"

# Causal Interpretability

- Causal interpretability
  - helps us understand the *real causes of decisions* made by machine learning algorithms, improve their performance, and prevent them from failing in unexpected circumstances

# Causal Interpretability for Clone Detection Models

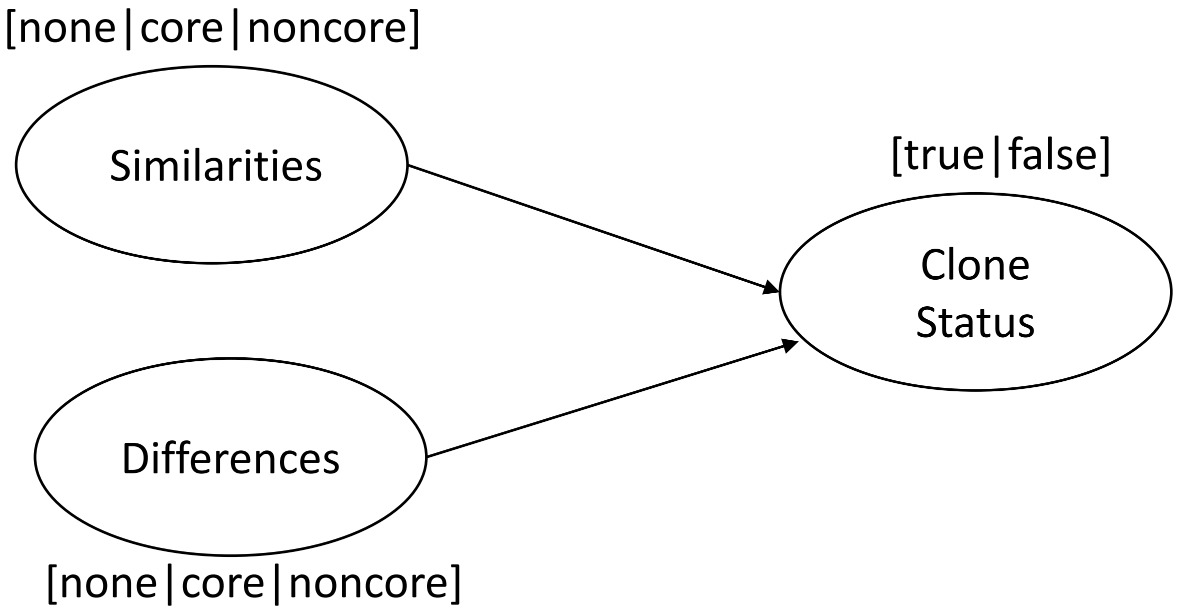
- Was it feature X that caused decision Y ?
  - *"Did the code similarities cause the model to predict the clone as a true clone?"*
- What would have happened to this decision of a classifier had we had a different input to it?
  - *"If we removed the code similarities from a clone pair, would the system still make the same decision?"*
- "Why did the classifier make this decision instead of another?"
  - *Why do we get a false prediction for a clone pair? What's causing the false prediction?*

# Research Goals

- How do we know the real causes of predictions?
  - Are true clone predictions caused by code similarities?
  - Are false clone predictions caused by differences?
  - Are mispredictions caused by distracting similarities or differences?
- How can we decide the best model for clone detection
  - Which is well aligned with human intuition
  - Which is robust, reliable and trustworthy
- How can we measure these attributes?

# How to do Causal Inference

1. **Causal Diagrams**: DAGs used to depict causal relationships between variables, helping to visualize the direction of causality and potential confounding factors.



# How to do Causal Inference

**2. Counterfactuals**: Causal inference often involves comparing observed outcomes with hypothetical outcomes that would have occurred under different conditions or interventions. These hypothetical outcomes are known as counterfactuals.

| Observed Outcome | Intervention | Hypothetical Outcome | Hypothetical == Actual Outcome? | Counterfactual Explanation |
|---|---|---|---|---|
| True clone | Remove similarities | False clone | ✓ | Similarities are causing the model's original prediction |
| | | | ✗ | The model's original prediction is influenced by confounding factors |

# How to do Causal Inference

**3. Measure Causal Effects**: Causal inference quantifies the effect of one variable (the cause or treatment) on another variable (the effect or outcome).

