5. Evaluating Code LMs: Function-Level Code Generation

Natural Language to Code (NL2Code)

- Remove specific characters from a string in python

Yin, P., Deng, B., Chen, E., Vasilescu, B., & Neubig, G. (2018, May). Learning to mine aligned code and natural language pairs from stack overflow. In Proceedings of the 15th international conference on mining software repositories (pp. 476-486).

Natural Language to Code (NL2Code)

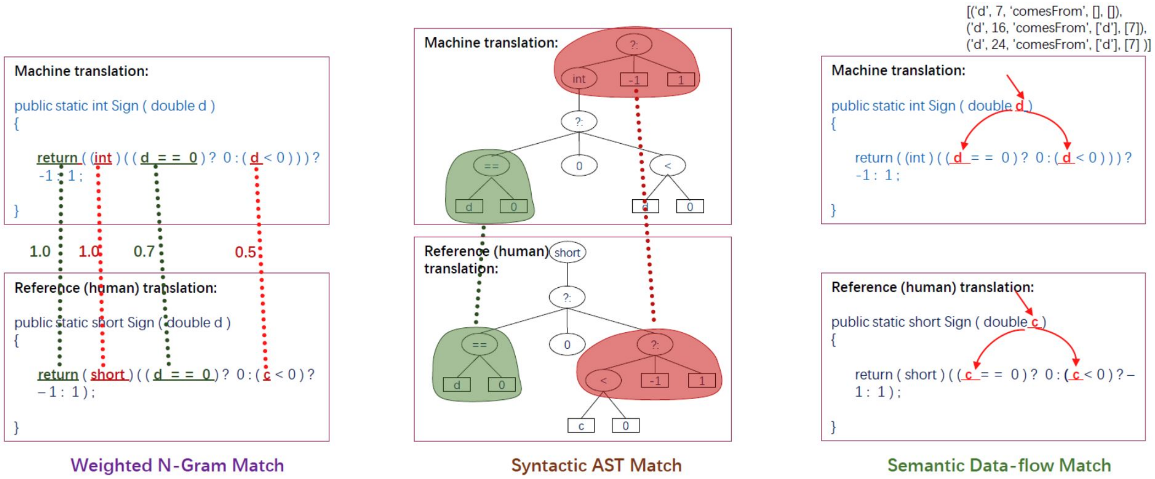
- Remove specific characters from a string in python

```
string.replace('1', '')

line = line.translate(None, '!@#$')

line = re.sub('[!@#$]', '', line)
```

Yin, P., Deng, B., Chen, E., Vasilescu, B., & Neubig, G. (2018, May). Learning to mine aligned code and natural language pairs from stack overflow. In Proceedings of the 15th international conference on mining software repositories (pp. 476-486).

NL2Code: Evaluation Metrics

- Lexical
  - ❑ ***CodeBLEU***: weighted combination of the original BLEU, the weighted n-gram match, the syntactic AST match, and the semantic data-flow match



Machine translation:

```
public static int Sign ( double d )
{
    return ( (int ) ( ( d = = 0 ) ? 0 : ( d < 0 ) ) ) ?
    -1 : 1 ;
}
```

Reference (human) translation:

```
public static short Sign ( double d )
{
    return ( short ) ( ( d = = 0 ) ? 0 : ( c < 0 ) ?
    − 1 : 1 ) ;
}
```

Weighted N-Gram Match

Machine translation:

Reference (human) translation:

Syntactic AST Match

[('d', 7, 'comesFrom', [], []),
('d', 16, 'comesFrom', ['d'], [7]),
('d', 24, 'comesFrom', ['d'], [7])]

Machine translation:

```
public static int Sign ( double d )
{
    return ( (int ) ( ( d = = 0 ) ? 0 : ( d < 0 ) ) ) ?
    -1 : 1 ;
}
```

Reference (human) translation:

```
public static short Sign ( double c )
{
    return ( short ) ( ( c = = 0 ) ? 0 : ( c < 0 ) ? −
    1 : 1 ) ;
}
```

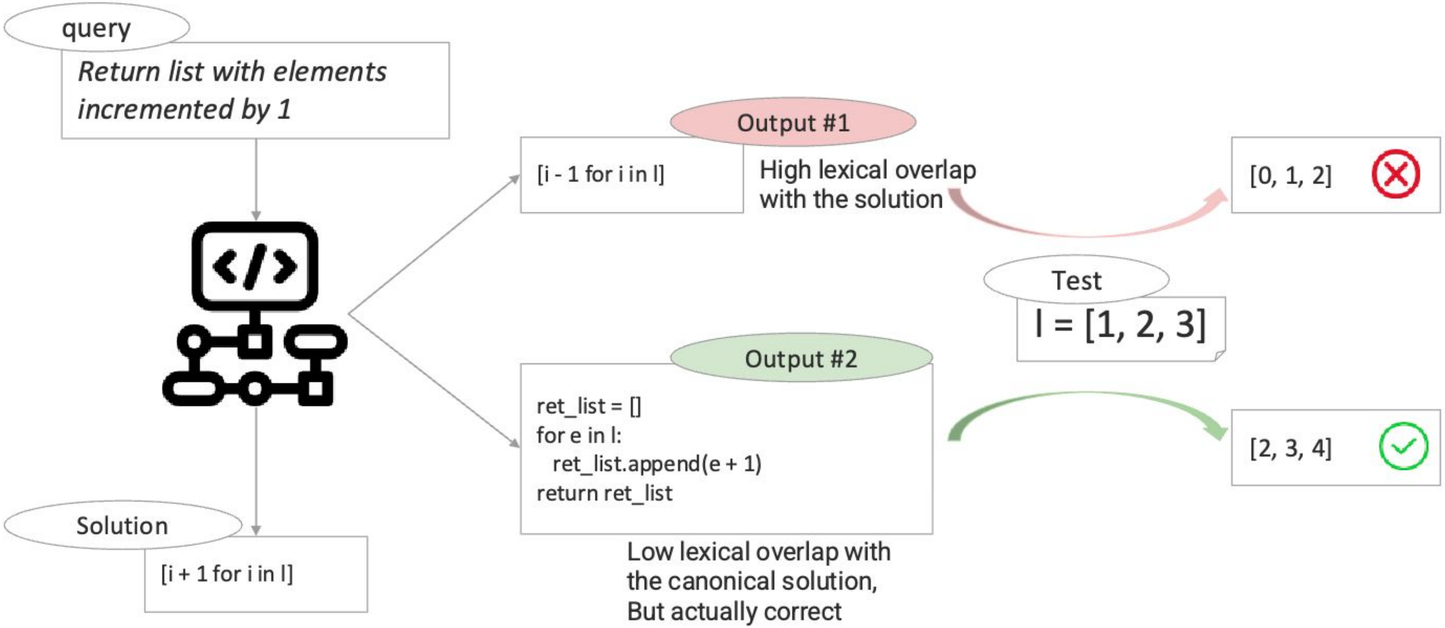
Semantic Data-flow Match

CodeBLEU $= \alpha \cdot$ N − Gram Match (BLEU) $+ \beta \cdot$ Weighted N-Gram Match $+ \gamma \cdot$ Syntactic AST Match $+ \delta \cdot$ Semantic Data-flow Match

Ren, S., Guo, D., Lu, S., Zhou, L., Liu, S., Tang, D., ... & Ma, S. (2020). Codebleu: a method for automatic evaluation of code synthesis. arXiv preprint arXiv:2009.10297.

NL2Code: Evaluation Metrics

➢ Test Case Execution



query

Return list with elements incremented by 1

Output #1

```
[i - 1 for i in l]
```

High lexical overlap with the solution

[0, 1, 2] ✗

Test

l = [1, 2, 3]

Output #2

```
ret_list = []
for e in l:
  ret_list.append(e + 1)
return ret_list
```

[2, 3, 4] ✓

Low lexical overlap with the canonical solution,
But actually correct

Solution

```
[i + 1 for i in l]
```

NL2Code: Evaluation Metrics

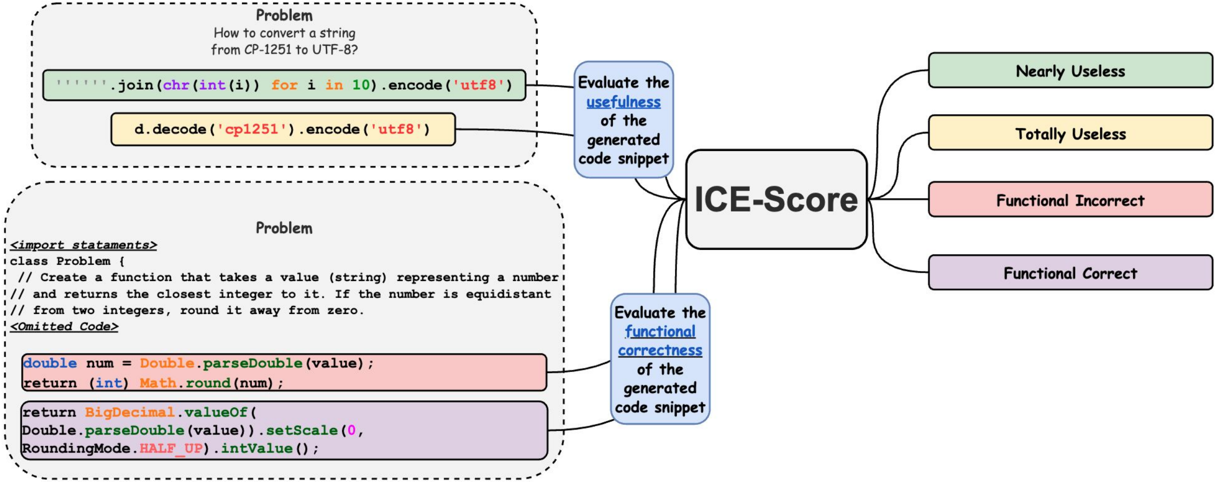
- Test Case Execution
  - *pass@k*: For each problem, sample k candidate solutions. If at least one of the k passes all test cases, count it as a success. Then compute the fraction of problems for which at least one sampled solution passes.

$$\text{Pass@}k = \mathbb{E}\left(1 - \frac{\binom{n-c}{k}}{\binom{n}{k}}\right)$$

Chen, M., Tworek, J., Jun, H., Yuan, Q., Pinto, H. P. D. O., Kaplan, J., ... & Zaremba, W. (2021). Evaluating large language models trained on code. arXiv preprint arXiv:2107.03374.

