

Relatório sobre Estratégias de Raciocínio em Agentes de IA

Sumário

- [1. Introdução](#)
- [2. Ajuste Fino Totalmente Supervisionado \(Fully Supervised Finetuning\)](#)
- [3. Prompting e In-context Learning](#)
- [4. Rationale Engineering](#)
- [5. Prompting em Cadeia de Pensamento \(Chain-of-Thought Prompting\)](#)
- [6. Amostragem de Autoconsistência \(Self-Consistency Sampling\)](#)
- [7. Decomposição de Problemas](#)
- [8. Raciocínio com Ferramentas Auxiliares \(Tool-Augmented Reasoning\)](#)
- [9. Raciocínio com Memória e Contexto \(Memory and Contextual Reasoning\)](#)
- [10. MCP \(Model Context Protocol\)](#)
- [11. Conclusão](#)
- [12. Referências Bibliográficas](#)

Introdução

As estratégias de raciocínio em modelos de linguagem representam abordagens que visam melhorar a capacidade desses sistemas em realizar inferências complexas, resolver problemas e tomar decisões de forma mais eficaz. Este relatório explora diversas técnicas emergentes nesse campo, apresentando suas definições, aplicações práticas e exemplos de implementação. Com a evolução dos modelos de linguagem de grande escala (LLMs), essas estratégias de raciocínio tornaram-se fundamentais para superar limitações, como falhas de raciocínio lógico, inconsistências e dificuldades em resolver problemas complexos.

Ajuste Fino Totalmente Supervisionado

Definição

O Ajuste Fino Totalmente Supervisionado (Fully Supervised Finetuning - FSF) é uma técnica onde um modelo pré-treinado é ajustado utilizando dados de alta qualidade e anotados com feedback humano, com o objetivo específico de alinhar o

comportamento do modelo com determinadas capacidades de raciocínio ou requisitos de desempenho.

Aplicações

- Especialização de modelos para domínios específicos como análise jurídica ou médica
- Redução de vieses e melhoria de respostas éticas
- Refinamento da capacidade de raciocínio matemático ou lógico
- Alinhamento com instruções complexas e preferências do usuário

Exemplo

Cenário: Ajuste fino para melhorar o raciocínio matemático

Implementação:

1. Coleta de um conjunto de dados de problemas matemáticos com soluções passo a passo verificadas
2. Preparação dos dados no formato de pares instrução-resposta:
 - a. Instrução: "Resolva a equação: $3x + 7 = 25$ "
 - b. Resposta: "Para resolver a equação $3x + 7 = 25$, precisamos isolar x .
Passo 1: Subtraia 7 de ambos os lados: $3x = 18$ Passo 2: Divida ambos os lados por 3: $x = 6$ A solução é $x = 6$."
3. Treinamento do modelo nos dados preparados por várias épocas
4. Avaliação usando um conjunto de teste separado de problemas matemáticos

O modelo ajustado demonstrou uma melhora significativa na resolução de problemas matemáticos, com um aumento de 67% na taxa de acerto em comparação com o modelo base.

Prompting e In-context Learning

Definição

Prompting e In-context Learning referem-se à capacidade de guiar um modelo de linguagem por meio de instruções (prompts) e exemplos dentro do próprio contexto de entrada, sem alterar os parâmetros do modelo. O modelo "aprende" a partir do contexto fornecido na entrada para realizar a tarefa especificada.

Aplicações

- Tradução e transformação de formatos de texto sem treino específico
- Adaptação rápida a tarefas novas sem necessidade de retreinamento
- Adaptação do estilo de resposta a diferentes contextos
- Implementação de classificadores de texto sem fine-tuning

Exemplo

Cenário: Classificação de sentimentos em comentários de produtos

Implementação:

Por favor, classifique os seguintes comentários como positivo, negativo ou neutro:

Exemplo 1:

Comentário: "Este produto superou todas as minhas expectativas! Recomendo fortemente."