Average Causal Effect Metrics

# Causal Interpretation of Code Clone Detection

- Causal framework to interpret a model's clone predictions
  - Are similarities the real cause of clone prediction?
  - Counterfactual explanations help establish causes
  - Using human labels to create counterfactual clone pairs

# VisualStudio Annotator Tool for Clone and Code Labeling

J Clone13.java

```java
public class Clone13 {
/*
* Semantic clone benchmark
*  Source code are extracted from Stack Overflow
*  Stack overflow Question #:453018
*  Stack Overflow answer #:1647015
*  And Stack Overflow answer#:39232425
*/
public int countLines (String filename) throws IOException {
    LineNumberReader reader = new LineNumberReader (new FileReader (filename));
    int cnt = 0;
    String lineRead = "";
    while ((lineRead = reader.readLine ()) != null) {
    }
    cnt = reader.getLineNumber ();
    reader.close ();
    return cnt;
}

public static int countLines (File input) throws IOException {
    try (InputStream is = new FileInputStream (input)) {
        int count = 1;
        for (int aChar = 0;
        aChar != - 1; aChar = is.read ()) count += aChar == '\n' ? 1 : 0;
        return count;
    }
}

}
```

# Label Resolution

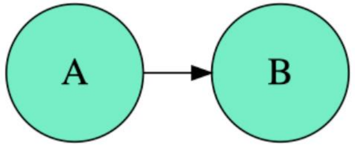
- Clone labels
  - Two human annotators and another that breaks ties
- Code labels
  - Two human annotation sets
  - Assign a label value (-2,-1,+1,+2) based on core differences, non-core differences, noncore similarities, and core similarities
  - Calculate average label values for overlapping label segments

# Mutation Strategy

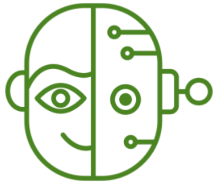
- Syntax-preserving mutations
- Mutation scope
  - Removing only core similarities or differences
  - Removing all core and noncore similarities or differences
- Mutate using AST parser
  - Remove a set of statements
  - Remove single statements
  - Remove parts of a statement