NL2Code: Evaluation Metrics

- ➢ Language model as a judge
  - ❑ Score rating based on the definition of an evaluation criterion



Problem

How to convert a string
from CP-1251 to UTF-8?

```
''''''.join(chr(int(i)) for i in 10).encode('utf8')
```

```
d.decode('cp1251').encode('utf8')
```

Problem

```
<import stataments>
class Problem {
 // Create a function that takes a value (string) representing a number
// and returns the closest integer to it. If the number is equidistant
// from two integers, round it away from zero.
<Omitted Code>
```

```
double num = Double.parseDouble(value);
return (int) Math.round(num);
```

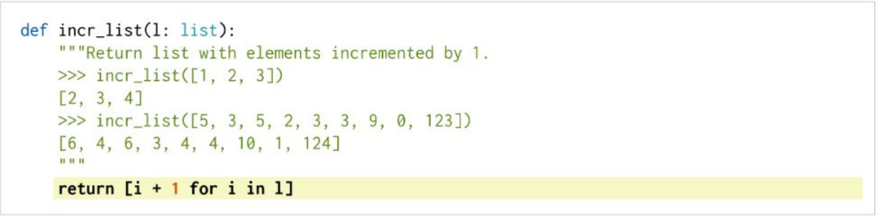
```
return BigDecimal.valueOf(
Double.parseDouble(value)).setScale(0,
RoundingMode.HALF_UP).intValue();
```

Evaluate the usefulness of the generated code snippet
Evaluate the functional correctness of the generated code snippet
ICE-Score
Nearly Useless
Totally Useless
Functional Incorrect
Functional Correct

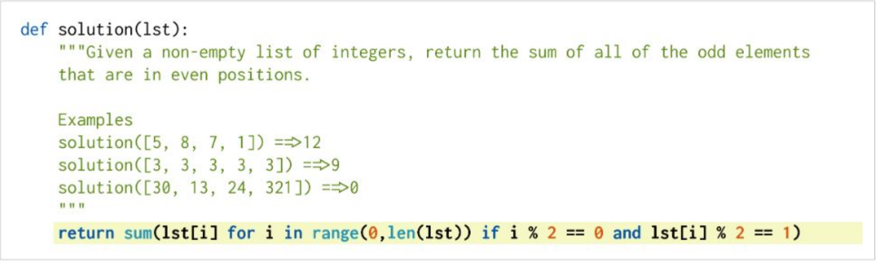
Zhuo, T. Y. (2024, March). ICE-Score: Instructing Large Language Models to Evaluate Code.
In Findings of the Association for Computational Linguistics: EACL 2024 (pp. 2232-2242).

NL2Code: HumanEval

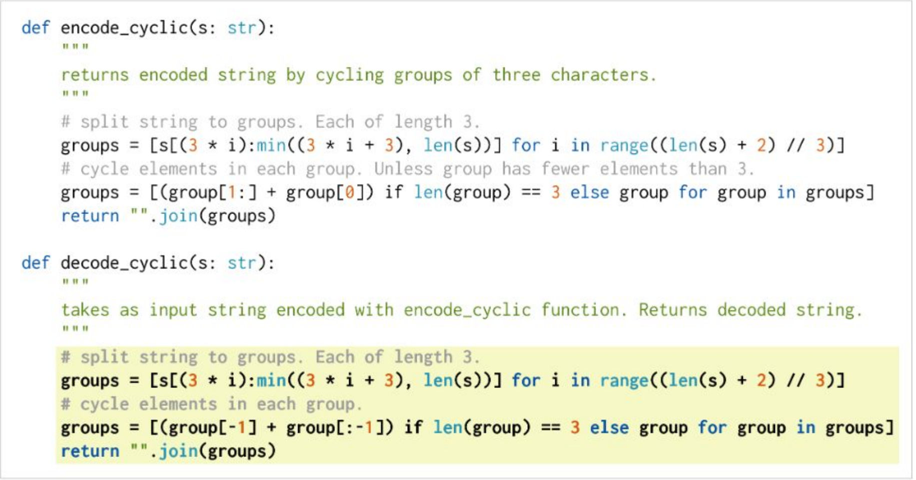
- 164 handwritten examples, by authors of the paper
- Why Handwritten? Avoid the data contamination in training.



```python
def incr_list(l: list):
    """Return list with elements incremented by 1.
    >>> incr_list([1, 2, 3])
    [2, 3, 4]
    >>> incr_list([5, 3, 5, 2, 3, 3, 9, 0, 123])
    [6, 4, 6, 3, 4, 4, 10, 1, 124]
    """
    return [i + 1 for i in l]
```



```python
def solution(lst):
    """Given a non-empty list of integers, return the sum of all of the odd elements
    that are in even positions.

    Examples
    solution([5, 8, 7, 1]) ==>12
    solution([3, 3, 3, 3, 3]) ==>9
    solution([30, 13, 24, 321]) ==>0
    """
    return sum(lst[i] for i in range(0,len(lst)) if i % 2 == 0 and lst[i] % 2 == 1)
```

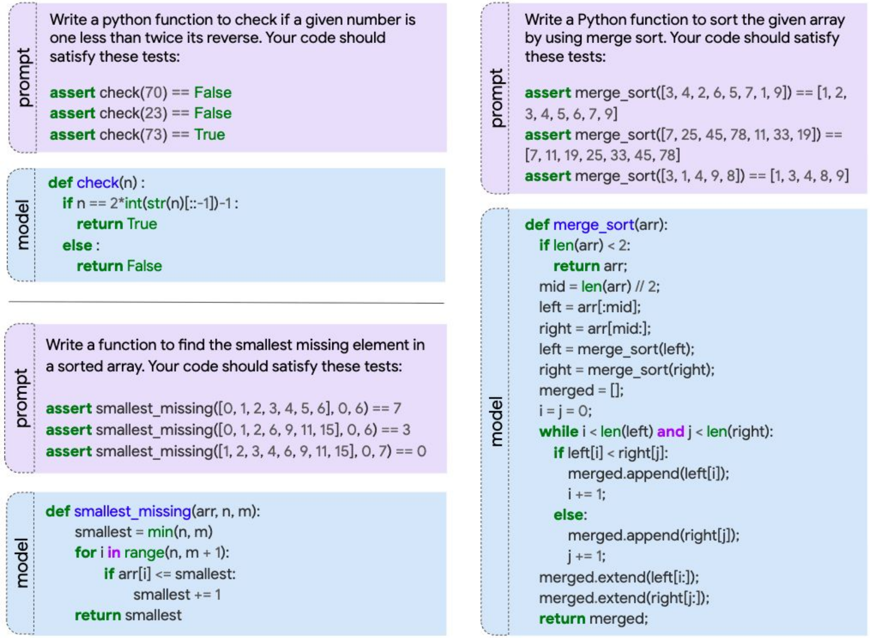


```python
def encode_cyclic(s: str):
    """
    returns encoded string by cycling groups of three characters.
    """
    # split string to groups. Each of length 3.
    groups = [s[(3 * i):min((3 * i + 3), len(s))] for i in range((len(s) + 2) // 3)]
    # cycle elements in each group. Unless group has fewer elements than 3.
    groups = [(group[1:] + group[0]) if len(group) == 3 else group for group in groups]
    return "".join(groups)

def decode_cyclic(s: str):
    """
    takes as input string encoded with encode_cyclic function. Returns decoded string.
    """
    # split string to groups. Each of length 3.
    groups = [s[(3 * i):min((3 * i + 3), len(s))] for i in range((len(s) + 2) // 3)]
    # cycle elements in each group.
    groups = [(group[-1] + group[:-1]) if len(group) == 3 else group for group in groups]
    return "".join(groups)
```

Chen, M., Tworek, J., Jun, H., Yuan, Q., Pinto, H. P. D. O., Kaplan, J., ... & Zaremba, W. (2021). Evaluating large language models trained on code. arXiv preprint arXiv:2107.03374.

NL2Code: MBPP

- ➢ Similar to HumanEval, but a bit easier.
- ➢ 974 short Python problems, written by crowdworkers.



