# Addressing Data Leakage in HumanEval using Combinatorial Test Design

**Jeremy S. Bradbury**, **Riddhi More**

Software Engineering & Education Research Lab
Ontario Tech University, Oshawa, Canada

http://www.seerlab.ca

SEER LAB | Ontario Tech University

ICST 2025

# Benchmarking & LLMs

- **Benchmarks** are standardized evaluation tools that enable systematic comparison of different approaches that solve the same problem or task [1]
- LLM benchmarks that have experienced **data leakage** (when benchmark data leaks into the training data) will likely exhibit inflated benchmark evaluation scores which can misrepresent their ability to address the underlying task [2]


**[1]** S. Sim, S. Easterbrook, and R. Holt, "Using benchmarking to advance research: a challenge to software engineering," in Proc. of the 25th International Conference on Software Engineering (ICSE 2003), May 2003, pp. 74–83.

**[2]** O. Sainz, J. Campos, I. Garc´ıa-Ferrero, J. Etxaniz, O. L. de Lacalle,and E. Agirre, "NLP evaluation in trouble: On the need to measure LLM data contamination for each benchmark," in Findings of the Association for Computational Linguistics (EMNLP), Dec. 2023, pp. 10 776–10 787.

# Benchmarking & LLMs: Best Practices

- Benchmark **Construction**
  - Tasks can not originate from sources that are part of LLM training data
    (1) hand-craft tasks (e.g., HumanEval)
    (2) tasks selected from private data sets
- Benchmark **Operation**
  - All benchmark data must be actively excluded from future LLM training data sets.

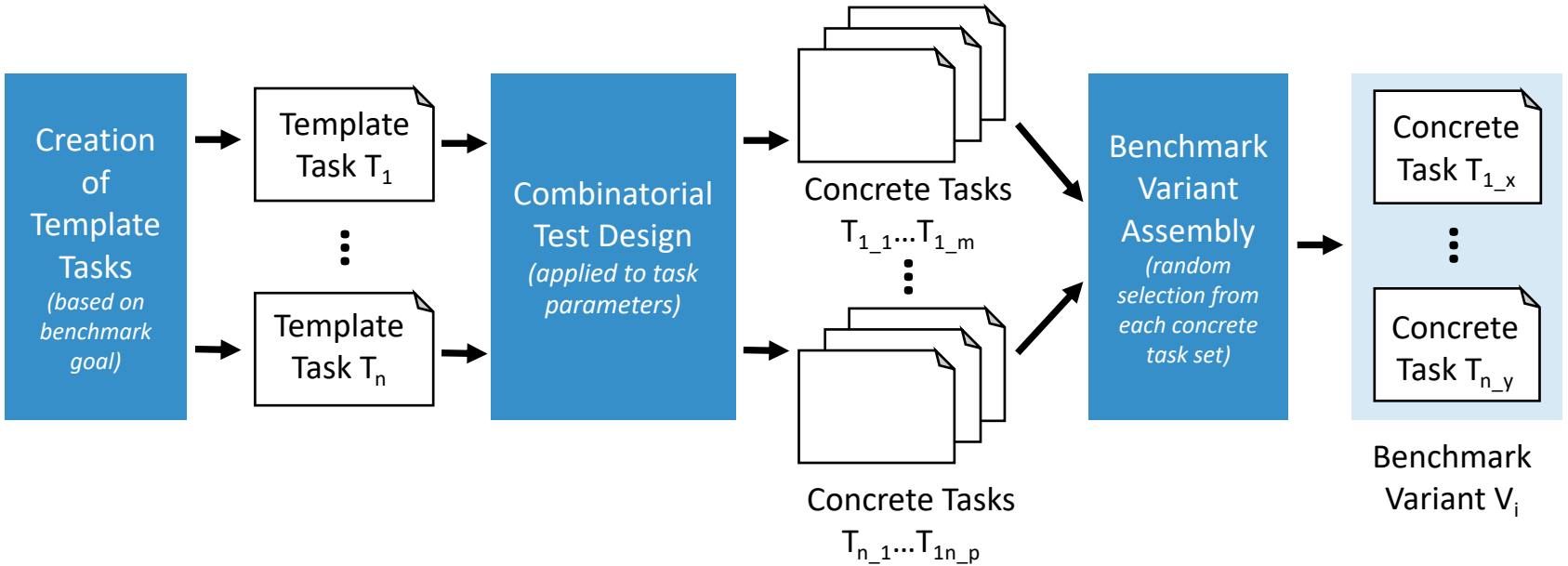
# HumanEval Benchmark

- Developed in 2021 at OpenAI [3]
- Assesses LLMs with respect to **program generation**
- Contains 164 hand crafted tasks
- Example:

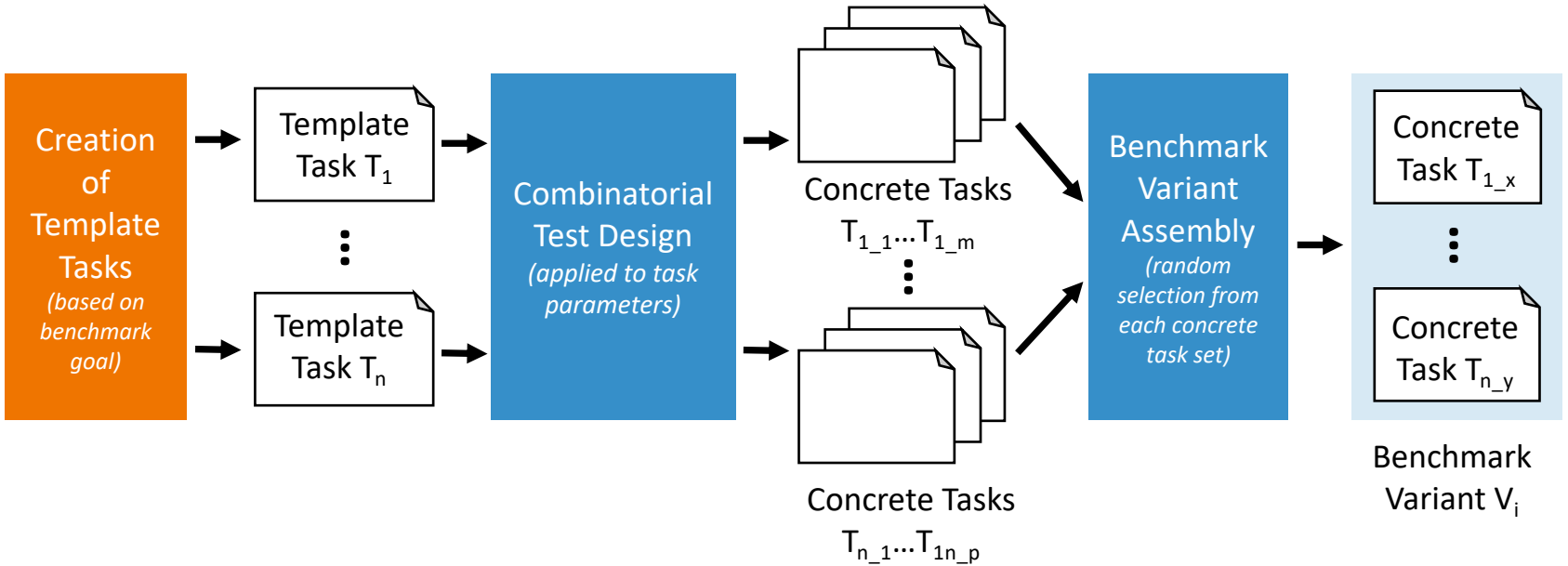
```
Given a list of numbers and a
threshold value, determine if any two
values are closer than the threshold.
...
```

**[3]** M. Chen, et al. “Evaluating large language models trained on code,” 2021. [Online]. Available: https://arxiv.org/abs/2107.03374

# Our Benchmark Construction Process



# Our Benchmark Construction Process

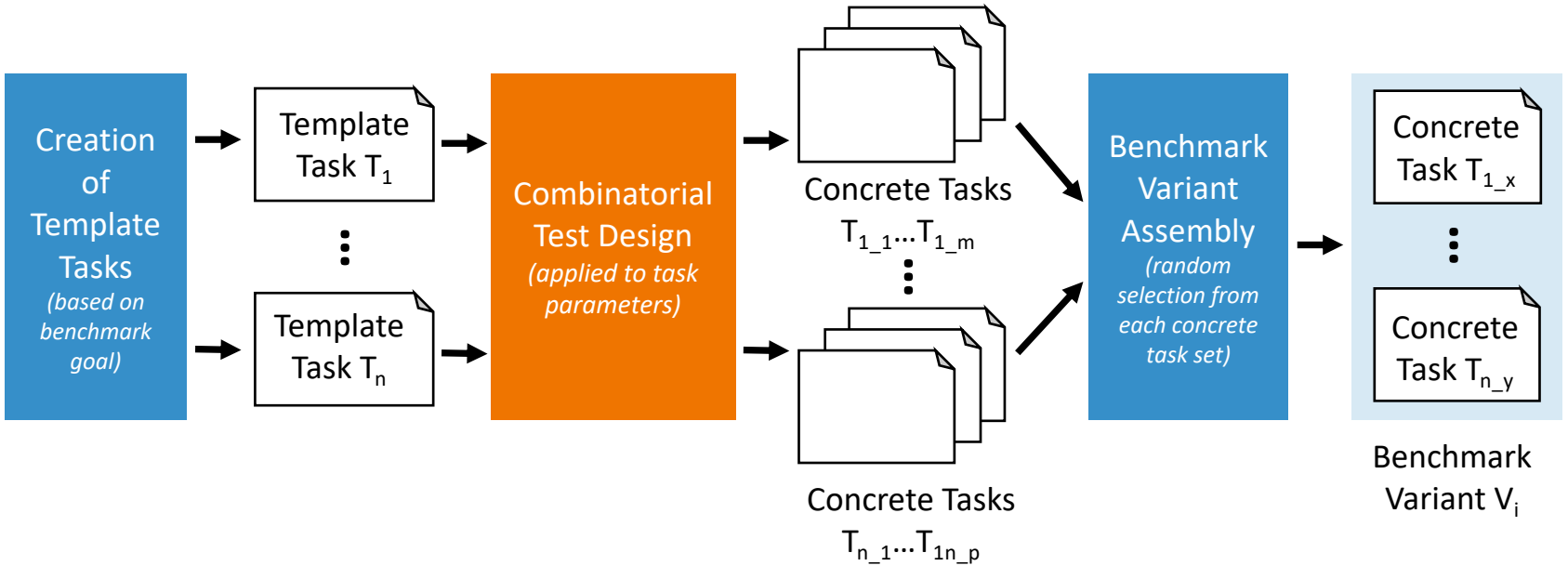


# Creation of Template Tasks

Given a list of numbers and a threshold value, determine if any two values are closer than the threshold.

