Coding Task Presentation

Hankun

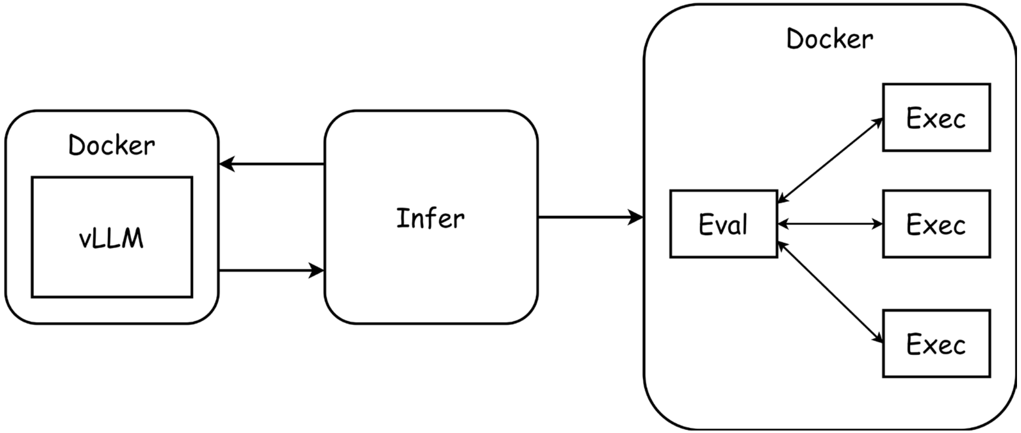
2025-05-15

Outline

- Project Structure
- Results
- Improvement

Project Structure

- Inference.py
  - Interact with server & extract code
- Evaluation.py
  - Task assignment & score computing



Inference

- Build prompt (LLM input)

```
prompt = (
    f"Write a Python function `{signature.group(1)}`
to solve the following problem.\n"
    f"{docstring.group(1)}\n"
    f"{snippet}"
)
```

  - Instruction & signature
  - Docstring
  - Snippet

```
{
"task_id": "HumanEval/0",
"prompt": "from typing import List\n\n\ndef
has_close_elements(numbers: List[float], threshold:
float) -> bool:\n    \"\"\" Check if in given list of
numbers, are any two numbers closer to each other
than\n    given threshold.\n    >>>
has_close_elements([1.0, 2.0, 3.0],
0.5)\n    False\n\"\"\"\n",
"entry_point": "has_close_elements",
"canonical_solution": "    if distance <
threshold:\n                        return
True\n\n    return False\n",
"test": "\n\nMETADATA = {\n    'author':
'jt',\n    'dataset': 'test'\n}\n\n\ndef
check(candidate):\n    assert candidate([1.0, 2.0, 3.9,
4.0, 5.0, 2.2], 0.3) == True\n“
}
```

Inference

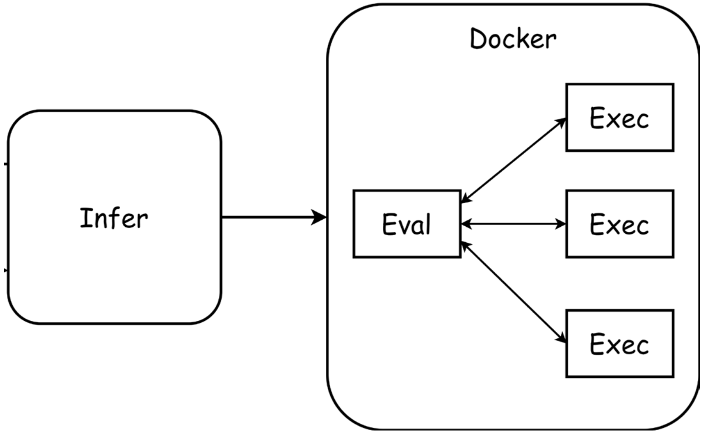
- Code extraction
  - Directly complete
  - Complete + think + code block
  - Repeat above
- Rules
  - Code block first, choose one
  - Drop test cases (no use: instruct more clearly)

````
"    for i in
range(len(numbers)):\n        for j in range(i
+ 1, len(numbers)):\n            if
abs(numbers[i] - numbers[j]) <
threshold:\n                return
True\n    return False\n\n\n# Test
cases\nassert has_close_elements([1.0, 2.0,
3.0], 0.5) == False\nassert
has_close_elements([1.0, 2.8, 3.0, 4.0, 5.0,
2.0], 0.3) == True\n```"
````