Write a python function to check if a given number is one less than twice its reverse. Your code should satisfy these tests:

```
assert check(70) == False
assert check(23) == False
assert check(73) == True
```

```
def check(n) :
  if n == 2*int(str(n)[::-1])-1 :
    return True
  else :
    return False
```

Write a function to find the smallest missing element in a sorted array. Your code should satisfy these tests:

```
assert smallest_missing([0, 1, 2, 3, 4, 5, 6], 0, 6) == 7
assert smallest_missing([0, 1, 2, 6, 9, 11, 15], 0, 6) == 3
assert smallest_missing([1, 2, 3, 4, 6, 9, 11, 15], 0, 7) == 0
```

```
def smallest_missing(arr, n, m):
    smallest = min(n, m)
    for i in range(n, m + 1):
        if arr[i] <= smallest:
            smallest += 1
    return smallest
```

Write a Python function to sort the given array by using merge sort. Your code should satisfy these tests:

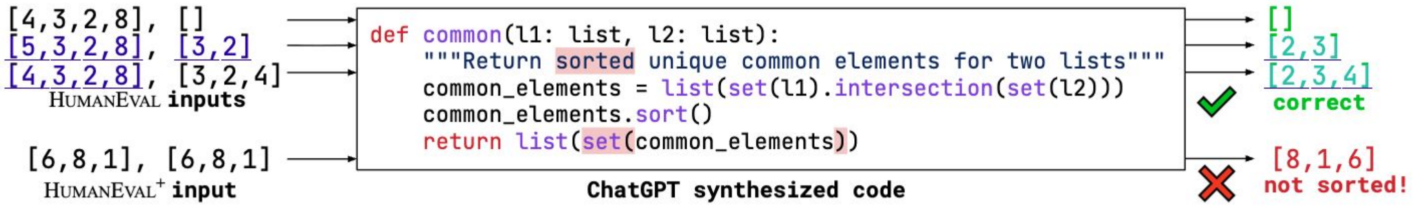
```
assert merge_sort([3, 4, 2, 6, 5, 7, 1, 9]) == [1, 2, 3, 4, 5, 6, 7, 9]
assert merge_sort([7, 25, 45, 78, 11, 33, 19]) == [7, 11, 19, 25, 33, 45, 78]
assert merge_sort([3, 1, 4, 9, 8]) == [1, 3, 4, 8, 9]
```

```
def merge_sort(arr):
  if len(arr) < 2:
    return arr;
  mid = len(arr) // 2;
  left = arr[:mid];
  right = arr[mid:];
  left = merge_sort(left);
  right = merge_sort(right);
  merged = [];
  i = j = 0;
  while i < len(left) and j < len(right):
    if left[i] < right[j]:
      merged.append(left[i]);
      i += 1;
    else:
      merged.append(right[j]);
      j += 1;
  merged.extend(left[i:]);
  merged.extend(right[j:]);
  return merged;
```

Austin, J., Odena, A., Nye, M., Bosma, M., Michalewski, H., Dohan, D., ... & Sutton, C. (2021). Program synthesis with large language models. arXiv preprint arXiv:2108.07732.

NL2Code: EvalPlus

- ➢ Existing test cases are not robust enough.

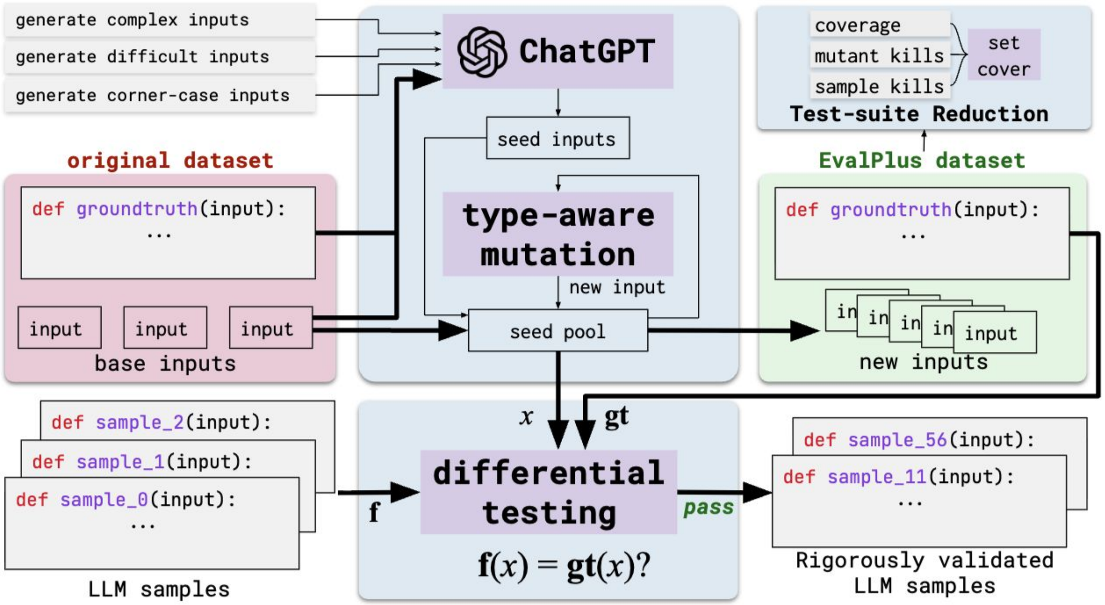


```
def common(l1: list, l2: list):
    """Return sorted unique common elements for two lists"""
    common_elements = list(set(l1).intersection(set(l2)))
    common_elements.sort()
    return list(set(common_elements))
```

Liu, J., Xia, C. S., Wang, Y., & Zhang, L. (2023). Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation. Advances in Neural Information Processing Systems, 36, 21558-21572.

NL2Code: EvalPlus

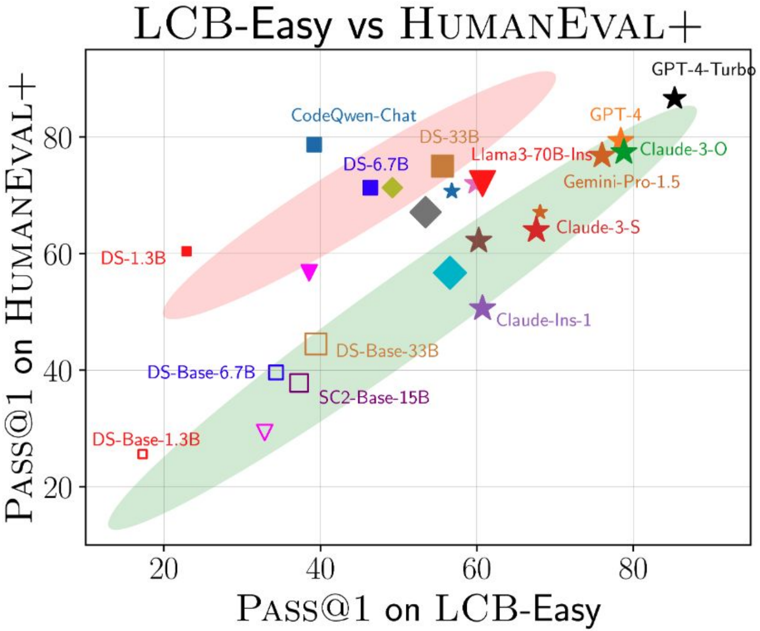
➢ Use LLMs and fuzzing (type-aware mutation) to create test cases.



Liu, J., Xia, C. S., Wang, Y., & Zhang, L. (2023). Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation. Advances in Neural Information Processing Systems, 36, 21558-21572.

NL2Code: LiveCodeBench

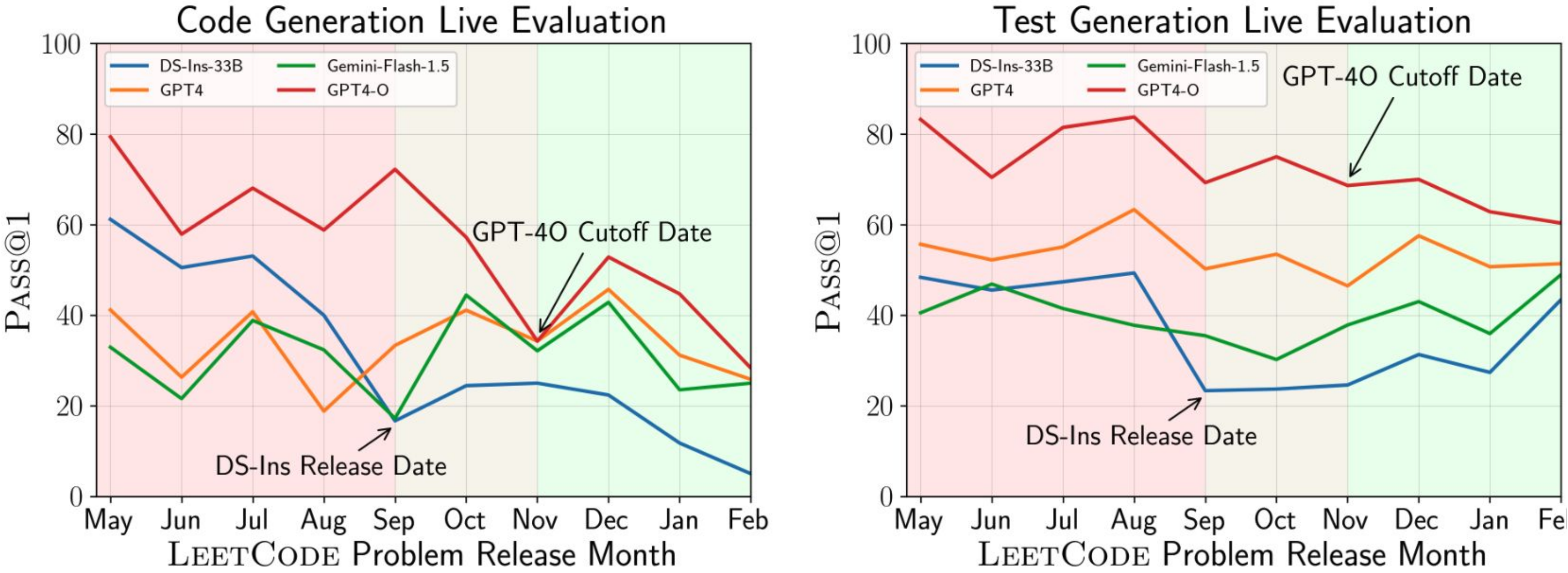
- ➢ Updated with problems created after models have been trained.