Classificação: Positivo

Exemplo 2:

Comentário: "A qualidade deixa muito a desejar e o preço é alto demais para o que oferece."

Classificação: Negativo

Exemplo 3:

Comentário: "O produto chegou no prazo estipulado e funciona como descrito."

Classificação: Neutro

Agora, classifique este comentário:

"Comprei há uma semana e já está apresentando defeitos. Atendimento ao cliente péssimo."

Classificação:

Resposta esperada: "Negativo"

Este exemplo demonstra como o modelo pode aprender a tarefa de classificação de sentimento a partir dos exemplos fornecidos no prompt, sem necessidade de ajustes específicos em seus parâmetros.

Rationale Engineering

Definição

Rationale Engineering é a prática de projetar e ajustar as explicações ou justificativas que um modelo de IA fornece para suas previsões ou recomendações. Envolve a otimização da forma como o modelo articula seu processo de raciocínio.

Aplicações

- Melhoria da transparência em sistemas de suporte à decisão
- Aumento da confiabilidade de sistemas críticos
- Facilitação da depuração de modelos
- Demonstração do processo de pensamento em ambientes educacionais

Exemplo

Cenário: Sistema de recomendação de investimentos

Implementação:

Versão original:

"Recomendamos investir em ações da empresa XYZ."

Versão com Rationale Engineering:

"Recomendamos investir em ações da empresa XYZ pelos seguintes motivos:

1. A empresa apresentou crescimento consistente de receita nos últimos 5 trimestres (aumento médio de 12%)
2. A recente expansão para mercados internacionais abriu novas oportunidades de crescimento
3. O P/L atual de 15 está abaixo da média do setor (18)
4. A empresa anunciou um novo produto inovador com patente recém-aprovada
5. Analistas independentes preveem aumento de 25% no valor das ações nos próximos 12 meses

Os riscos potenciais incluem volatilidade no mercado de commodities e possível regulamentação mais rigorosa no setor."

Esta versão melhorada permite que o usuário compreenda o raciocínio completo por trás da recomendação, aumentando a transparência e confiabilidade do sistema.

Prompting em Cadeia de Pensamento

Definição

Chain-of-Thought Prompting (CoT) é uma técnica que incentiva modelos de linguagem a gerar passos intermediários de raciocínio antes de chegar a uma resposta final, simulando um processo de pensamento passo a passo, semelhante ao raciocínio humano.

Aplicações

- Resolução de problemas matemáticos complexos
- Raciocínio lógico e dedutivo
- Análise de múltiplas variáveis em processos decisórios
- Explicação de conceitos científicos complexos

Exemplo

Cenário: Resolução de um problema de lógica

Implementação:

Pergunta: Em uma sala há 7 adultos e algumas crianças. Se $\frac{2}{3}$ do total de pessoas na sala são adultos, quantas crianças estão na sala?

Raciocínio:

Passo 1: Vamos chamar o número total de pessoas na sala de n .

Passo 2: Sabemos que 7 pessoas são adultos.

Passo 3: Também sabemos que $\frac{2}{3}$ do total são adultos, então: $7 = (\frac{2}{3}) \times n$

Passo 4: Multiplicando ambos os lados por $\frac{3}{2}$: $7 \times (\frac{3}{2}) = n$

Passo 5: Calculando: $7 \times (\frac{3}{2}) = 10,5$. Como estamos falando de pessoas, e não podemos ter 0,5 pessoa, isto sugere um erro.

Passo 6: Revisando: se 7 adultos representam $\frac{2}{3}$ do total, então $7 = (\frac{2}{3}) \times n$

Passo 7: Isso significa que $n = 7 \div (\frac{2}{3}) = 7 \times (\frac{3}{2}) = 10,5$

Passo 8: Como não podemos ter 10,5 pessoas, vamos verificar se há um erro de arredondamento. Se tivermos 10 pessoas no total, $\frac{2}{3}$ seriam

6,67 adultos, o que não corresponde aos 7 adultos.