Given a list of **<input_type>** and **<threshold_descriptor>**, check if any two **<value_descriptor>** are closer than the given **<threshold_descriptor>**.
Where:
- **<input_type>**: numbers, float values, measurements
- **<threshold_descriptor>**: threshold, minimum distance, tolerance
- **<value_descriptor>**: values, elements, data points

# Our Benchmark Construction Process



# Creation of Template Tasks

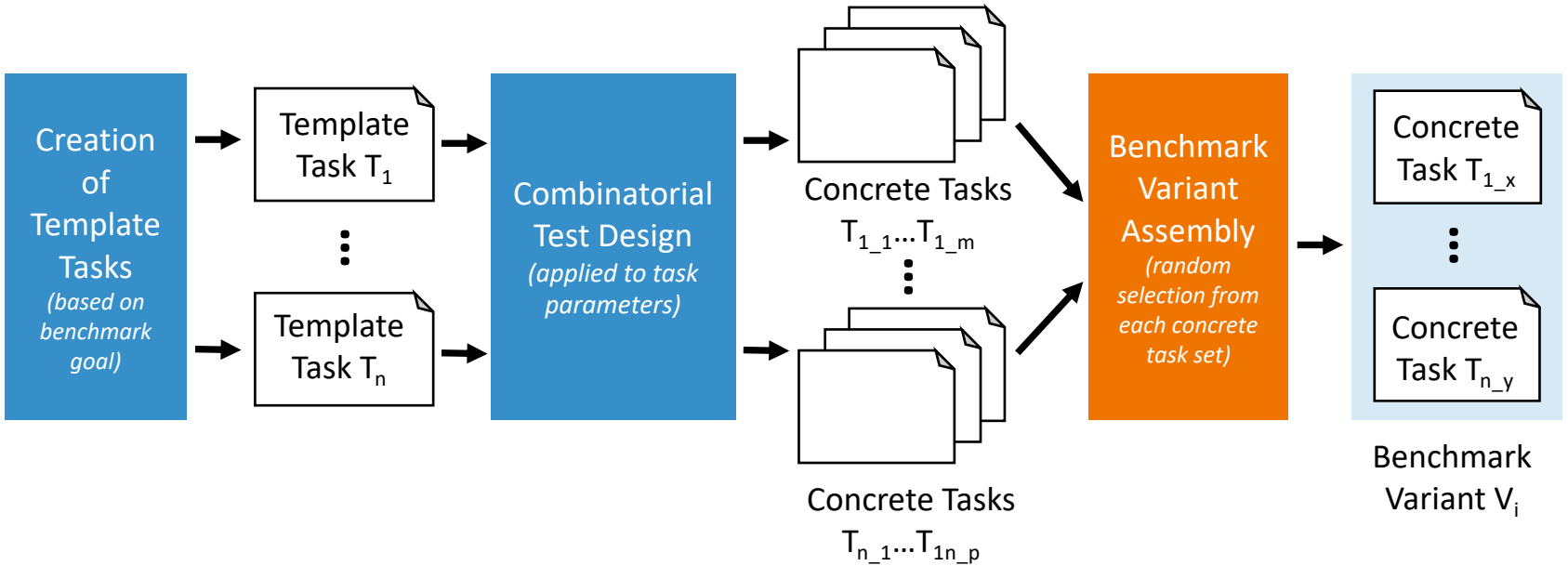
Given a list of **<input_type>** and **<threshold_descriptor>**, check if any two **<value_descriptor>** are closer than the given **<threshold_descriptor>**.
...

Given a list of **numbers** and a **threshold**, check if any two **values** are closer than the given **threshold**

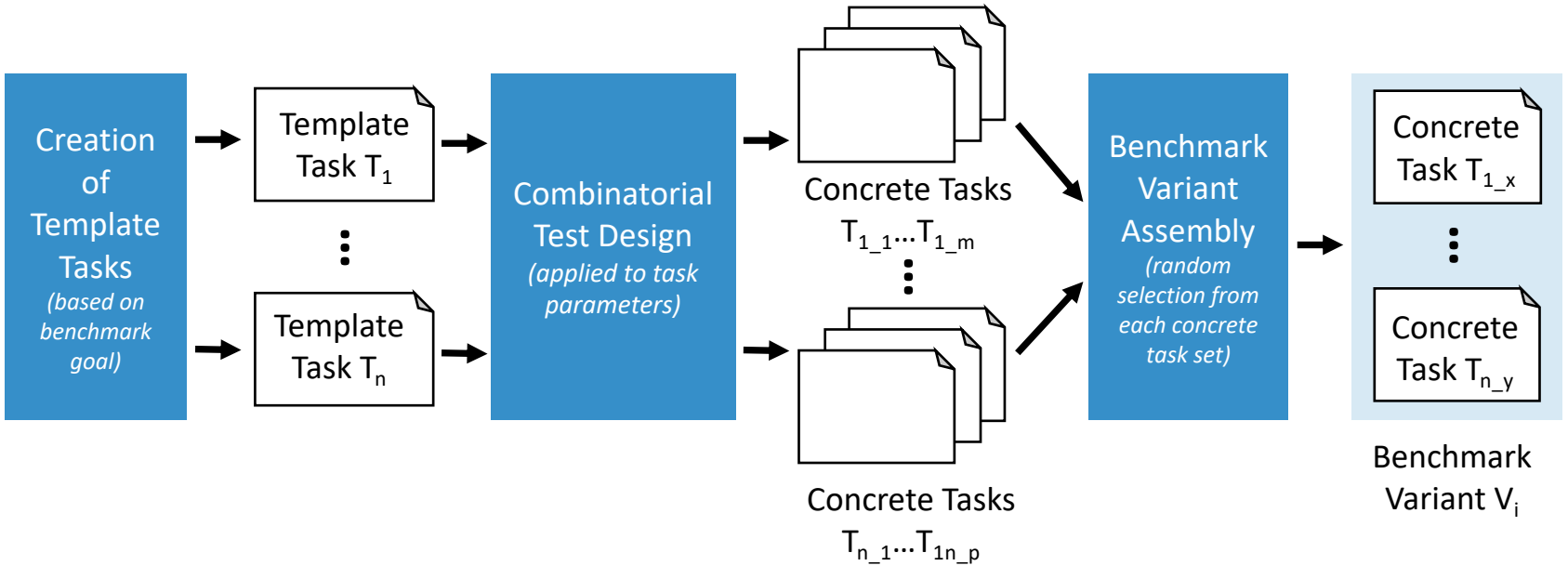
...