# Evaluation

# Evaluation Metrics




- Average Causal Effect (ACE) of removing similarities and differences

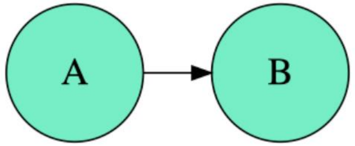


- Human-model code similarity intuition alignment

- Confounding Frequency

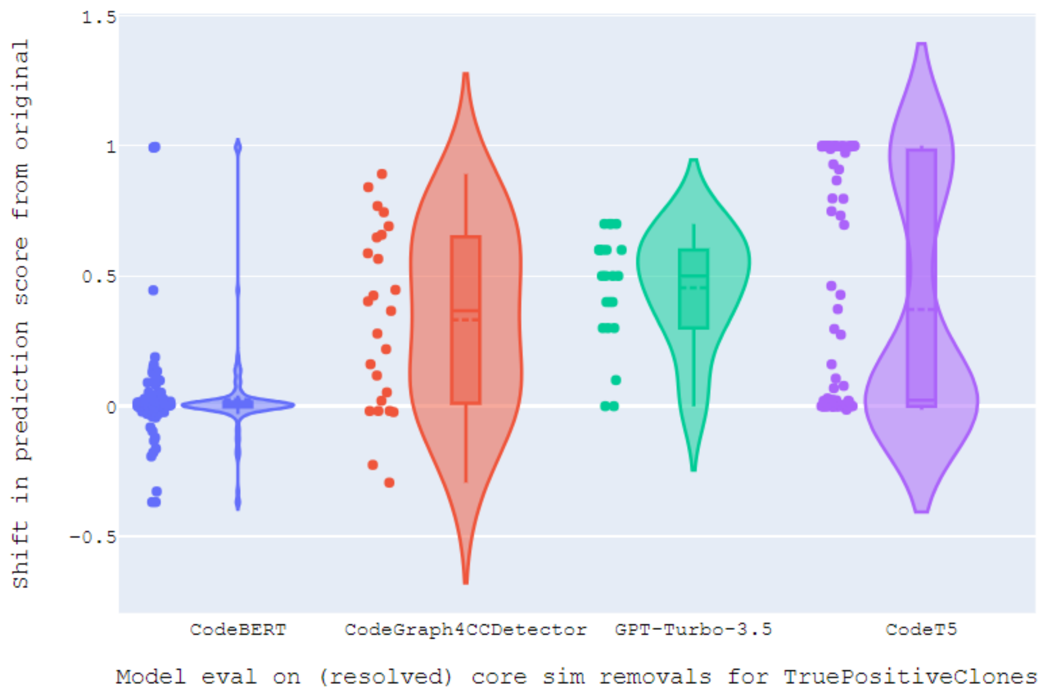
- Prediction Consistency

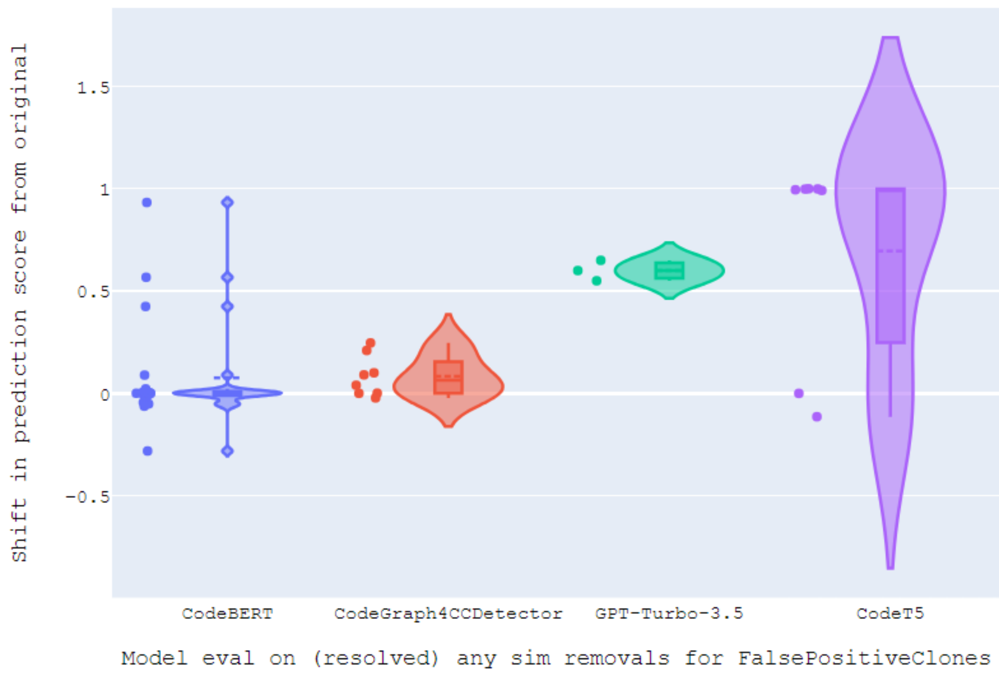
# Evaluation Metrics



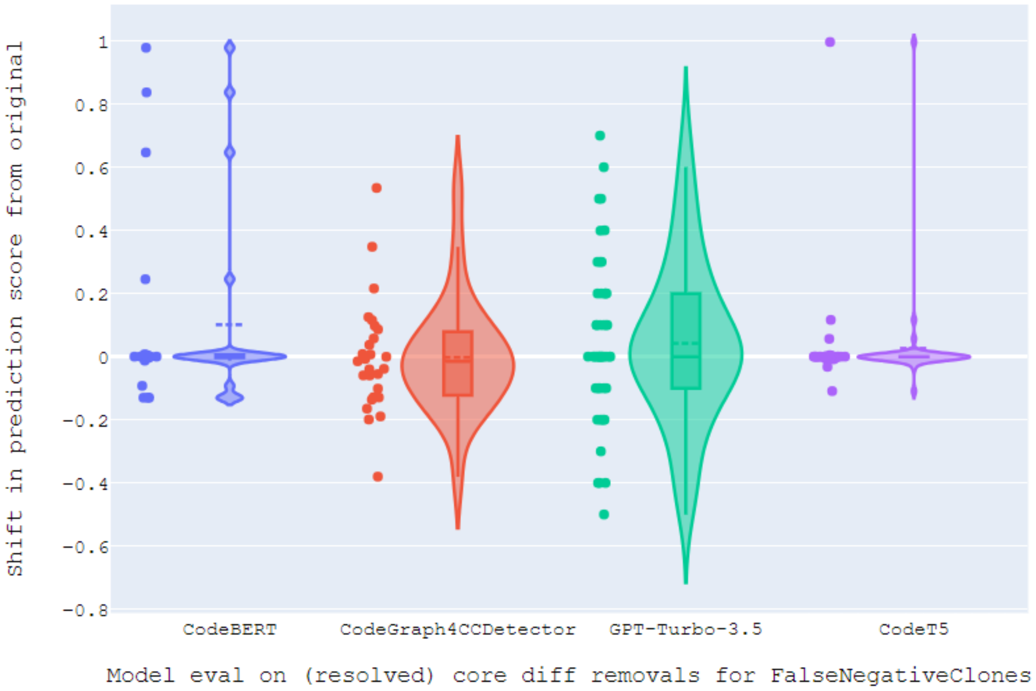

- Average Causal Effect (ACE) of removing similarities and differences
  - Measures the average of the model's prediction shifts on mutated clone pairs
