Datathon 2025: Sentient Challenge

Chernoff Bound Team

Setup

Model used : Llama3.1-70B
Reasoning Flavor: CodeAct Agent
Hackers: Louis Barinka, Luis Carretero, Daniyar Zakarin, Aleks Stepančič

Multi-query prompting

Motivation

Going through the logs of the Open Reasoning Agent, we identified what seemed to be a limitation: since every new step in our agent reasoning added more tokens to the input, the total token count was scaling quadratically with the number of reasoning steps. This not only led to inefficient resource usage but also increased the risk of overwhelming the model with redundant or less relevant information.

Description

To address this issue, we developed a technique we call **multi-query prompting**. Instead of issuing a single query at each reasoning step, the agent now generates and dispatches multiple queries—typically three—concurrently. This approach distributes the search effort across several queries, ensuring that each query can capture different facets of the task at hand. Consequently, this technique reduces the quadratic growth in token input size while simultaneously enhancing the diversity and quality of retrieved information.

Example

Here, 3 queries were launched at the same time, allowing to get a higher amount of information in the same query already.

```
─ Executing parsed code: ──────────────────────────────────────────────
year_1 = web_search(query="HMS Holland 1 launch year")
print(year_1)
year_2 = web_search(query="launch year of HMS Holland 1")
print(year_2)
year_3 = web_search(query="when was HMS Holland 1 launched")
print(year_3)
──────────────────────────────────────────────────────────────────────
```

Result

In our experiments, this approach improved the performance of the vanilla method by a 4 percent on FRAME while keeping the overall token usage similar, making it indeed more efficient.
The numerical values will be uncovered to you once we successfully wake up the team member that ran those experiments.

Future experimentations

For now, the multiple search queries made during the same step are fairly similar. What we might want to explore is making the agent do more diverse search during the execution. We believe this might help lead to better results.