

Lab 079: Deep Research Agents ? Síntese de Conhecimento em Múltiplas Etapas

Nível: L300 Trilha: Todas as trilhas Tempo: ~90 min ? Custo: Grátis ? Usa dados simulados de rastreamento de pesquisa

info "Tipo de prática" Este lab analisa um trace simulado de deep research. Ele ensina o fluxo multiagente de pesquisa e métricas de qualidade sem executar um agente de pesquisa web ao vivo.

O que Você Vai Aprender

Como Deep Research Agents usam um pipeline multi-agente para síntese de conhecimento A arquitetura Planner ? Researcher ? Writer ? Reviewer e as responsabilidades de cada papel Como o rastreamento de citações garante que cada afirmação seja mapeada de volta a uma fonte Analisar um rastreamento de pesquisa de 14 etapas com papéis de agentes, uso de tokens e pontuações de qualidade Identificar gargalos, distribuição de tokens e padrões de qualidade ao longo do pipeline

Introdução

Deep Research Agents implementam um pipeline de múltiplas etapas para produzir relatórios de pesquisa abrangentes e bem referenciados. Em vez de um único LLM gerar um relatório inteiro, o trabalho é dividido entre agentes especializados:

O Pipeline

[diagram or code sample]

Agente | Papel | Saída Principal | Planner | Decompõe a pergunta de pesquisa em sub-perguntas e cria um plano de pesquisa | Sub-perguntas, estratégia de busca | Researcher | Executa buscas, lê fontes, extrai descobertas principais com citações | Descobertas com citações de fontes | Writer | Sintetiza descobertas em um relatório coerente e bem estruturado | Rascunho do relatório com citações inline | Reviewer | Revisa o rascunho quanto à precisão, completude e qualidade das citações | Feedback, pontuação de qualidade, aprovação/revisão |

Rastreamento de Citações

Cada afirmação no relatório final deve ser rastreável até uma fonte. O pipeline rastreia: sources_cited: Número de fontes únicas citadas em cada etapa quality_score: Autoavaliação de qualidade da saída pelo agente (0.0?1.0)

O Cenário

Você é um Líder de Equipe de Pesquisa avaliando um sistema de agentes de pesquisa profunda. Você tem um rastreamento de pesquisa de 14 etapas (research_trace.csv) de uma execução de pesquisa concluída. Sua tarefa: analisar o rastreamento para entender o comportamento dos agentes, uso de tokens, padrões de qualidade e identificar oportunidades de otimização.

info "Dados Simulados" Este laboratório usa um CSV de rastreamento de pesquisa simulado. Os dados representam uma execução realista de pesquisa profunda com 14 etapas em 4 papéis de agentes, incluindo planejamento, pesquisa multi-fonte, escrita e revisão iterativa.

Pré-requisitos

Requisito | Por quê | Python 3.10+ | Executar os scripts de análise | Biblioteca pandas | Manipulação de dados |

[diagram or code sample]

tip "Início Rápido com GitHub Codespaces" Open in GitHub Codespaces

Todas as dependências estão pré-instaladas no devcontainer.

? Arquivos de Apoio

note "Baixe estes arquivos antes de iniciar o laboratório" Salve todos os arquivos em uma pasta lab-079/ no seu diretório de trabalho.

Arquivo | Descrição | Download | broken_research.py | Exercício de correção de bugs (3 bugs)

+ autotestes) | ? Download | research_trace.csv | Rastreamento de pesquisa de 14 etapas com papéis de agentes, tokens e qualidade | ? Download |

Etapa 1: Entenda o Formato do Rastreamento

Cada linha no rastreamento representa uma etapa no pipeline de pesquisa:

Coluna | Descrição | step_id | Número sequencial da etapa (1?14) | agent_role | Qual agente executou esta etapa: planner, researcher, writer, reviewer | action | O que o agente fez (ex.: decompose_query, search_sources, write_section) | tokens_used | Número de tokens consumidos nesta etapa | sources_cited | Número de fontes citadas na saída desta etapa | quality_score | Avaliação de qualidade da saída desta etapa (0.0?1.0) | duration_sec | Tempo gasto nesta etapa em segundos |

Etapa 2: Carregue e Explore o Rastreamento

[diagram or code sample]

Saída esperada:

[diagram or code sample]

Etapa 3: Analise o Uso de Tokens por Agente

[diagram or code sample]

[diagram or code sample]

tip "Insight de Otimização" O Researcher normalmente consome a maior quantidade de tokens porque processa múltiplas fontes por sub-pergunta. Para reduzir custos, considere armazenar em cache as extrações de fontes e limitar o número de fontes por sub-pergunta.