Given a list of **measurements** and a **minimum distance**, check if any two **data points** are closer than given **minimum distance**

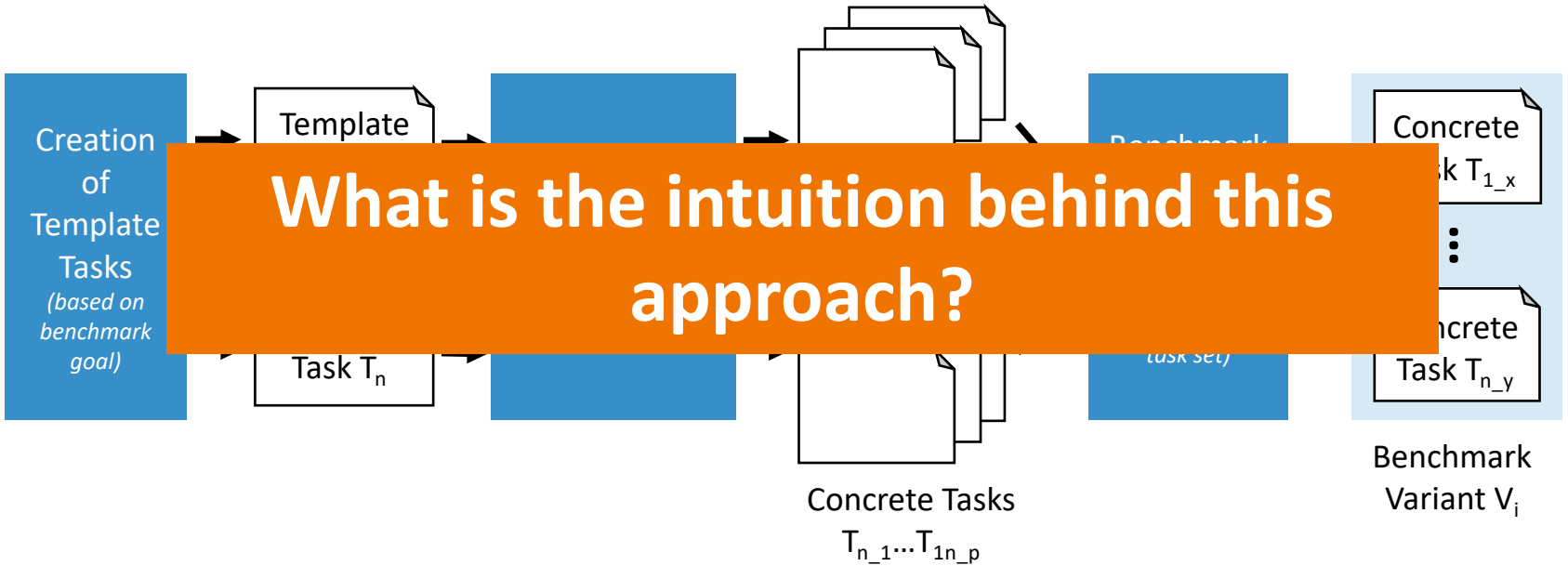
# Our Benchmark Construction Process



# Our Benchmark Construction Process

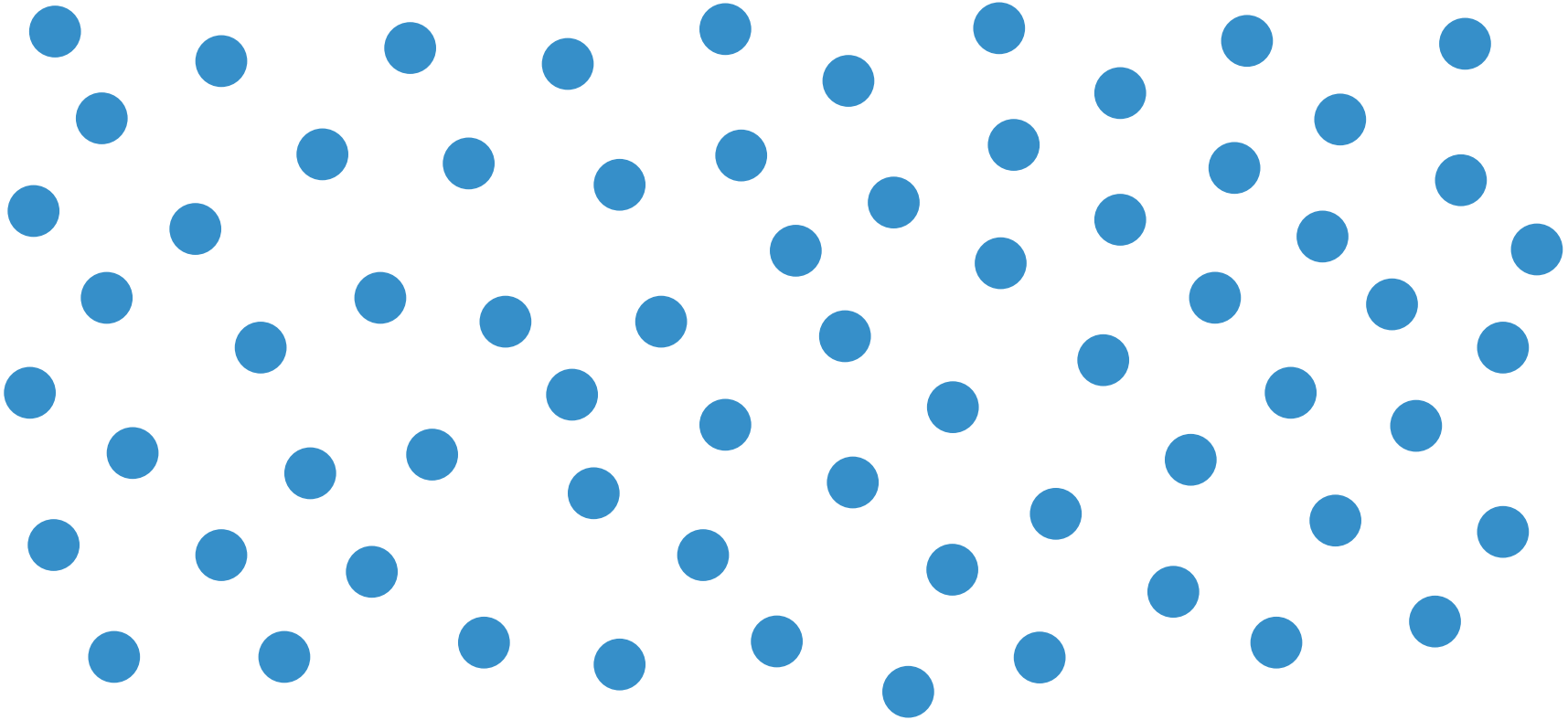


# Our Benchmark Construction Process

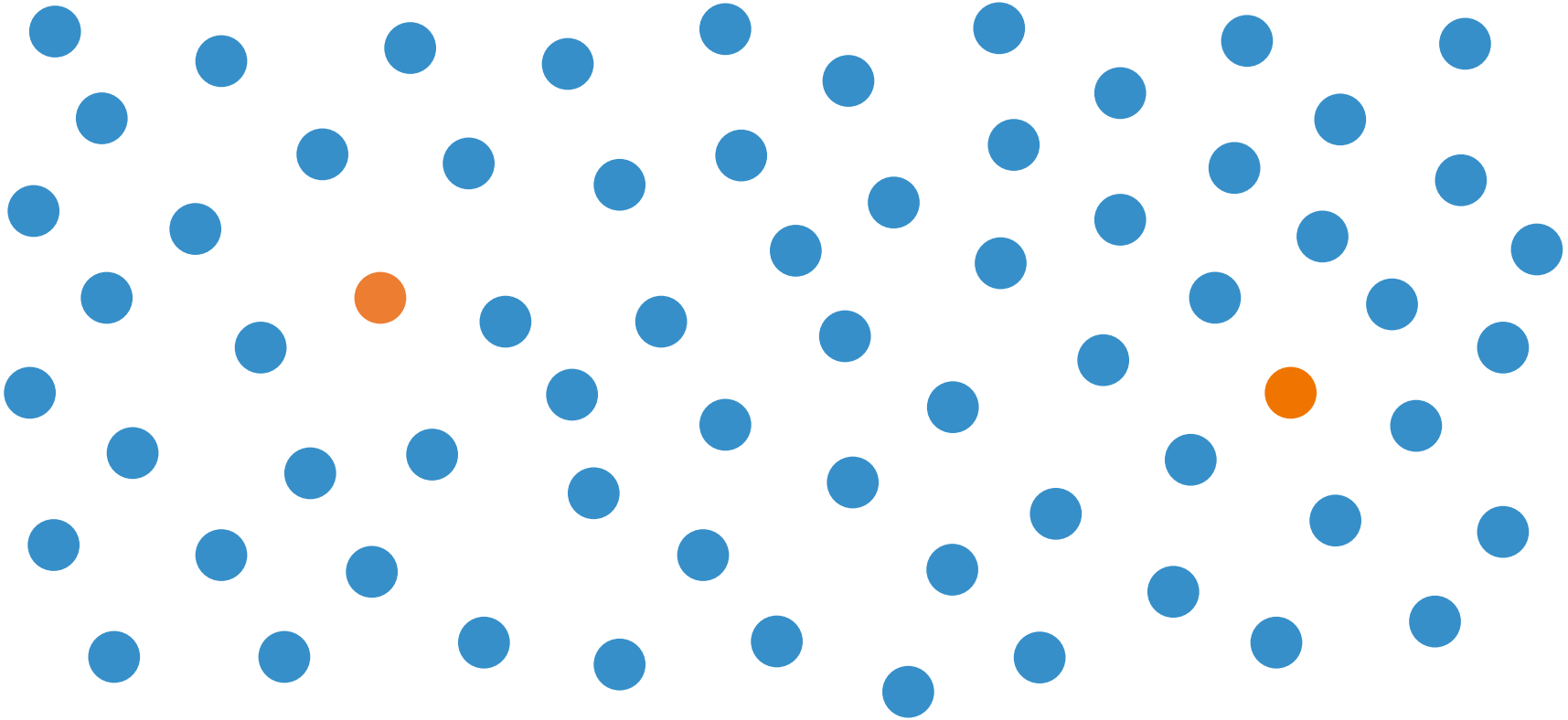



**What is the intuition behind this approach?**

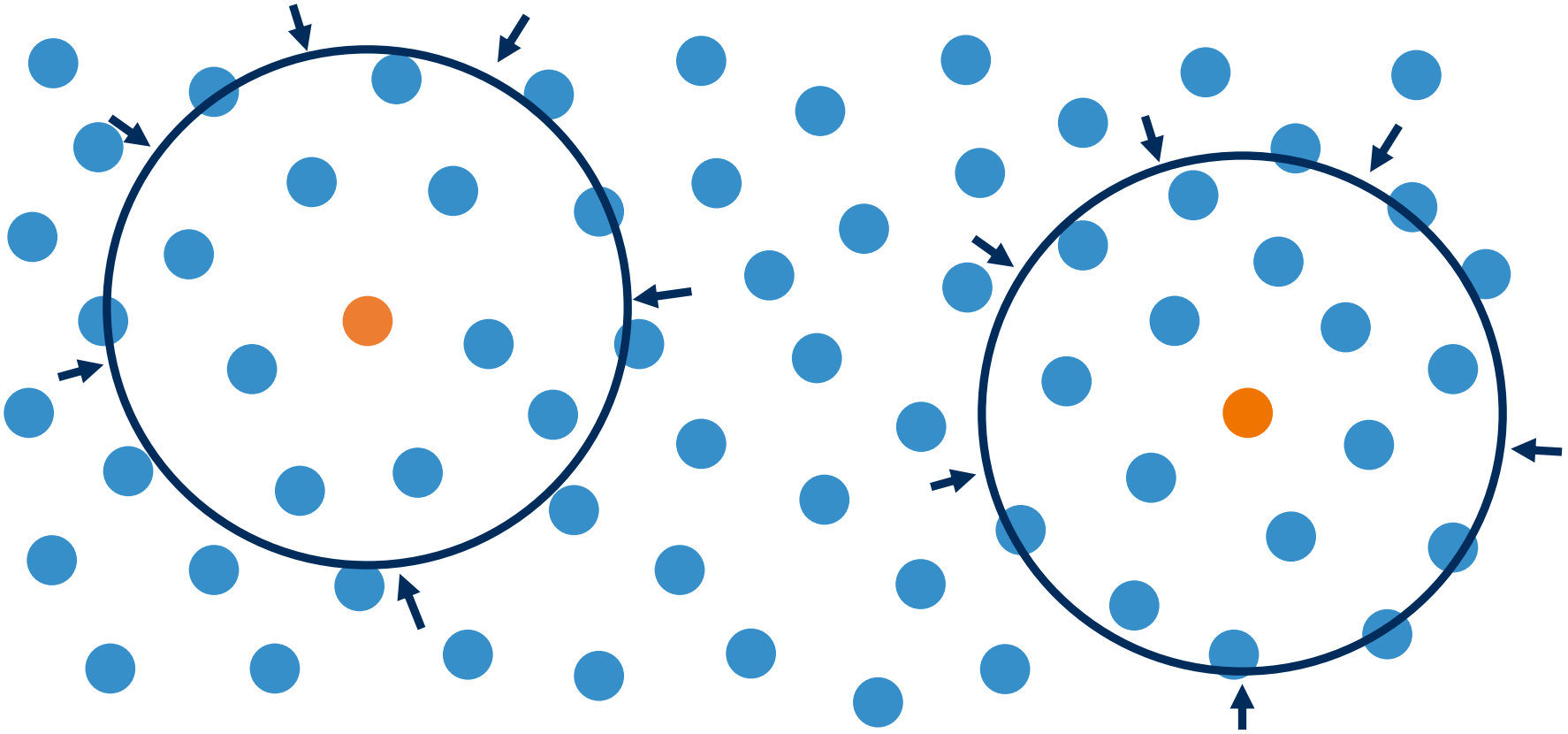