Jain, N., Han, K., Gu, A., Li, W. D., Yan, F., Zhang, T., ... & Stoica, I. LiveCodeBench: Holistic and Contamination Free Evaluation of Large Language Models for Code. In The Thirteenth International Conference on Learning Representations.

NL2Code: LiveCodeBench

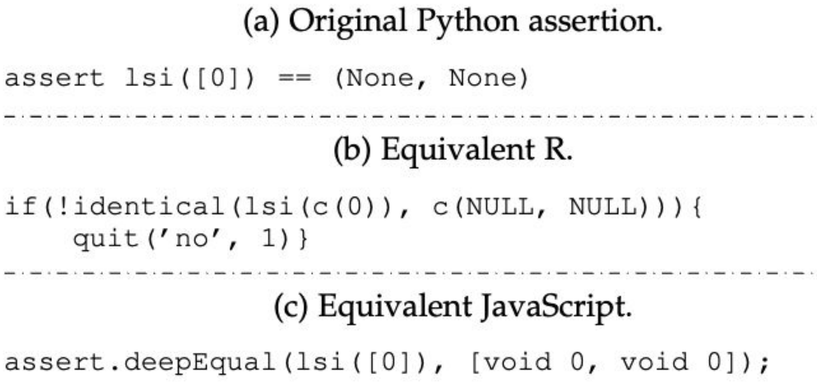
➢ Data contamination is hard to be fully mitigated.



Jain, N., Han, K., Gu, A., Li, W. D., Yan, F., Zhang, T., ... & Stoica, I. LiveCodeBench: Holistic and Contamination Free Evaluation of Large Language Models for Code. In The Thirteenth International Conference on Learning Representations.

NL2Code: MultiPL-E

- Relatively easy to translate test cases on simple types (e.g. nomatrices or functions) from Python to other languages.



(a) Original Python assertion.

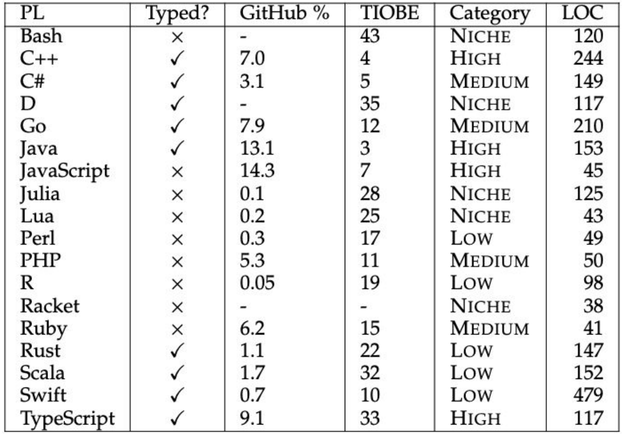
```
assert lsi([0]) == (None, None)
```

(b) Equivalent R.

```
if(!identical(lsi(c(0)), c(NULL, NULL))){
    quit('no', 1)}
```

(c) Equivalent JavaScript.

```
assert.deepEqual(lsi([0]), [void 0, void 0]);
```

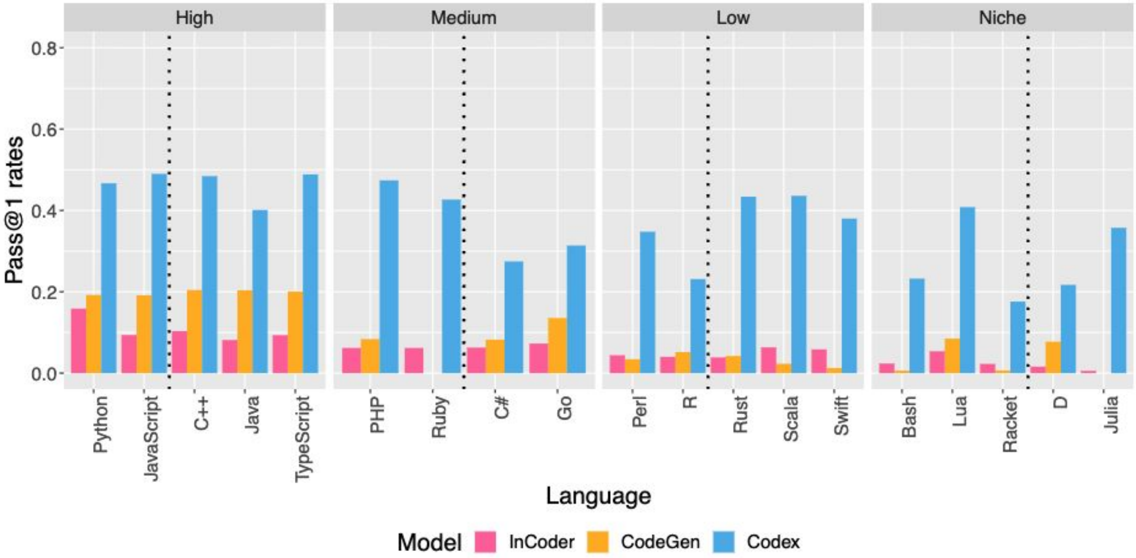


PL	Typed?	GitHub %	TIOBE	Category	LOC
Bash	×	-	43	Niche	120
C++	✓	7.0	4	High	244
C#	✓	3.1	5	Medium	149
D	✓	-	35	Niche	117
Go	✓	7.9	12	Medium	210
Java	✓	13.1	3	High	153
JavaScript	×	14.3	7	High	45
Julia	×	0.1	28	Niche	125
Lua	×	0.2	25	Niche	43
Perl	×	0.3	17	Low	49
PHP	×	5.3	11	Medium	50
R	×	0.05	19	Low	98
Racket	×	-	-	Niche	38
Ruby	×	6.2	15	Medium	41
Rust	✓	1.1	22	Low	147
Scala	✓	1.7	32	Low	152
Swift	✓	0.7	10	Low	479
TypeScript	✓	9.1	33	High	117

Cassano, F., Gouwar, J., Nguyen, D., Nguyen, S., Phipps-Costin, L., Pinckney, D., ... & Jangda, A. (2023). Multipl-e: A scalable and polyglot approach to benchmarking neural code generation. IEEE Transactions on Software Engineering, 49(7), 3675-3691.

NL2Code: MultiPL-E

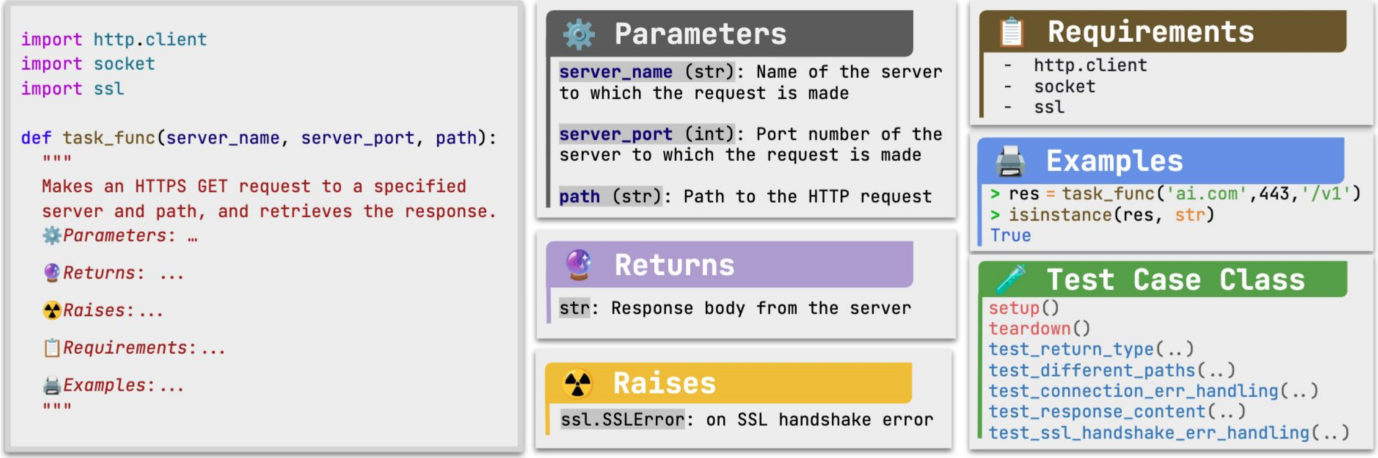
➢ Allows porting HumanEval & MBPP to 18 other languages.