Passo 9: Se tivermos 11 pessoas no total, $\frac{2}{3}$ seriam aproximadamente 7,33 adultos, o que também não corresponde exatamente.

Passo 10: Se tivermos 12 pessoas, $\frac{2}{3}$ seriam exatamente 8 adultos. Isso não corresponde aos 7 adultos dados.

Passo 11: Vamos resolver algebricamente: $\frac{7}{n} = \frac{2}{3}$

Passo 12: Cross-multiplication: $7 \times 3 = n \times 2$

Passo 13: $21 = 2n$

Passo 14: $n = 10,5$

Passo 15: Como não podemos ter um número fracionário de pessoas, isso sugere que a formulação do problema tem uma imprecisão ou que a fração $\frac{2}{3}$ é aproximada.

Passo 16: Se tentarmos números inteiros próximos: para $n = 10$, temos $\frac{7}{10} = 0,7$ ou 70%, que é próximo mas não igual a $\frac{2}{3}$ (aproximadamente 66,7%).

Passo 17: Para $n = 11$, temos $\frac{7}{11} \approx 63,6\%$, que é mais próximo de $\frac{2}{3}$.

Passo 18: Para $n = 12$, temos $\frac{7}{12} \approx 58,3\%$, que está mais distante.

Passo 19: Vamos assumir que a fração mais próxima de um número inteiro é a correta: $n = 10,5$ está entre 10 e 11. Como $n = 11$ dá uma fração mais próxima de $\frac{2}{3}$, vamos usar $n = 11$.

Passo 20: Se há 11 pessoas no total e 7 são adultos, então há $11 - 7 = 4$ crianças.

Resposta: Há 4 crianças na sala.

Este exemplo demonstra como o modelo explicita todo o processo de raciocínio, incluindo tentativas, correções e refinamentos até chegar à solução final.

Amostragem de Autoconsistência

Definição

Self-Consistency Sampling é uma técnica que envolve gerar múltiplas cadeias de raciocínio independentes para o mesmo problema e depois selecionar a resposta mais consistente entre essas cadeias, aumentando a confiabilidade das respostas.

Aplicações

- Problemas matemáticos com múltiplos caminhos de solução
- Tomada de decisão em condições de incerteza

- Verificação de consistência lógica em raciocínios complexos
- Aumento da robustez em sistemas críticos

Exemplo

Cenário: Resolução de um problema aritmético

Implementação:

Questão: Se um trem viaja a 120 km/h e percorre 240 km, quanto tempo leva a viagem?

Cadeia de raciocínio 1:

Para encontrar o tempo, divido a distância pela velocidade.

$\text{Tempo} = 240 \text{ km} \div 120 \text{ km/h} = 2 \text{ horas.}$

Cadeia de raciocínio 2:

Usando a fórmula $t = d/v$, onde t é tempo, d é distância e v é velocidade.

$t = 240/120 = 2 \text{ horas.}$

Cadeia de raciocínio 3:

Se o trem percorre 120 km em 1 hora, então para percorrer 240 km, precisará de:

$240 \div 120 = 2 \text{ horas.}$

Cadeia de raciocínio 4:

Vamos usar proporção:

120 km \rightarrow 1 h

240 km \rightarrow x h

$x = (240 \times 1) \div 120 = 2 \text{ horas.}$

Cadeia de raciocínio 5:

A velocidade é distância/tempo, então tempo = distância/velocidade.

$\text{Tempo} = 240/120 = 2 \text{ horas.}$

Resposta por consenso: 2 horas

Este exemplo mostra como, ao gerar múltiplas linhas de raciocínio independentes que convergem para a mesma resposta (2 horas), aumentamos a confiança na solução.

Decomposição de Problemas

Definição

A Decomposição de Problemas é uma estratégia que divide problemas complexos em subproblemas menores e mais gerenciáveis, resolvendo cada um separadamente antes de combinar as soluções para responder à questão original.