# Program Generation Problem Space



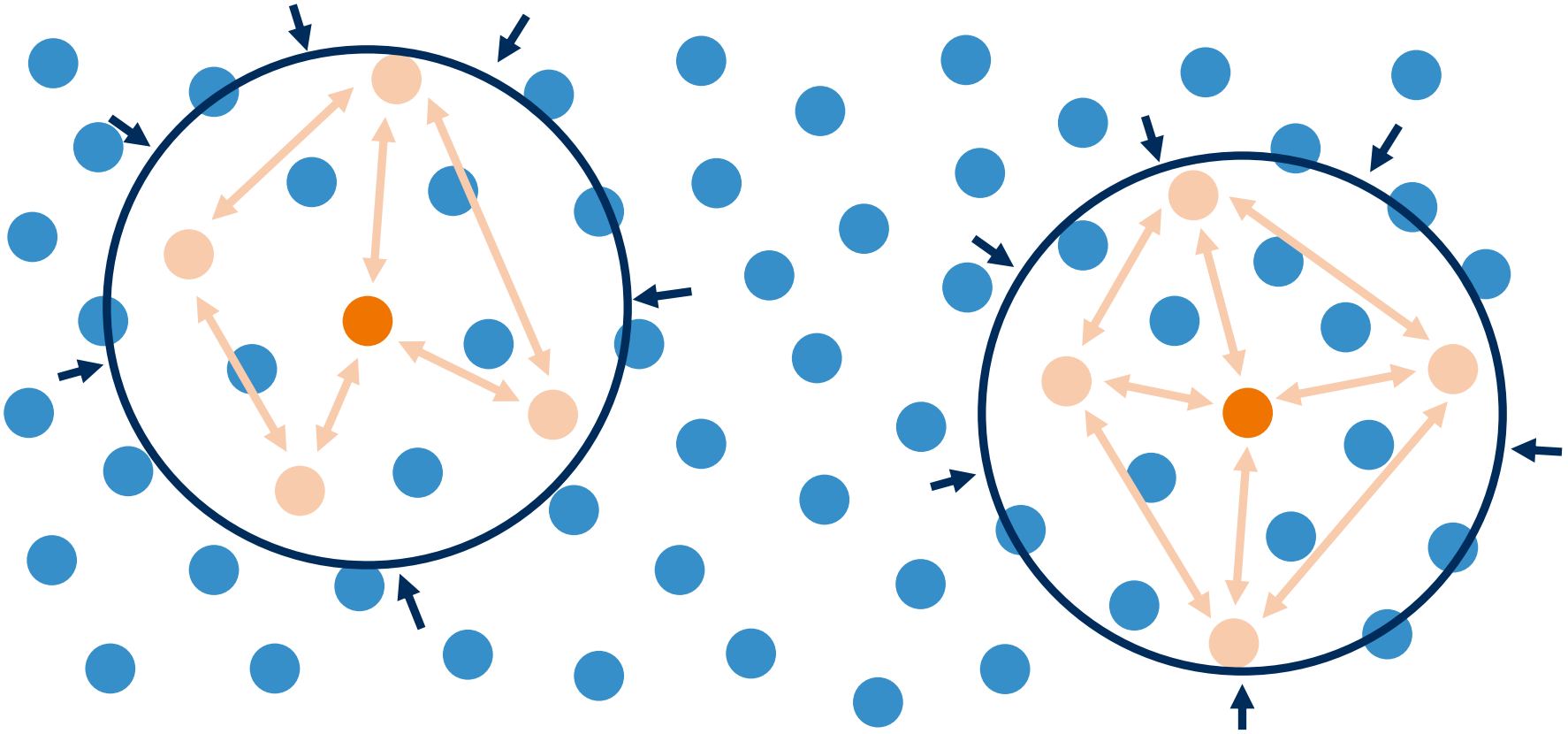
# HumanEval Program Generation Tasks



# HumanEval_T Template Tasks



# HumanEval_T Concrete Tasks



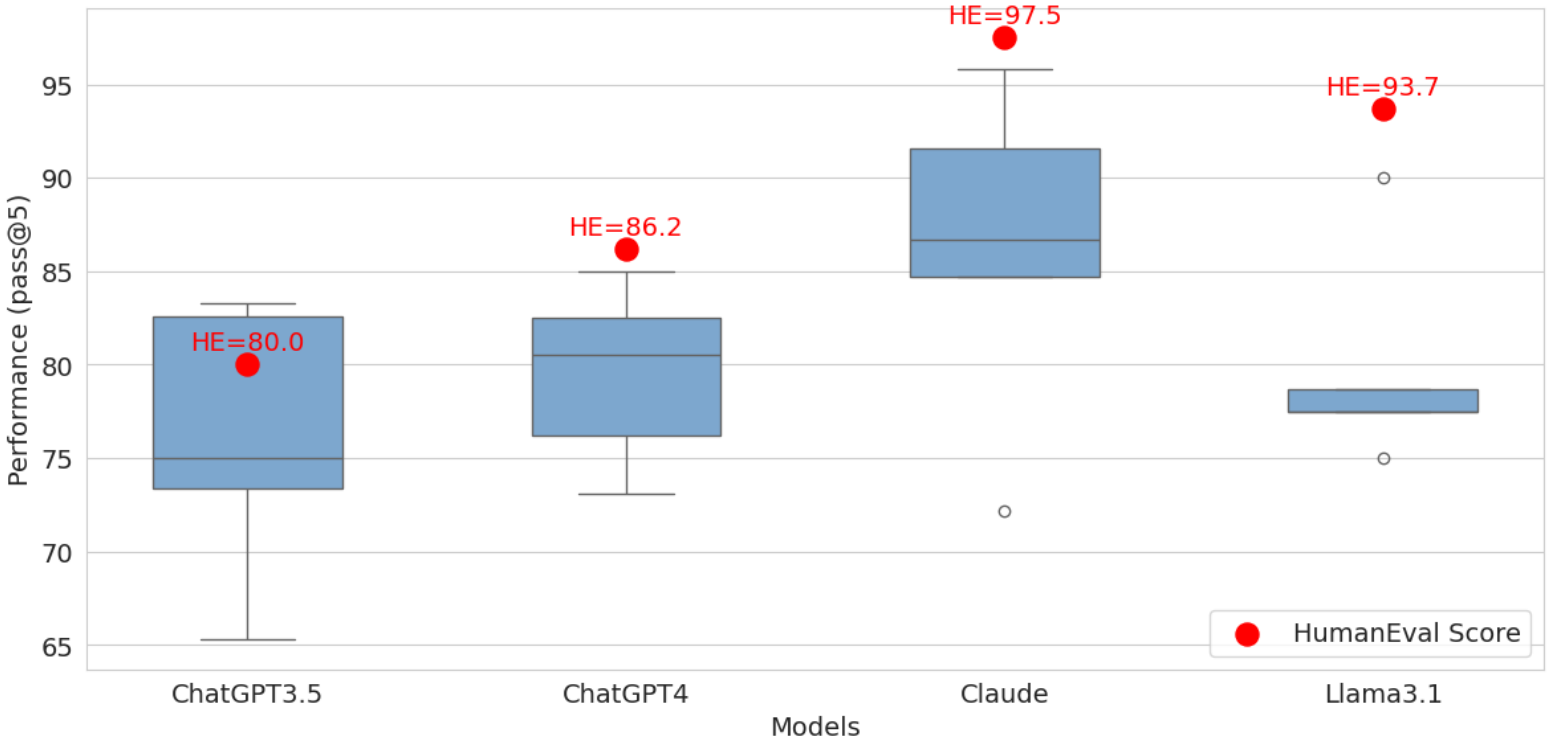
# Experimental Design

- Randomly selected 10 **HumanEval** tasks as a baseline
- Created corresponding template tasks by hand
- Generated five unique **HumanEval_T** benchmark variants
- Evaluated the HumanEval tasks and the HumanEval_T variants on four LLMs: **GPT3**, **GPT4**, **Claude**, **Llama3.1**