Cassano, F., Gouwar, J., Nguyen, D., Nguyen, S., Phipps-Costin, L., Pinckney, D., ... & Jangda, A. (2023). Multipl-e: A scalable and polyglot approach to benchmarking neural code generation. IEEE Transactions on Software Engineering, 49(7), 3675-3691.

NL2Code: BigCodeBench

➢ Code evaluation should be more aligned with real-world development.



```
import http.client
import socket
import ssl

def task_func(server_name, server_port, path):
  """
  Makes an HTTPS GET request to a specified
  server and path, and retrieves the response.
  ⚙️Parameters: …

  🔮Returns:  ...

  ☢️Raises:...

  📋Requirements:...

  🖨️Examples:...
  """
```

⚙️ Parameters

`server_name (str)`: Name of the server to which the request is made

`server_port (int)`: Port number of the server to which the request is made

`path (str)`: Path to the HTTP request

🔮 Returns

`str`: Response body from the server

☢️ Raises

`ssl.SSLError`: on SSL handshake error

📋 Requirements

- http.client
- socket
- ssl

🖨️ Examples

```
> res = task_func('ai.com',443,'/v1')
> isinstance(res, str)
True
```

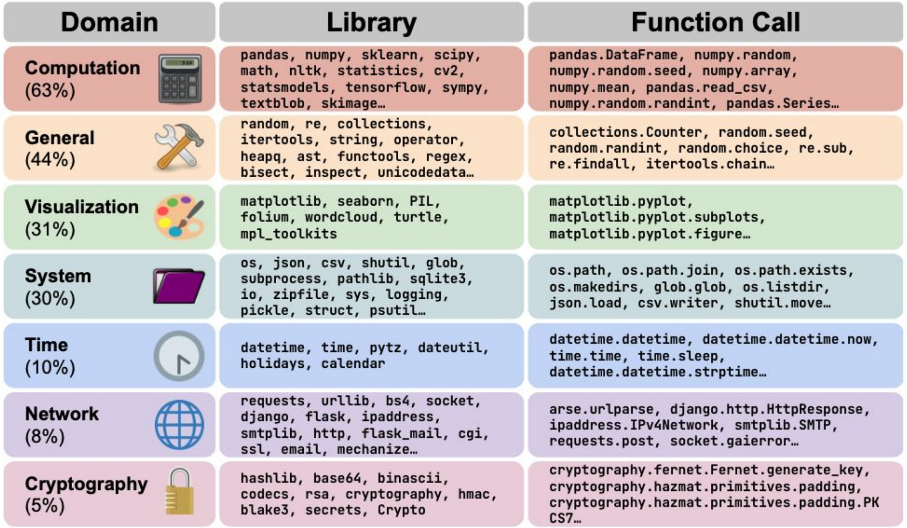
🧪 Test Case Class

```
setup()
teardown()
test_return_type(..)
test_different_paths(..)
test_connection_err_handling(..)
test_response_content(..)
test_ssl_handshake_err_handling(..)
```

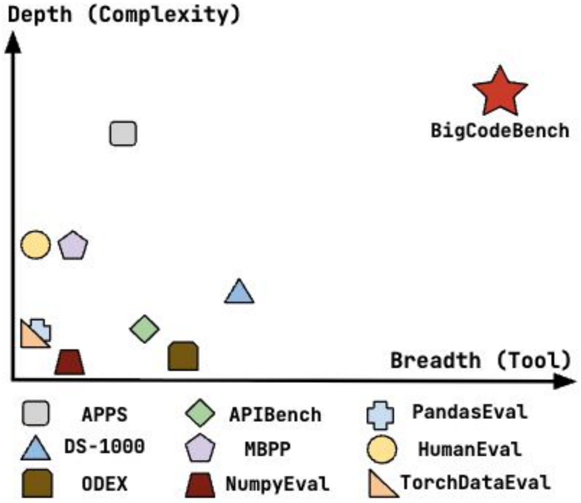
Zhuo, T. Y., Chien, V. M., Chim, J., Hu, H., Yu, W., Widyasari, R., ... & Von Werra, L. BigCodeBench: Benchmarking Code Generation with Diverse Function Calls and Complex Instructions. In The Thirteenth International Conference on Learning Representations.

NL2Code: BigCodeBench

- Scaling up to more complex library-using problems and instructions.



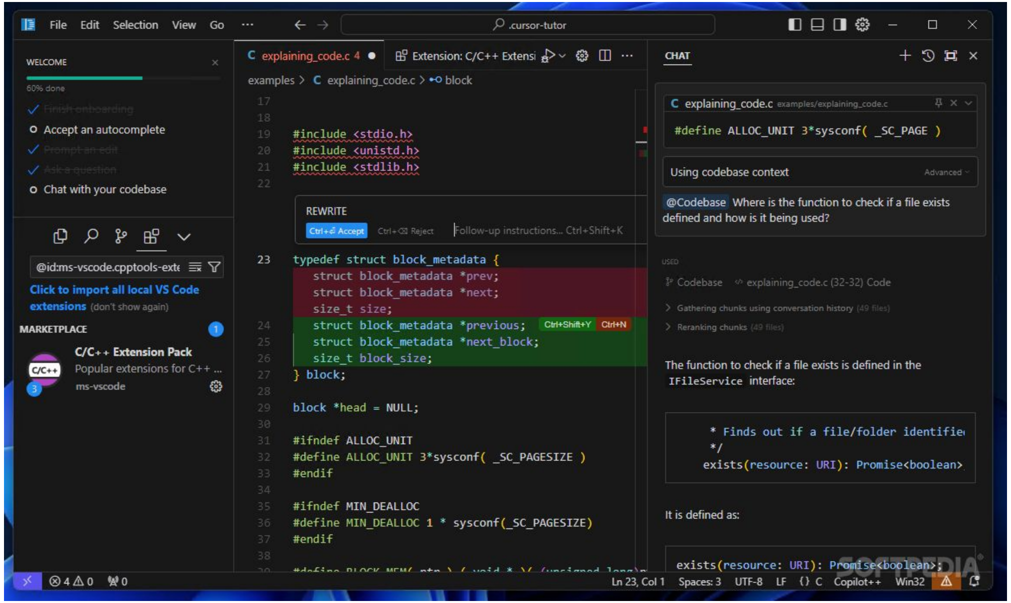
Domain	Library	Function Call
Computation (63%)	pandas, numpy, sklearn, scipy, math, nltk, statistics, cv2, statsmodels, tensorflow, sympy, textblob, skimage…	pandas.DataFrame, numpy.random, numpy.random.seed, numpy.array, numpy.mean, pandas.read_csv, numpy.random.randint, pandas.Series…
General (44%)	random, re, collections, itertools, string, operator, heapq, ast, functools, regex, bisect, inspect, unicodedata…	collections.Counter, random.seed, random.randint, random.choice, re.sub, re.findall, itertools.chain…
Visualization (31%)	matplotlib, seaborn, PIL, folium, wordcloud, turtle, mpl_toolkits	matplotlib.pyplot, matplotlib.pyplot.subplots, matplotlib.pyplot.figure…
System (30%)	os, json, csv, shutil, glob, subprocess, pathlib, sqlite3, io, zipfile, sys, logging, pickle, struct, psutil…	os.path, os.path.join, os.path.exists, os.makedirs, glob.glob, os.listdir, json.load, csv.writer, shutil.move…
Time (10%)	datetime, time, pytz, dateutil, holidays, calendar	datetime.datetime, datetime.datetime.now, time.time, time.sleep, datetime.datetime.strptime…
Network (8%)	requests, urllib, bs4, socket, django, flask, ipaddress, smtplib, http, flask_mail, cgi, ssl, email, mechanize…	arse.urlparse, django.http.HttpResponse, ipaddress.IPv4Network, smtplib.SMTP, requests.post, socket.gaierror…
Cryptography (5%)	hashlib, base64, binascii, codecs, rsa, cryptography, hmac, blake3, secrets, Crypto	cryptography.fernet.Fernet.generate_key, cryptography.hazmat.primitives.padding, cryptography.hazmat.primitives.padding.PKCS7…



Zhuo, T. Y., Chien, V. M., Chim, J., Hu, H., Yu, W., Widyasari, R., ... & Von Werra, L. BigCodeBench: Benchmarking Code Generation with Diverse Function Calls and Complex Instructions. In The Thirteenth International Conference on Learning Representations.