Aplicações

- Planejamento de projetos complexos
- Resolução de problemas multidisciplinares
- Quebra de tarefas de raciocínio em etapas sequenciais
- Organização de pensamento em tópicos complexos

Exemplo

Cenário: Análise do impacto ambiental de um novo produto

Implementação:

Problema principal: Qual é o impacto ambiental da produção e uso de um carro elétrico em comparação com um carro a gasolina ao longo de 10 anos?

Decomposição:

1. Subproblema 1: Impacto ambiental da fabricação

- Questão 1.1: Quanta energia é consumida na fabricação de cada tipo de veículo?
- Questão 1.2: Quais materiais específicos são usados em cada tipo de veículo e qual é o impacto ambiental da extração e processamento desses materiais?
- Questão 1.3: Qual é a pegada de carbono da fabricação de baterias para carros elétricos?

2. Subproblema 2: Impacto ambiental do uso

- Questão 2.1: Qual é a emissão média de CO₂ de um carro a gasolina ao longo de 10 anos, considerando uma quilometragem média anual?
- Questão 2.2: Qual é o impacto ambiental da geração de eletricidade necessária para abastecer um carro elétrico por 10 anos?

- Questão 2.3: Como a fonte de eletricidade (renovável vs. não renovável) afeta este cálculo?

3. Subproblema 3: Impacto do fim de vida útil

- Questão 3.1: Quais são as opções de reciclagem para cada tipo de veículo?

- Questão 3.2: Qual é o impacto ambiental do descarte de baterias de carros elétricos?

- Questão 3.3: Quais componentes de cada tipo de veículo podem ser reutilizados?

4. Integração dos resultados

- Análise combinada dos três subproblemas

- Cálculo do impacto total ao longo do ciclo de vida

- Comparação final entre os dois tipos de veículos

Este exemplo demonstra como um problema complexo de análise de impacto ambiental pode ser decomposto em componentes mais gerenciáveis, facilitando uma análise mais completa e estruturada.

Raciocínio com Ferramentas Auxiliares

Definição

Tool-Augmented Reasoning refere-se à estratégia de integrar ferramentas externas (como calculadoras, bases de conhecimento, APIs, ou sistemas de busca) ao processo de raciocínio do modelo de IA, permitindo que ele acesse recursos adicionais para melhorar suas capacidades de resolução de problemas.

Aplicações

- Solução de problemas matemáticos complexos com auxílio de calculadoras
- Acesso a informações atualizadas via APIs de pesquisa
- Integração com sistemas especializados para tarefas específicas
- Verificação de fatos usando bases de conhecimento externas

Exemplo

Cenário: Assistente de planejamento financeiro utilizando ferramentas

Implementação:

Usuário: "Quanto preciso investir mensalmente para acumular R\$ 500.000 em 20 anos, assumindo um retorno anual de 8%?"

Sistema (processo interno):

1. Identifica necessidade de cálculo financeiro complexo
2. Chama ferramenta auxiliar: Calculadora de investimentos
 - Parâmetros:
 - Valor alvo: R\$ 500.000
 - Período: 20 anos (240 meses)
 - Taxa de juros: 8% a.a. (0,64% a.m.)
 - Resultado da calculadora: R\$ 866,44 mensais
3. Verifica resultado usando fórmula alternativa:
 - $PMT = FV / ((1 + r)^n - 1) / r \times (1 + r)$
 - Onde FV = valor futuro, r = taxa mensal, n = número de meses
 - Resultado: R\$ 866,44 mensais
4. Consulta base de dados de inflação histórica para contextualizar:
 - Inflação média no Brasil nos últimos 10 anos: 5,3% a.a.
 - Retorno real considerando inflação: aproximadamente 2,7% a.a.

Resposta ao usuário:

"Para acumular R\$ 500.000 em 20 anos com um retorno anual de 8%, você precisaria investir aproximadamente R\$ 866,44 por mês.

Vale notar que este cálculo considera um retorno nominal. Considerando a inflação histórica média brasileira de 5,3%, o retorno real seria de aproximadamente 2,7%, o que significa que o poder de compra dos R\$ 500.000 no futuro seria equivalente a cerca de R\$ 275.000 em valores atuais."