  - ACE of similarities in TP and FP
    - *How much do human-identified similarities influence a model's predictions?*
  - ACE of differences in TN and FN
    - *How much do human-identified differences influence a model's predictions?*
  - A positive causal effect value > 0 means the model aligns with human intuition
  - A 0 or negative causal effect <0 means the model does not align with human intuition

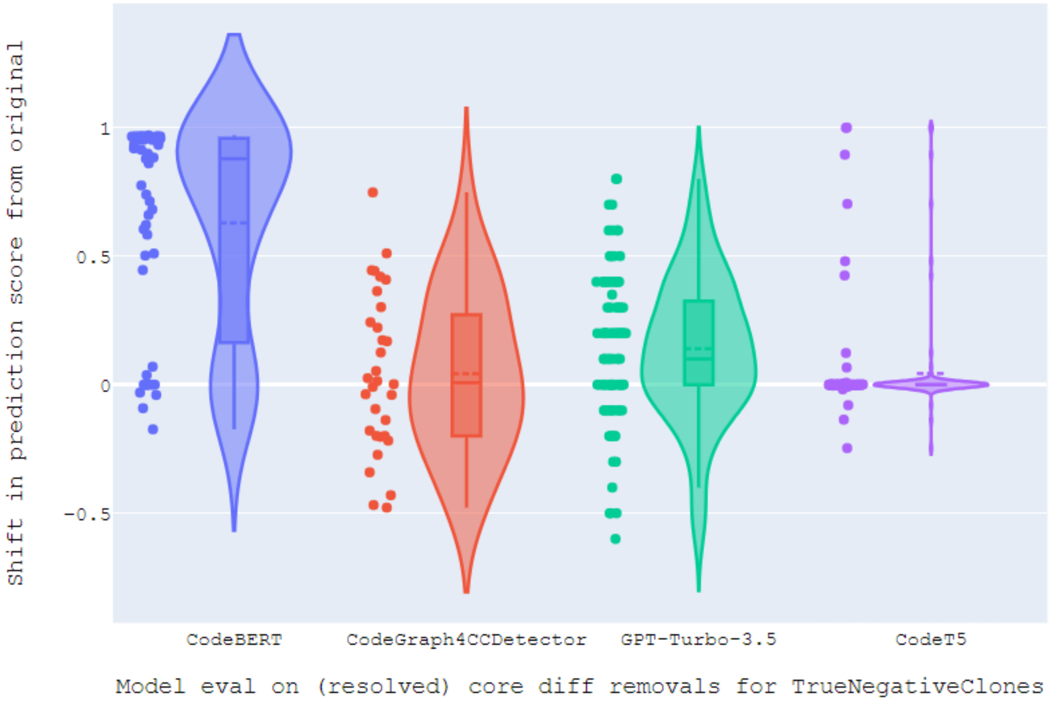
# Sensitivity of models' prediction scores to similarities removal