Code Editing

- ➢ Feature removal/addition, Bug fixing, Code refactoring…
  - ○ Input: Natural Language Instructions + Code
  - ○ Output: Code



```c
#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>

typedef struct block_metadata {
   struct block_metadata *prev;
   struct block_metadata *next;
   size_t size;
   struct block_metadata *previous;
   struct block_metadata *next_block;
   size_t block_size;
} block;

block *head = NULL;

#ifndef ALLOC_UNIT
#define ALLOC_UNIT 3*sysconf( _SC_PAGESIZE )
#endif

#ifndef MIN_DEALLOC
#define MIN_DEALLOC 1 * sysconf(_SC_PAGESIZE)
#endif

#define BLOCK_MEM( ptr ) ( void * )( (unsigned long)
```

explaining_code.c examples/explaining_code.c

```c
#define ALLOC_UNIT 3*sysconf( _SC_PAGE )
```

Using codebase context Advanced

@Codebase Where is the function to check if a file exists defined and how is it being used?

USED

Codebase explaining_code.c (32-32) Code

> Gathering chunks using conversation history (49 files)

> Reranking chunks (49 files)

The function to check if a file exists is defined in the `IFileService` interface:

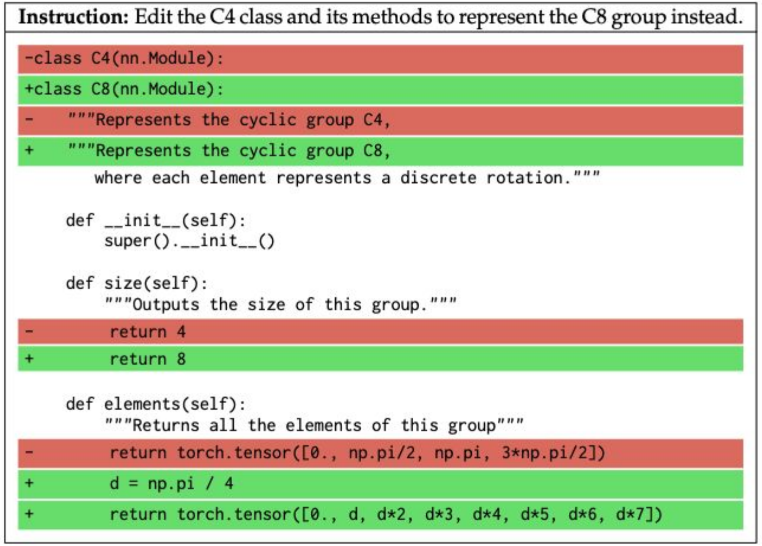
```
     * Finds out if a file/folder identifie
     */
    exists(resource: URI): Promise<boolean>
```

It is defined as:

```
exists(resource: URI): Promise<boolean>;
```

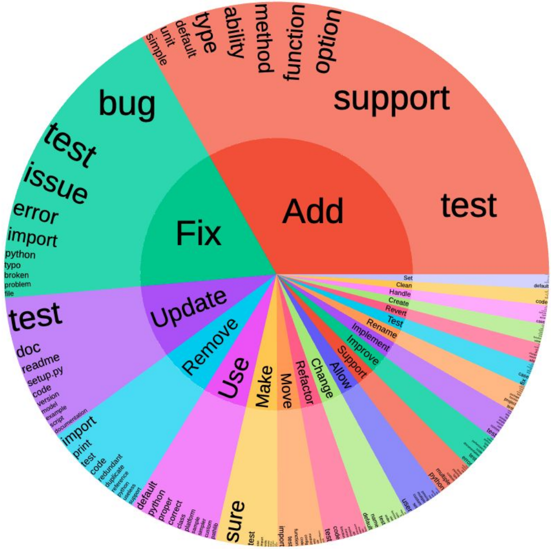
Code Editing: CanItEdit

- Write correct code modifications without introducing unnecessary code.
  - *pass@k*
  - *ExcessCode*: uncovered changed lines in passing completions.



Instruction: Edit the C4 class and its methods to represent the C8 group instead.

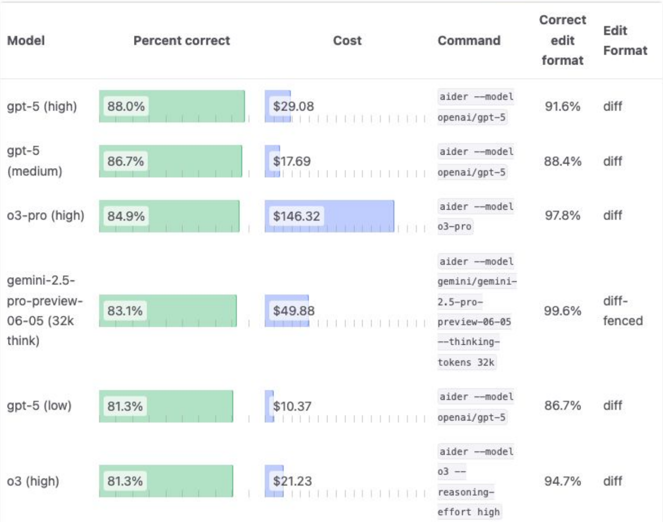
```
-class C4(nn.Module):
+class C8(nn.Module):
-    """Represents the cyclic group C4,
+    """Represents the cyclic group C8,
        where each element represents a discrete rotation."""

    def __init__(self):
        super().__init__()

    def size(self):
        """Outputs the size of this group."""
-        return 4
+        return 8

    def elements(self):
        """Returns all the elements of this group"""
-        return torch.tensor([0., np.pi/2, np.pi, 3*np.pi/2])
+        d = np.pi / 4
+        return torch.tensor([0., d, d*2, d*3, d*4, d*5, d*6, d*7])
```



Cassano, F., Li, L., Sethi, A., Shinn, N., Brennan-Jones, A., Ginesin, J., ... & Guha, A. Can It Edit? Evaluating the Ability of Large Language Models to Follow Code Editing Instructions. In First Conference on Language Modeling.

Code Editing: Aider Polyglot

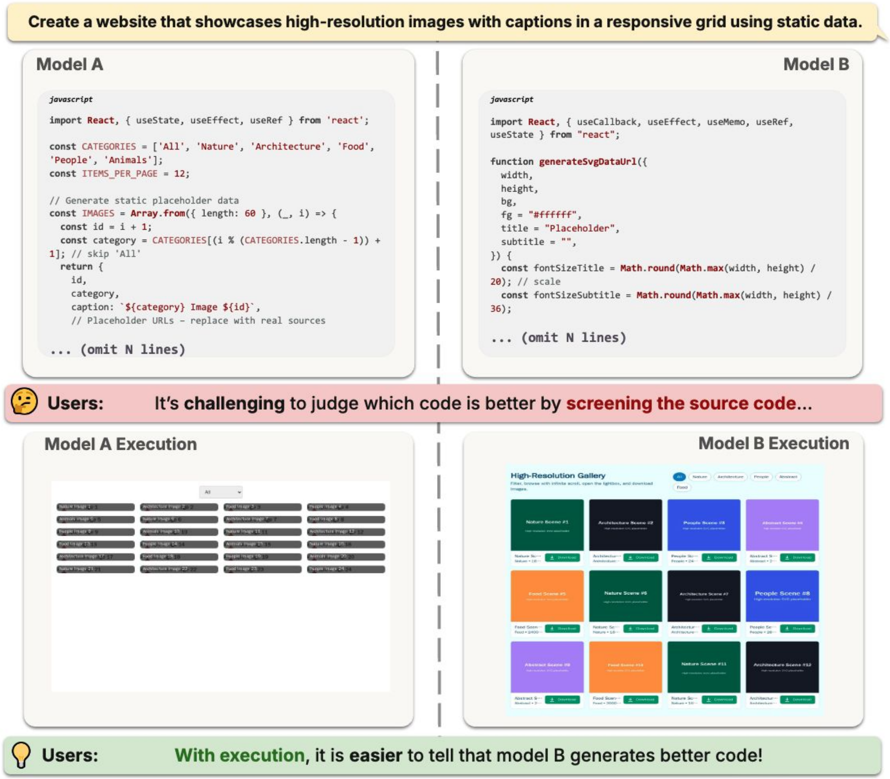
- Extend to multilingual code editing in 6 programming languages.
  - *Edit format*: format that the model is instructed to use to edit files
  - *Edit format accuracy*: percentage of problems for which the model complies with the specified edit format
  - *Cost*



Model	Percent correct	Cost	Command	Correct edit format	Edit Format
gpt-5 (high)	88.0%	$29.08	`aider --model openai/gpt-5`	91.6%	diff
gpt-5 (medium)	86.7%	$17.69	`aider --model openai/gpt-5`	88.4%	diff
o3-pro (high)	84.9%	$146.32	`aider --model o3-pro`	97.8%	diff
gemini-2.5-pro-preview-06-05 (32k think)	83.1%	$49.88	`aider --model gemini/gemini-2.5-pro-preview-06-05 --thinking-tokens 32k`	99.6%	diff-fenced
gpt-5 (low)	81.3%	$10.37	`aider --model openai/gpt-5`	86.7%	diff
o3 (high)	81.3%	$21.23	`aider --model o3 --reasoning-effort high`	94.7%	diff

https://aider.chat/docs/leaderboards/#polyglot-leaderboard

Judging Code Generation via Human Preference



Create a website that showcases high-resolution images with captions in a responsive grid using static data.