Este exemplo demonstra como o modelo utiliza ferramentas externas (calculadora financeira e base de dados de inflação) para fornecer uma resposta mais precisa e contextualizada.

Raciocínio com Memória e Contexto

Definição

Memory and Contextual Reasoning é uma abordagem que permite ao modelo manter, acessar e utilizar informações de interações anteriores ou conhecimentos

armazenados para informar o raciocínio atual, possibilitando uma compreensão mais profunda do contexto e personalização do raciocínio.

Aplicações

- Sistemas de atendimento ao cliente com histórico de interações
- Assistentes pessoais que aprendem preferências do usuário
- Sistemas educacionais adaptáveis ao progresso do aluno
- Sistemas de recomendação personalizados

Exemplo

Cenário: Assistente pessoal para planejamento de refeições

Implementação:

[Histórico armazenado na memória]

- Usuário informou alergia a amendoim (3 meses atrás)
- Usuário prefere refeições vegetarianas (1 mês atrás)
- Usuário está tentando reduzir carboidratos (2 semanas atrás)
- Usuário gostou especialmente de uma receita de curry de grão-de-bico (1 semana atrás)

[Interação atual]

Usuário: "Pode me sugerir algo para o jantar de hoje?"

Sistema (processo interno):

1. Recupera preferências relevantes da memória
2. Cruza informações para criar personalização
3. Prioriza restrições de saúde (alergia)
4. Aplica preferências dietéticas recentes

Resposta:

"Considerando suas preferências, sugiro uma salada de quinoa com legumes assados e tofu grelhado. É vegetariana, relativamente baixa em carboidratos (especialmente se você moderar a quantidade de quinoa), e não contém amendoim. A preparação tem um perfil de sabor semelhante ao curry de grão-de-bico que você gostou na semana passada, com cominho e cúrcuma. Gostaria da receita detalhada?"

Este exemplo mostra como o sistema utiliza seu conhecimento acumulado sobre o usuário para fornecer sugestões personalizadas, demonstrando compreensão contextual e aprendizado contínuo.

MCP (Model Context Protocol)

Definição

O Model Context Protocol (MCP) é uma arquitetura estruturada para organizar o contexto interno de um modelo, geralmente implementada como um conjunto de instruções específicas sobre como o modelo deve processar informações, raciocinar e responder em diferentes cenários.

Aplicações

- Padronização da interação com modelos de linguagem
- Definição de regras de engajamento para assistentes virtuais
- Estabelecimento de protocolos de segurança e ética
- Implementação de fluxos de raciocínio específicos para tarefas complexas

Exemplo

Cenário: Protocolo para análise financeira

Implementação:

[Instruções de MCP para Análise Financeira]

1. Contexto Inicial:

- Identificar o tipo de requisição (análise de investimento, orçamento pessoal, etc.)
- Verificar quais informações estão disponíveis e quais estão faltando
- Determinar o nível de conhecimento do usuário para ajustar o nível técnico da resposta

2. Protocolo de Raciocínio:

- Iniciar sempre com análise de risco antes de considerar retornos potenciais
- Considerar sempre o horizonte temporal mencionado ou inferido
- Incluir análise de cenários (otimista, pessimista, mais provável)

- Verificar consistência matemática de todos os cálculos antes de apresentar conclusões

3. Formato de Resposta:

- Iniciar com um resumo executivo em linguagem simples
- Estruturar a análise detalhada em seções claramente demarcadas
- Incluir sempre uma seção de "Considerações Importantes" para limitações e avisos
- Concluir com recomendações práticas e próximos passos sugeridos

4. Regras de Segurança:

- Nunca recomendar produtos financeiros específicos
- Sempre indicar que a análise não constitui consultoria financeira regulamentada
- Não solicitar nem armazenar dados financeiros sensíveis
- Alertar sobre a necessidade de consultar um profissional qualificado para decisões importantes

Aplicando este protocolo a uma consulta do usuário:

Usuário: "Vale a pena investir em criptomoedas atualmente?"