Etapa 4: Analise o Fluxo de Citações

[diagram or code sample]

Etapa 5: Análise de Qualidade

[diagram or code sample]

warning "Variância de Qualidade" Fique atento a quedas de qualidade nas etapas posteriores do Researcher ? isso frequentemente indica esgotamento de fontes (o agente está encontrando fontes de menor qualidade para sub-perguntas mais difíceis). Considere adicionar um limiar de qualidade que acione uma nova busca com consultas alternativas.

Etapa 6: Construa o Relatório de Análise da Pesquisa

[diagram or code sample]

? Exercício de Correção de Bugs

O arquivo lab-079/broken_research.py contém 3 bugs que produzem análises de pesquisa incorretas. Você consegue encontrar e corrigir todos?

Execute os autotestes para ver quais falham:

[diagram or code sample]

Você deve ver 3 testes com falha. Cada teste corresponde a um bug:

Teste | O que verifica | Dica | Teste 1 | Total de fontes citadas | Deve somar sources_cited, não contar linhas | Teste 2 | Contagem de tokens do Writer | Deve filtrar agent_role == "writer", não "researcher" | Teste 3 | Contagem de etapas do Researcher | Deve contar linhas onde agent_role == "researcher", não somar tokens |

Corrija todos os 3 bugs e execute novamente. Quando você vir All passed!, está pronto!

? Verificação de Conhecimento

question "Q1 (Múltipla Escolha): Qual é a principal vantagem de um pipeline multi-agente em relação a uma abordagem de LLM único para pesquisa?"

- A) Usa menos tokens no total - B) Cada agente se especializa em uma tarefa, permitindo melhor qualidade e rastreabilidade - C) Requer apenas uma implantação de modelo - D)

Elimina a necessidade de citações

??? success "? Revelar Resposta" Correto: B) Cada agente se especializa em uma tarefa,

permitindo melhor qualidade e rastreabilidade

Ao dividir a pesquisa em planejamento, busca, escrita e revisão, cada agente pode ser otimizado para sua tarefa específica. O Researcher pode focar na qualidade das fontes, o Writer na coerência da prosa e o Reviewer na precisão factual. Essa especialização normalmente produz saídas de maior qualidade do que uma geração única de ponta a ponta.

question "Q2 (Múltipla Escolha): Por que o rastreamento de citações é importante em agentes de pesquisa profunda?"

- A) Reduz o uso de tokens - B) Garante que cada afirmação seja mapeada de volta a uma fonte, permitindo verificação e confiança - C) Torna o relatório mais longo - D) É exigido pelos termos de serviço do LLM

??? success "? Revelar Resposta" Correto: B) Garante que cada afirmação seja mapeada de volta a uma fonte, permitindo verificação e confiança

O rastreamento de citações cria uma cadeia auditável de cada afirmação no relatório final até sua fonte. Isso permite que revisores verifiquem a precisão factual, que usuários explorem fontes primárias e que organizações mantenham a integridade da pesquisa ? crítico para aplicações de alto risco como pesquisa jurídica, médica ou financeira.

question "Q3 (Execute o Laboratório): Qual é o número total de fontes citadas em todas as etapas?"

Execute a análise da Etapa 4 em ? research_trace.csv e some a coluna sources_cited.

??? success "? Revelar Resposta" 10 fontes

A soma de todos os valores de sources_cited nas 14 etapas é igual a 10. A maioria das fontes é citada durante as etapas do Researcher, com algumas citações adicionais incluídas durante a síntese do Writer.

question "Q4 (Execute o Laboratório): Quantos tokens no total o agente Writer consumiu?"

Execute a análise da Etapa 3 e encontre o total de tokens para o papel writer.

??? success "? Revelar Resposta" Soma de tokens_used onde agent_role == "writer"

A contagem total de tokens do Writer inclui todas as etapas de escrita e síntese. Filtre o rastreamento por agent_role == "writer" e some a coluna tokens_used para obter o valor exato.

question "Q5 (Execute o Laboratório): Quantas etapas o agente Researcher executou?"

Conte as linhas onde agent_role == "researcher".

??? success "? Revelar Resposta" 6 etapas

O Researcher executou 6 etapas ? o maior número de qualquer papel de agente. Isso faz sentido porque o Researcher lida com múltiplas sub-perguntas do Planner, com cada sub-pergunta potencialmente exigindo múltiplas etapas de busca e extração.