Model A

```javascript
import React, { useState, useEffect, useRef } from 'react';

const CATEGORIES = ['All', 'Nature', 'Architecture', 'Food',
'People', 'Animals'];
const ITEMS_PER_PAGE = 12;

// Generate static placeholder data
const IMAGES = Array.from({ length: 60 }, (_, i) => {
  const id = i + 1;
  const category = CATEGORIES[(i % (CATEGORIES.length - 1)) +
1]; // skip 'All'
  return {
    id,
    category,
    caption: `${category} Image ${id}`,
    // Placeholder URLs - replace with real sources

... (omit N lines)
```

Model B

```javascript
import React, { useCallback, useEffect, useMemo, useRef,
useState } from "react";

function generateSvgDataUrl({
  width,
  height,
  bg,
  fg = "#ffffff",
  title = "Placeholder",
  subtitle = "",
}) {
  const fontSizeTitle = Math.round(Math.max(width, height) /
20); // scale
  const fontSizeSubtitle = Math.round(Math.max(width, height) /
36);

... (omit N lines)
```

🤔 **Users:** It's **challenging** to judge which code is better by **screening the source code...**

Model A Execution

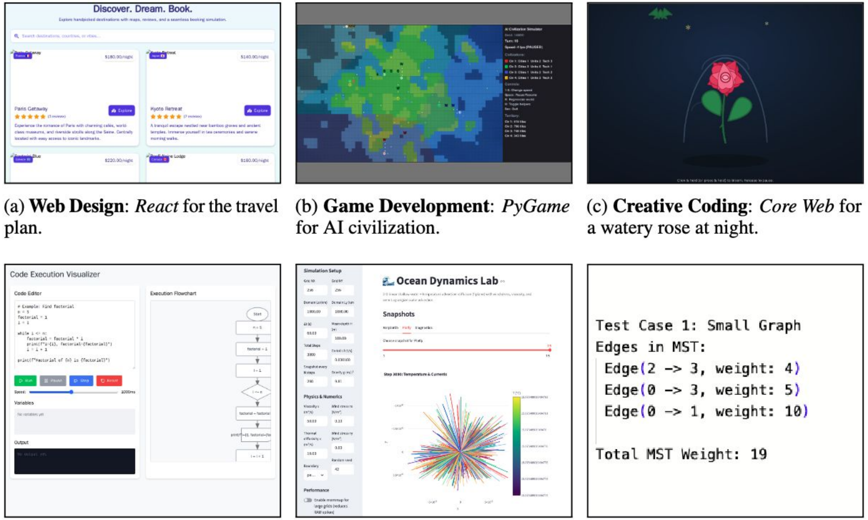
Model B Execution

💡 **Users:** **With execution**, it is **easier** to tell that model B generates better code!

Zhuo, T. Y., Jin, X., Liu, H., Jiang, J., Liu, T., Gong, C., ... & von Werra, L. (2025). BigCodeArena: Unveiling More Reliable Human Preferences in Code Generation via Execution. arXiv preprint arXiv:2510.08697.

Judging via Human Preference: BigCodeArena

- An open human evaluation platform for code generation that enables real-time execution and interaction, revealing preferences and capabilities of LLMs in coding tasks.
  - *14K conversations across 10 widely used LLMs, with 10 languages and 8 frameworks*



(a) **Web Design**: *React* for the travel plan.

(b) **Game Development**: *PyGame* for AI civilization.

(c) **Creative Coding**: *Core Web* for a watery rose at night.

```
Test Case 1: Small Graph
Edges in MST:
 Edge(2 -> 3, weight: 4)
 Edge(0 -> 3, weight: 5)
 Edge(0 -> 1, weight: 10)

Total MST Weight: 19
```

(d) **Diagram Creation**: *Vue* for deployment workflow visualization.

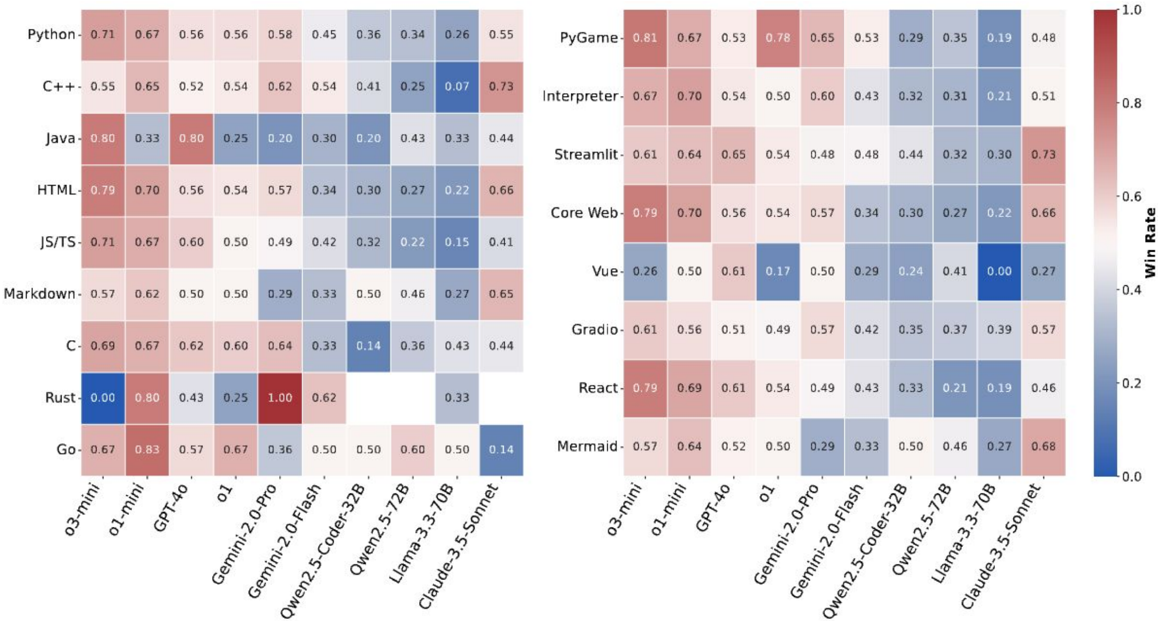
(e) **Scientific Computing**: *Streamlit* for Ocean simulation.

(f) **Problem Solving**: *Interpreter* for Kruskal's Algorithm for MST.

Zhuo, T. Y., Jin, X., Liu, H., Jiang, J., Liu, T., Gong, C., ... & von Werra, L. (2025). BigCodeArena: Unveiling More Reliable Human Preferences in Code Generation via Execution. arXiv preprint arXiv:2510.08697.

Judging via Human Preference: BigCodeArena

- An open human evaluation platform for code generation that enables real-time execution and interaction, revealing preferences and capabilities of LLMs in coding tasks.
    - *4.7K multi-turn pairwise conversations with human voting*



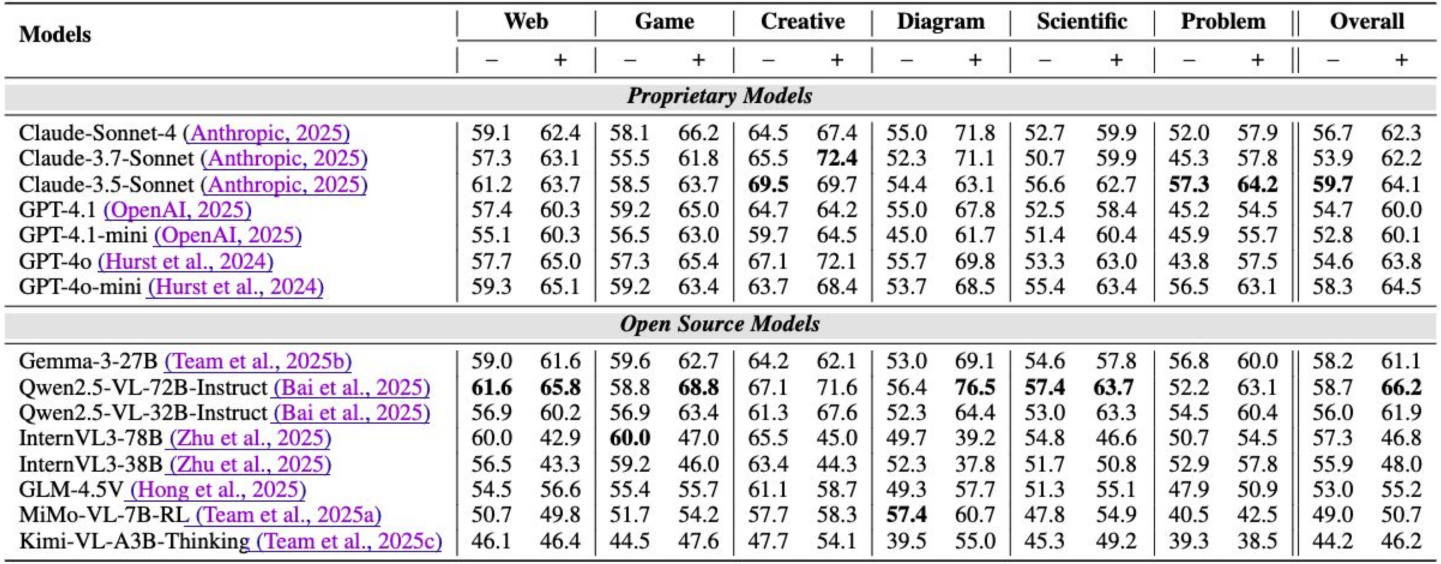
	o3-mini	o1-mini	GPT-4o	o1	Gemini-2.0-Pro	Gemini-2.0-Flash	Qwen2.5-Coder-32B	Qwen2.5-72B	Llama-3.3-70B	Claude-3.5-Sonnet
Python	0.71	0.67	0.56	0.56	0.58	0.45	0.36	0.34	0.26	0.55
C++	0.55	0.65	0.52	0.54	0.62	0.54	0.41	0.25	0.07	0.73
Java	0.80	0.33	0.80	0.25	0.20	0.30	0.20	0.43	0.33	0.44
HTML	0.79	0.70	0.56	0.54	0.57	0.34	0.30	0.27	0.22	0.66
JS/TS	0.71	0.67	0.60	0.50	0.49	0.42	0.32	0.22	0.15	0.41
Markdown	0.57	0.62	0.50	0.50	0.29	0.33	0.50	0.46	0.27	0.65
C	0.69	0.67	0.62	0.60	0.64	0.33	0.14	0.36	0.43	0.44
Rust	0.00	0.80	0.43	0.25	1.00	0.62			0.33	
Go	0.67	0.83	0.57	0.67	0.36	0.50	0.50	0.60	0.50	0.14

