

Relatório 2 - Prática: Python: Criando um ReAct Agent do Zero (I)

Guilherme Mileib

Descrição da atividade

Nesta atividade foi proposto a implementação da criação de um agente de IA usando o ReAct, um método de respostas de um agente LLM. A ideia é que a partir de uma query, pergunta de um usuário ele passe por esses processos:

1. Pensamento

Ao invés de chutar uma resposta na hora, o modelo primeiro pensa sobre o caso. Analisa com as informações iniciais, formula hipóteses sobre como resolver o mistério e traça um plano para entregar a resposta.

2. Ação

Com o plano traçado, agora chega a parte da ação. A qual, ele usa as ferramentas para coletar novas informações e realizar testes.

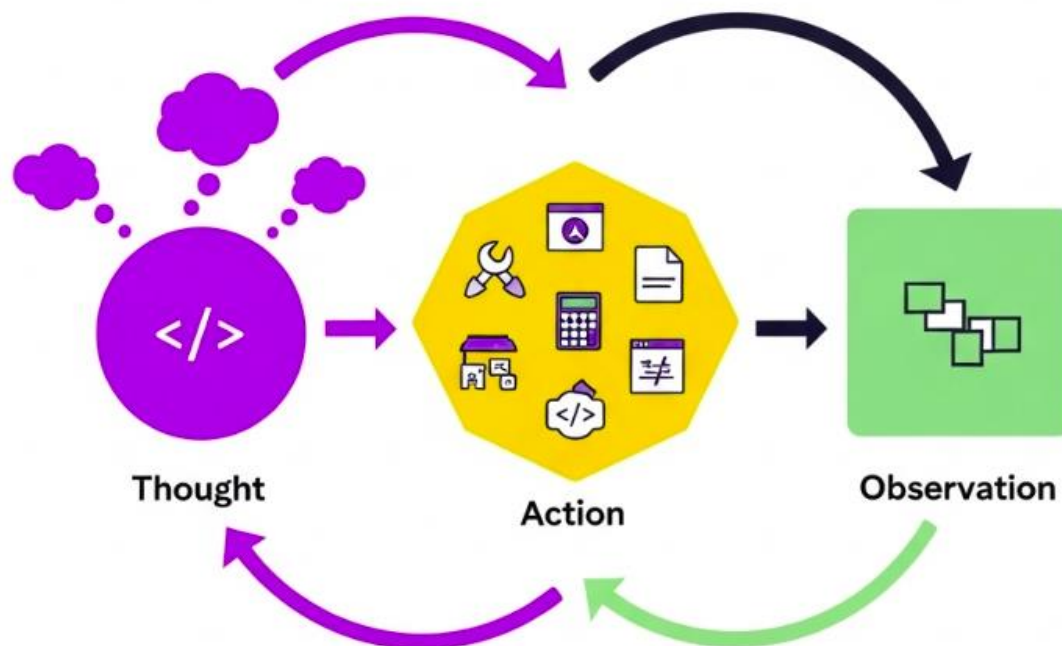
3. Observação

Após cada ação, o modelo observa os resultados. Verificando se as informações encontradas correspondem às suas hipóteses.

4. Iteração do Ciclo

Se o resultado da observação não resolver o caso por completo, o modelo reajusta o plano – novo pensamento. Realizando o ciclo novamente – novas investigações, usar outras ferramentas. Esse ciclo de Pensar – Agir – Observar - Refletir continua até que a avaliação probabilística e baseada em critérios definidos seja satisfatória.

Isso pode ser visualizado nessa imagem:



Sobre a parte prática

Fiquei com um pouco de dúvida do que colocar, então pensei em utilizar algo que venho pensando ultimamente, 'quantas braçadas um nadador poderia dar em uma piscina olímpica?'

- ⑩ Para isso como dito no vídeo, eu instrui a LLM seguir o ciclo: Thought → Action → PAUSE → Observation.
- ⑩ Depois declarei as ferramentas que o agente poderia usar (calculate e estimate_swimming_strokes)
 - ↳ *estimate_swimming_strokes(height, level, pool_length)*: Esta é a ferramenta principal. É uma função Python que recebe os dados do nadador (altura, nível) e da piscina e faz um cálculo real para retornar uma estimativa numérica.
- ⑩ Depois dei um exemplo de como poderia ser feito, para que o modelo tivesse um direcionamento – percebi que tinha que ser bem claro principalmente pelo input necessitar de parâmetros. Pois esses parâmetros seriam o que direcionaria a resposta que eu desejava que era saber a partir do nível de um nadador (level), quantas braçadas para finalizar a piscina.

Dificuldades

Minhas dificuldades foram na parte prática a qual tive que pesquisar muito dos termo de python, pois havia me esquecido ou nunca tive contato. Por exemplo, a função eval(), que pega uma string como argumento e executa-a como se fosse uma expressão, gerando direto o resultado da expressão. Como result = eval('5 + 10 * 2'), se eu printo resultado disso em, print(result), tenho a saída de 25.

O eval é usado para pegar o texto que o LLM gera e colocá-lo em ação executável no código python.

Conclusões

Com essa prática pude aprender como fazer um sistema de IA que apenas não responde ao que pergunto, mas que utiliza o meu código como ferramenta para chegar a uma conclusão mais precisa e fundamentada da pergunta que desejo.

Referencias

- <https://console.groq.com/docs/quickstart>
- <https://til.simonwillison.net/llms/python-react-pattern>
- <https://www.youtube.com/watch?v=hKVhRA9kfeM>