Resposta (seguindo o MCP):

[Resumo Executivo]

O investimento em criptomoedas apresenta potencial de retorno elevado, mas com riscos igualmente significativos, exigindo análise cuidadosa de sua adequação ao seu perfil e objetivos financeiros.

[Análise de Risco]

- Volatilidade: As criptomoedas frequentemente apresentam oscilações de mais de 10% em um único dia
- Regulatório: O ambiente regulatório ainda está em evolução em muitos países
- Segurança: Riscos de hackers, perda de chaves privadas e fraudes

[Análise de Cenários]

- Cenário Pessimista: Regulação restritiva global poderia reduzir significativamente o valor
- Cenário Mais Provável: Continuidade da adoção gradual com volatilidade persistente
- Cenário Otimista: Adoção mais ampla como meio de pagamento e

reserva de valor

[Considerações Importantes]

- Esta análise não constitui recomendação de investimento
- O mercado de criptomoedas é relativamente novo e com histórico limitado
- A tecnologia subjacente (blockchain) tem aplicações além das moedas digitais

[Próximos Passos Recomendados]

1. Educar-se sobre os fundamentos da tecnologia blockchain
2. Considerar iniciar com uma pequena alocação (geralmente recomenda-se não mais que 5% do portfólio)
3. Consultar um consultor financeiro qualificado antes de fazer investimentos significativos
4. Considerar a diversificação entre diferentes criptoativos se decidir investir

Este exemplo demonstra como o MCP fornece uma estrutura consistente para o modelo abordar consultas complexas, garantindo que todas as considerações importantes sejam abordadas de maneira sistemática.

Conclusão

As estratégias de raciocínio em agentes de IA representam um campo em rápida evolução, com abordagens inovadoras sendo constantemente desenvolvidas para melhorar a capacidade desses sistemas de enfrentar problemas complexos. Da estruturação explícita do pensamento (Chain-of-Thought) à integração com ferramentas externas (Tool-Augmented Reasoning), cada técnica oferece vantagens específicas para diferentes cenários e aplicações.

À medida que os modelos de linguagem continuam a evoluir, a combinação dessas estratégias promete criar sistemas cada vez mais capazes de raciocínio nuançado, contextualizado e confiável. O futuro dos agentes de IA dependerá significativamente do refinamento dessas abordagens e do desenvolvimento de novas técnicas que superem as limitações atuais.

Para pesquisadores e desenvolvedores, a compreensão e implementação deliberada dessas estratégias é essencial para criar sistemas de IA que possam verdadeiramente auxiliar em tarefas que exigem raciocínio sofisticado e tomada de decisão complexa.

Referências Bibliográficas

Antropic. (2023). Constitutional AI: Harmlessness from AI Feedback. *arXiv preprint arXiv:2212.08073*.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33, 1877-1901.

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*.

Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., & Iwasawa, Y. (2022). Large language models are zero-shot reasoners. *Advances in Neural Information Processing Systems*, 35.

Nye, M., Andreassen, A., Gur-Ari, G., Michalewski, H., Austin, J., Bieber, D., ... & Sutskever, I. (2021). Show your work: Scratchpads for intermediate computation with language models. *arXiv preprint arXiv:2112.00114*.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., ... & Lowe, R. (2022). Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35, 27730-27744.

Wang, X., Wei, J., Schuurmans, D., Le, Q., Chi, E., & Zhou, D. (2022). Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., ... & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35.

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., & Cao, Y. (2022). React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*.

Zhang, S., Roller, S., Goyal, N., Artetxe, M., Chen, M., Chen, S., ... & Zettlemoyer, L. (2022). Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

Zhou, D., Schärli, N., Hou, L., Wei, J., Scales, N., Wang, X., ... & Chi, E. H. (2023). Least-to-most prompting enables complex reasoning in large language models. *arXiv preprint arXiv:2205.10625*.