Evaluation

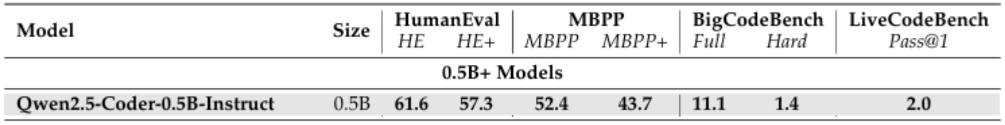
- Build Python code
  - Signature + completion + test
- Execution
  - Multiprocessing to run
- Compute pass@k score

$$\text{pass@}k := \mathbb{E}_{\text{Problems}}\left[1 - \frac{\binom{n-c}{k}}{\binom{n}{k}}\right]$$



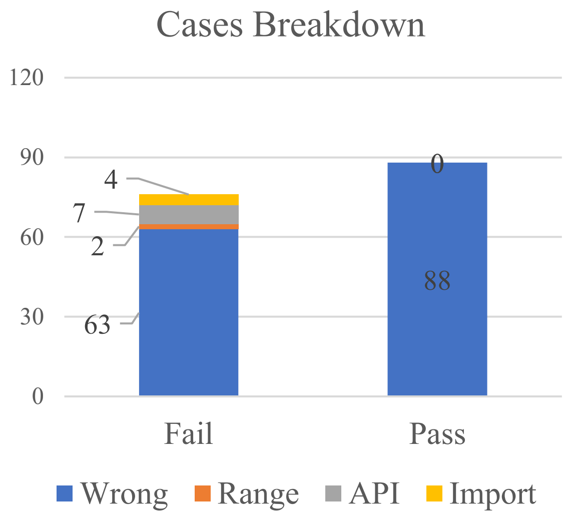
Results

- Score
  - 53.7% vs 61.6%



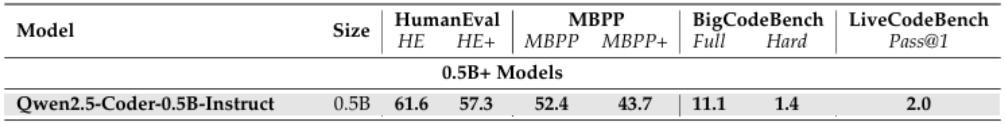
Model	Size	HumanEval		MBPP		BigCodeBench		LiveCodeBench
		HE	*HE+*	*MBPP*	*MBPP+*	*Full*	*Hard*	*Pass@1*
0.5B+ Models								
Qwen2.5-Coder-0.5B-Instruct	0.5B	**61.6**	**57.3**	**52.4**	**43.7**	**11.1**	**1.4**	**2.0**

- Failed cases - 76
  - Runtime error
    - API & range: corner cases
  - Wrong results

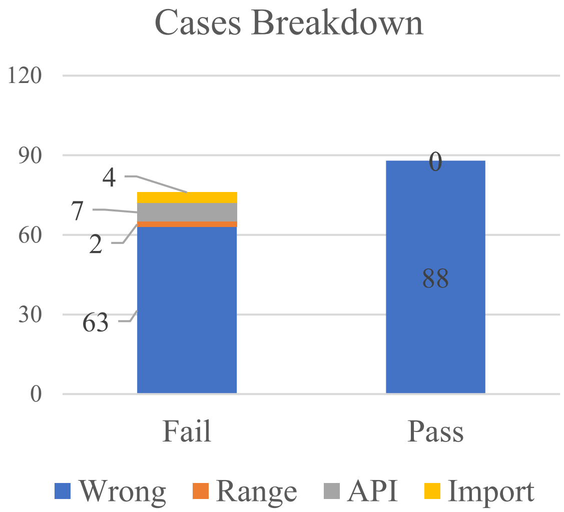


Results

- Score - 53.7%
- Failed cases - 76
  - Runtime error
  - Wrong results
- Improve
  - Prompt: official, prompt engineering is marginal
  - Code extraction: already complete
  - Exclude runtime error: 61.6, same as official!



Model	Size	HumanEval		MBPP		BigCodeBench		LiveCodeBench
		HE	*HE+*	*MBPP*	*MBPP+*	*Full*	*Hard*	*Pass@1*
0.5B+ Models								
Qwen2.5-Coder-0.5B-Instruct	0.5B	**61.6**	**57.3**	**52.4**	**43.7**	**11.1**	**1.4**	**2.0**



Performance Improvement

- Performance: score
    - Sampling parameters: raise k
    - MCTS
    - Tools for verification
    - Post-training
        - RL, distillation
- Efficiency: vLLM
    - Multiprocessing
    - Quantize, prune, speculative decoding, …

Summary

- Project Structure
- Results
- Improvement