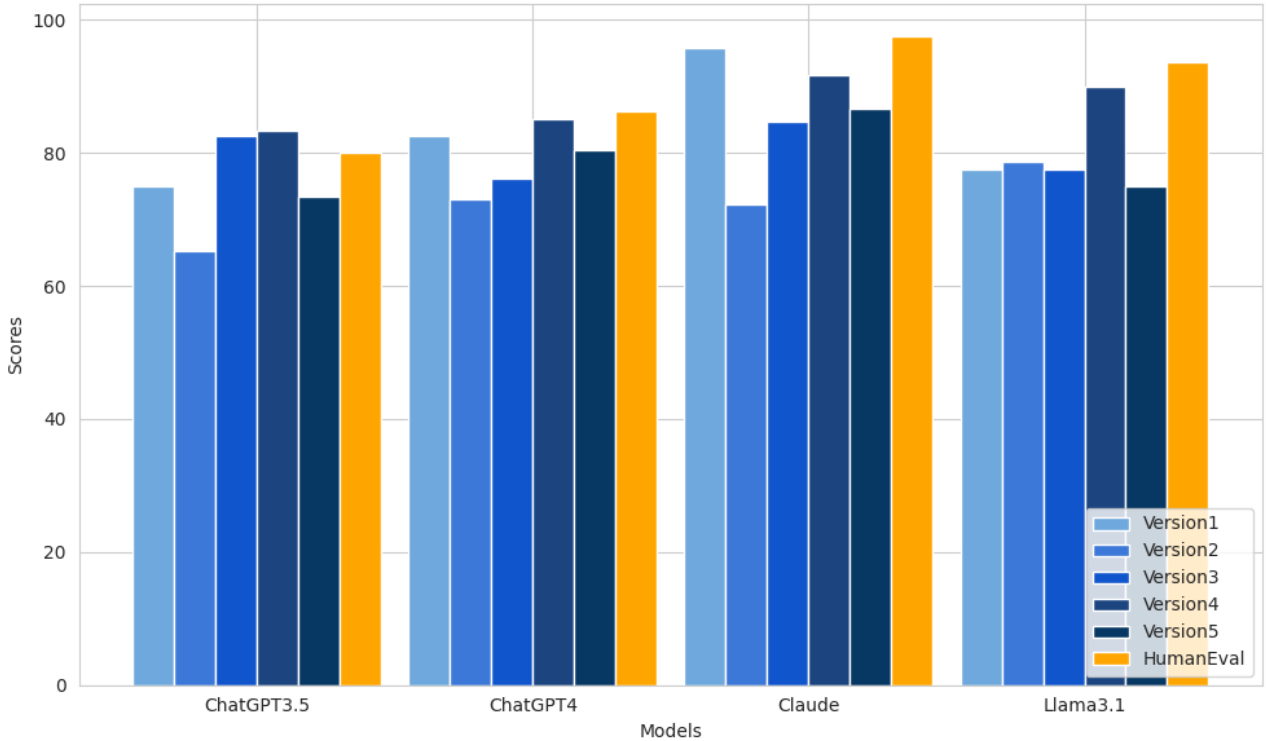
**RQ1:** Is there evidence of data leakage in HumanEval?

**RQ2:** Do concrete versions of the same template task in HumanEval_T produce similar results?

# RQ1: Is there evidence of data leakage in HumanEval?



# RQ2: Do concrete versions of the same template task in HumanEval_T produce similar results?



# Conclusions

- We observed ***evidence of data leakage*** in the **HumanEval** benchmark
- Our benchmark construction process systematically creates **HumanEval_T** benchmark variants that are ***robust to data leakage*** from HumanEval
- Low variance in **HumanEval_T** benchmark variant performance is a positive indicator that they ***are similar in difficulty***

# Conclusions

- We observed ***evidence of data leakage*** in the **HumanEval** benchmark
- Our benchmark construction process systematically creates **HumanEval_T** benchmark variants that are ***robust to data leakage*** from HumanEval
- Low variance in **HumanEval_T** benchmark variant performance is a positive indicator that they ***are similar in difficulty***

***Open Question:*** *How do we evaluate concrete tasks from the same template tasks to assess their suitability as equivalent tasks for assessment?*

# Addressing Data Leakage in HumanEval using Combinatorial Test Design

**Jeremy S. Bradbury**, **Riddhi More**

Software Engineering & Education Research Lab
Ontario Tech University, Oshawa, Canada

http://www.seerlab.ca

SEER LAB | Ontario Tech University

ICST 2025