	o3-mini	o1-mini	GPT-4o	o1	Gemini-2.0-Pro	Gemini-2.0-Flash	Qwen2.5-Coder-32B	Qwen2.5-72B	Llama-3.3-70B	Claude-3.5-Sonnet
PyGame	0.81	0.67	0.53	0.78	0.65	0.53	0.29	0.35	0.19	0.48
Interpreter	0.67	0.70	0.54	0.50	0.60	0.43	0.32	0.31	0.21	0.51
Streamlit	0.61	0.64	0.65	0.54	0.48	0.48	0.44	0.32	0.30	0.73
Core Web	0.79	0.70	0.56	0.54	0.57	0.34	0.30	0.27	0.22	0.66
Vue	0.26	0.50	0.61	0.17	0.50	0.29	0.24	0.41	0.00	0.27
Gradio	0.61	0.56	0.51	0.49	0.57	0.42	0.35	0.37	0.39	0.57
React	0.79	0.69	0.61	0.54	0.49	0.43	0.33	0.21	0.19	0.46
Mermaid	0.57	0.64	0.52	0.50	0.29	0.33	0.50	0.46	0.27	0.68

Zhuo, T. Y., Jin, X., Liu, H., Jiang, J., Liu, T., Gong, C., ... & von werra, L. (2025). BigCodeArena: Unveiling More Reliable Human Preferences in Code Generation via Execution. arXiv preprint arXiv:2510.08697.

Judging via Human Preference: BigCodeArena

➢ *BigCodeReward*: Alignment between reward models and human judgments in code

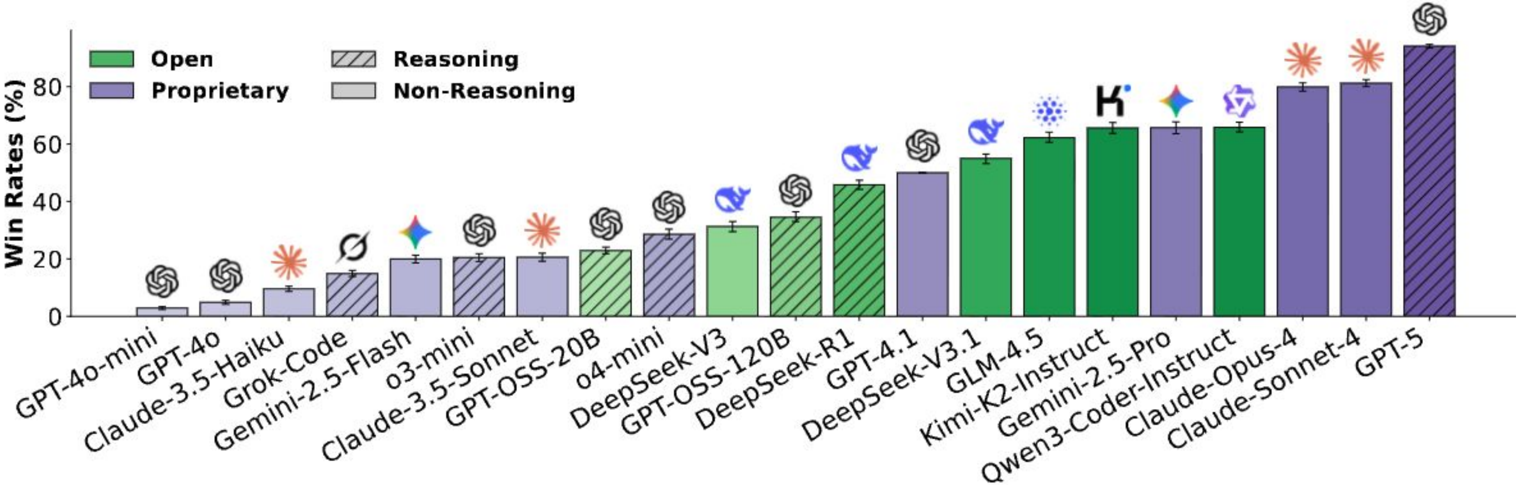


Models	Web		Game		Creative		Diagram		Scientific		Problem		Overall	
	–	+	–	+	–	+	–	+	–	+	–	+	–	+
Proprietary Models														
Claude-Sonnet-4 (Anthropic, 2025)	59.1	62.4	58.1	66.2	64.5	67.4	55.0	71.8	52.7	59.9	52.0	57.9	56.7	62.3
Claude-3.7-Sonnet (Anthropic, 2025)	57.3	63.1	55.5	61.8	65.5	**72.4**	52.3	71.1	50.7	59.9	45.3	57.8	53.9	62.2
Claude-3.5-Sonnet (Anthropic, 2025)	61.2	63.7	58.5	63.7	**69.5**	69.7	54.4	63.1	56.6	62.7	**57.3**	**64.2**	**59.7**	64.1
GPT-4.1 (OpenAI, 2025)	57.4	60.3	59.2	65.0	64.7	64.2	55.0	67.8	52.5	58.4	45.2	54.5	54.7	60.0
GPT-4.1-mini (OpenAI, 2025)	55.1	60.3	56.5	63.0	59.7	64.5	45.0	61.7	51.4	60.4	45.9	55.7	52.8	60.1
GPT-4o (Hurst et al., 2024)	57.7	65.0	57.3	65.4	67.1	72.1	55.7	69.8	53.3	63.0	43.8	57.5	54.6	63.8
GPT-4o-mini (Hurst et al., 2024)	59.3	65.1	59.2	63.4	63.7	68.4	53.7	68.5	55.4	63.4	56.5	63.1	58.3	64.5
Open Source Models														
Gemma-3-27B (Team et al., 2025b)	59.0	61.6	59.6	62.7	64.2	62.1	53.0	69.1	54.6	57.8	56.8	60.0	58.2	61.1
Qwen2.5-VL-72B-Instruct (Bai et al., 2025)	**61.6**	**65.8**	58.8	**68.8**	67.1	71.6	56.4	**76.5**	**57.4**	**63.7**	52.2	63.1	58.7	**66.2**
Qwen2.5-VL-32B-Instruct (Bai et al., 2025)	56.9	60.2	56.9	63.4	61.3	67.6	52.3	64.4	53.0	63.3	54.5	60.4	56.0	61.9
InternVL3-78B (Zhu et al., 2025)	60.0	42.9	**60.0**	47.0	65.5	45.0	49.7	39.2	54.8	46.6	50.7	54.5	57.3	46.8
InternVL3-38B (Zhu et al., 2025)	56.5	43.3	59.2	46.0	63.4	44.3	52.3	37.8	51.7	50.8	52.9	57.8	55.9	48.0
GLM-4.5V (Hong et al., 2025)	54.5	56.6	55.4	55.7	61.1	58.7	49.3	57.7	51.3	55.1	47.9	50.9	53.0	55.2
MiMo-VL-7B-RL (Team et al., 2025a)	50.7	49.8	51.7	54.2	57.7	58.3	**57.4**	60.7	47.8	54.9	40.5	42.5	49.0	50.7
Kimi-VL-A3B-Thinking (Team et al., 2025c)	46.1	46.4	44.5	47.6	47.7	54.1	39.5	55.0	45.3	49.2	39.3	38.5	44.2	46.2

Zhuo, T. Y., Jin, X., Liu, H., Jiang, J., Liu, T., Gong, C., ... & von Werra, L. (2025). BigCodeArena: Unveiling More Reliable Human Preferences in Code Generation via Execution. arXiv preprint arXiv:2510.08697.

Judging via Human Preference: BigCodeArena

➢ *AutoCodeArena*: Low-cost and automated version of BigCodeArena



Zhuo, T. Y., Jin, X., Liu, H., Jiang, J., Liu, T., Gong, C., ... & von Werra, L. (2025). BigCodeArena: Unveiling More Reliable Human Preferences in Code Generation via Execution. arXiv preprint arXiv:2510.08697.