# Sensitivity of various models' prediction scores to differences removal



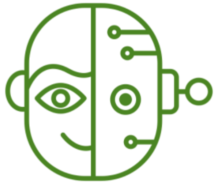



# Model Ranks

| ACE of sim TP cases | ACE of diff TN cases | ACE of sim FP cases | ACE of diff FN cases | Aggregate ACE |
|---|---|---|---|---|
| CodeT5 | GPT-Turbo-3.5 | CodeT5 | CodeBERT | GPT-Turbo-3.5 |
| GPT-Turbo-3.5 | CodeT5 | GPT-Turbo-3.5 | GPT-Turbo-3.5 | CodeT5 |
| CodeGraph4CCDetector | CodeGraph4CCDetector | CodeGraph4CCDetector | CodeT5 | CodeGraph4CCDetector |
| CodeBERT | CodeBERT | CodeBERT | CodeGraph4CCDetector | CodeBERT |

CodeBERT CodeT5 GPT-Turbo-3.5 CodeGraph4CCDetector

# Evaluation Metrics



- Human-model code similarity intuition alignment metric
  - Case H=1, M=1.
    - Is model's true clone prediction for a true clone pair based on human-identified code similarities?
    - *Human-model alignment percentage = average no. of prediction flips caused by similarities removal x 100*

| CodeBERT | CodeGraph4CCDetector | GPT-Turbo-3.5 | CodeT5 |
|---|---|---|---|
| 4.44% | 53.6% | **89.03%** | 49.13% |

# Evaluation Metrics

- Confounding Frequency Metric
  - Measures the number of times a model's *prediction gets flipped* on mutated clone pairs (for FP and FN cases)
  - For TP cases, if the *prediction doesn't flip* by removing similarities, we count it as the model being confounded.
  - *Confounding frequency = (no. of times flipped on FP + no. of time flipped on FN + no. of times didn't flip on TP) / (|FP+FN+TP|)*
  - Model with lower confounding frequency is better

| Mutation Scope | CodeBERT | CodeGraph4CCDetector | GPT-Turbo-3.5 | CodeT5 |
|---|---|---|---|---|
| Core | 0.77 | 0.2 | **0.09** | 0.45 |
| All | 0.73 | 0.2 | **0.1** | 0.28 |

# Evaluation Metrics

- Prediction Consistency
  - A model’s predictions across two runs on the same data should be the same
  - We calculate the Jaccard similarity between the predictions for a model on the same set of clone pairs

| CodeBERT | CodeGraph4CCDetector | GPT-Turbo-3.5 | CodeT5 |
| --- | --- | --- | --- |
| 1 | 1 | 0.75 | 1 |

# Model Ranks

| F1 Score | Human Alignment for TP sim | Least Confounded | Prediction Consistency |
|---|---|---|---|
| | | | |
| | | | |
| | | | |
| | | | |

CodeBERT CodeT5 GPT-Turbo-3.5 CodeGraph4CCDetector

# Model Ranks for Semantic Code Clone Detection

| Models | Gold stars |
|---|---|
| | |
| | |
| | |

CodeBERT CodeT5 GPT-Turbo-3.5 CodeGraph4CCDetector

# Future work

## Using automated techniques to generate the counterfactual samples

Using SOTA model intuitions

- First verify SHAP-based explanations of a SOTA model using human evaluation
- Perform SHAP-based mutations to get counterfactuals
- Evaluate other models on mutated counterfactual samples

## Using our labeled data to finetune ML models using contrastive learning

# Thanks!

- Questions and feedback and comments welcome!


- shamsaabid@smu.edu.sg
- https://shamsa-abid.github.io/