Resumo

Tópico | O que Você Aprendeu | Deep Research Agents | Pipeline multi-agente para síntese de conhecimento com rastreamento de citações | Arquitetura do Pipeline | Planner ? Researcher ? Writer ? Reviewer com papéis especializados | Rastreamento de Citações | Cada afirmação é mapeada de volta a uma fonte ao longo do pipeline | Distribuição de Tokens | Researcher usa mais tokens; Writer sintetiza; Reviewer valida | Padrões de Qualidade | A qualidade varia por etapa ? etapas de pesquisa posteriores podem mostrar esgotamento de fontes | Otimização | Armazenar fontes em cache, paralelizar pesquisa, adicionar portões de qualidade |

Próximos Passos

Lab 034 ? Multi-Agent com Semantic Kernel (construa os agentes) Lab 067 ? GraphRAG

(aprimore a pesquisa com recuperação por grafo de conhecimento) Lab 033 ? Observabilidade de Agentes (monitore pipelines de pesquisa profunda em produção) Lab 076 ? Microsoft Agent Framework (implemente pipelines com MAF Graph Workflows)

Phase 3 · Simulação / Caminho Real

Simulação Caminho Real

Preserve the existing free simulation and its fixtures. Estimated cost: US\$0.

Reuse the same acceptance criteria against the documented hosted service. Before running, record a service-specific estimate and a hard spending cap. After validation: Exclua implantações, endpoints, índices, traces e dados de teste após a validação.

Kit de apoio

O build do site gera downloads estáveis em PDF e ZIP. O ZIP contém o guia do lab e todos os arquivos de apoio declarados para este lab.

Baixar PDF gerado Baixar ZIP gerado

Supporting file: broken_research.py

```
/usr/bin/env python3 """Bug-fix: Broken Research Analyzer - Lab 079.""" import io, pandas
as pd def total_sources_cited(trace): return trace["sources_found"].sum() def
writer_token_usage(trace): return trace["tokens_used"].mean() def avg_quality(trace):
return trace["quality_score"].min() def run_tests():
csv="step_id,agent,tokens_used,sources_found,sources_cited,quality_score\nS1,Planner,280,0,0
,0.95\nS2,Researcher,0,12,0,0.90\nS3,Writer,1200,0,5,0.92\nS4,Reviewer,600,0,5,0.88"
t=pd.read_csv(io.StringIO(csv));p,f=0,0 s=total_sources_cited(t);exp=10 if
s==exp:print(f"OK T1: {s}");p+=1 else:print(f"FAIL T1: {s} (exp {exp})");f+=1
w=writer_token_usage(t);exp=1200.0 if abs(w-exp)<1:print(f"OK T2: {w:.0f}");p+=1
else:print(f"FAIL T2: {w:.0f} (exp {exp:.0f})");f+=1 a=avg_quality(t);exp=0.9125 if
abs(a-exp)<0.01:print(f"OK T3: {a:.3f}");p+=1 else:print(f"FAIL T3: {a:.3f} (exp
{exp:.3f})");f+=1 print(f"\n{'All passed!' if f==0 else f'failed'}") if
__name__=="__main__":print("Tests:\n");run_tests()
```

Supporting file: research_trace.csv

```
step_id,agent,action,input_type,output_type,duration_sec,tokens_used,sources_found,sources_c
ited,quality_score S01,Planner,decompose_query,user_question,research_plan,3.2,280,0,0,0.95
S02,Researcher,web_search,search_query_1,raw_results,5.8,0,12,0,0.90
S03,Researcher,web_search,search_query_2,raw_results,4.5,0,8,0,0.88
S04,Researcher,web_search,search_query_3,raw_results,6.2,0,15,0,0.92
S05,Researcher,read_and_extract,url_1,extracted_facts,8.0,450,1,1,0.85
S06,Researcher,read_and_extract,url_2,extracted_facts,7.5,420,1,1,0.88
S07,Researcher,read_and_extract,url_3,extracted_facts,9.2,510,1,1,0.82
S08,Researcher,read_and_extract,url_4,extracted_facts,6.8,380,1,1,0.90
S09,Researcher,read_and_extract,url_5,extracted_facts,7.0,400,1,1,0.87
S10,Writer,synthesize_report,all_facts,draft_report,15.0,1200,0,5,0.92
S11,Reviewer,check_citations,draft_report,review_notes,8.5,600,0,5,0.88
S12,Reviewer,check_accuracy,draft_report,accuracy_report,10.0,700,0,5,0.85
S13,Writer,revise_report,review_notes,final_report,12.0,900,0,5,0.94
S14,Planner,quality_gate,final_report,approval,2.0,150,0,0,0.96
```