Extracting formal smart contract specifications from natural language with LLMs

Experimentando novamente com modelo GPT-4o da openAI.

Dificuldades

Limite de tokens por minuto

O 4o possui um limite mais baixo para usuários tier1.
Nossa estratégia foi colocar um time.sleep() entre iterações do loop.

Tempo

Consequência do time.sleep().
Especialmente nos casos em que o modelo errava bastante.

Custo financeiro

O limite de tokens por minuto fez com que muitas iterações não fossem concluídas, impedido a sua avaliação. Mas ainda assim custando caro.

Dificuldades

Experimentação manual

Vários experimentos falhavam e tinham que ser retomados, então dificulta a anotação precisa e observação dos resultados.

Resultados anteriores

Testamos novamente o GPT-4o para avaliação de melhora de um mesmo modelo com um gap temporal.

	Input		
Dem. Context	20	721	1155
ϵ	2 [2;4]	1 [9;9]	0
20	10 [0;3]	5 [1;4]	✓2 [8;9]
721	4 [1;5]	10 [0;0]	✓2 [6;9]
1155	5 [1;5]	6 [1;4]	✓10 [0;2]
20, 721	10 [0;7]	10 [0;1]	✓3 [7;9]
20, 1155	10 [0;5]	7 [1;2]	✓10 [0;0]
721, 1155	7 [1;7]	10 [0;4]	✓10 [0;0]
20, 721, 1155	9 [0;4]	10 [0;4]	✓10 [0;2]

Resultados novos

Modelo: GPT-4o

	input		
Dem. context	**20**	**721**	**1155**
ε	4/10	3/10	3/10
20	9/10	5/10	8/10
721	5/10	10/10	
1155			10/10
20, 721	10/10	10/10	
20, 1155	10/10		10/10
721, 1155		10/10	10/10
20, 721, 1155	10/10	9/10	10/10

Comparação dos resultados

	Input		
Dem. Context	20	721	1155
ϵ	2 [2;4]	1 [9;9]	0
20	10 [0;3]	5 [1;4]	✓2 [8;9]
721	4 [1;5]	10 [0;0]	✓2 [6;9]
1155	5 [1;5]	6 [1;4]	✓10 [0;2]
20, 721	10 [0;7]	10 [0;1]	✓3 [7;9]
20, 1155	10 [0;5]	7 [1;2]	✓10 [0;0]
721, 1155	7 [1;7]	10 [0;4]	✓10 [0;0]
20, 721, 1155	9 [0;4]	10 [0;4]	✓10 [0;2]

Resultados anteriores

Dem. context	20	721	1155
ε	4/10	3/10	3/10
20	9/10	5/10	8/10
721	5/10	10/10	
1155			10/10
20, 721	10/10	10/10	
20, 1155	10/10		10/10
721, 1155		10/10	10/10
20, 721, 1155	10/10	9/10	10/10

Novos resultados

Conclusões

Resultado semelhante

Leve diferença a favor do novo modelo quando era aplicado com menos contexto.

Dificuldade em corrigir erros

Quando o modelo começava a errar, ele errava até o final. Acertava de primeira.

Necessidade de planejamento

Só de sabe da dificuldade quando passa. Mas um planejamento de custos teria evitado gastos desnecessários. Talvez um pivotamento precoce.

Experimentos mal sucedidos

qwen2.5-coder (7B, 14B)

Não acertou nada. Investimos em uma máquina com GPU a100 para conseguir rodar. Mas não resolveu nenhum caso.

qwen2.5 (7B, 14B)

Não acertou nada. Investimos em uma máquina com GPU a100 para conseguir rodar. Mas não resolveu nenhum caso.

deepseek-coder-v2 (16B)

Não acertou nada. Investimos em uma máquina com GPU a100 para conseguir rodar. Mas não resolveu nenhum caso.

gemini2.0-flash

Não acertou nada. A API é grátis com certos limites, mas o modelo não performa bem o suficiente para trazer insights valiosos.

Muito Obrigado!