

Lab 079 : Agents de recherche approfondie ? Synthèse de connaissances multi-étapes
Niveau : L300 Parcours : Tous les parcours Durée : ~90 min ? Coût : Gratuit ? Utilise des données de trace de recherche simulées

info "Type de pratique" Ce lab analyse une trace deep research simulée. Il enseigne le workflow de recherche multi-agent et les métriques de qualité sans exécuter un agent de recherche web réel.

Ce que vous apprendrez

Comment les agents de recherche approfondie utilisent un pipeline multi-agents pour la synthèse de connaissances L'architecture Planificateur ? Chercheur ? Rédacteur ? Réviseur et les responsabilités de chaque rôle Comment le suivi des citations garantit que chaque affirmation est reliée à une source Analyser une trace de recherche en 14 étapes avec les rôles d'agents, l'utilisation de tokens et les scores de qualité Identifier les goulots d'étranglement, la distribution des tokens et les schémas de qualité dans le pipeline

Introduction

Les agents de recherche approfondie implémentent un pipeline multi-étapes pour produire des rapports de recherche complets et bien sourcés. Au lieu qu'un seul LLM génère un rapport entier, le travail est réparti entre des agents spécialisés :

Le pipeline

[diagram or code sample]

Agent | Rôle | Sortie clé | Planner | Décompose la question de recherche en sous-questions et crée un plan de recherche | Sous-questions, stratégie de recherche | Researcher | Exécute les recherches, lit les sources, extrait les résultats clés avec citations | Résultats avec citations de sources | Writer | Synthétise les résultats en un rapport cohérent et bien structuré | Brouillon de rapport avec citations en ligne | Reviewer | Révise le brouillon pour la précision, l'exhaustivité et la qualité des citations | Retour d'information, score de qualité, approbation/révision |

Suivi des citations

Chaque affirmation du rapport final doit être reliée à une source. Le pipeline suit :
sources_cited : Nombre de sources uniques citées à chaque étape quality_score :
Auto-évaluation de la qualité de la sortie par l'agent (0.0?1.0)

Le scénario

Vous êtes un responsable d'équipe de recherche évaluant un système d'agent de recherche approfondie. Vous disposez d'une trace de recherche en 14 étapes (research_trace.csv) provenant d'une exécution de recherche terminée. Votre mission : analyser la trace pour comprendre le comportement des agents, l'utilisation des tokens, les schémas de qualité et identifier les opportunités d'optimisation.

info "Données simulées" Ce lab utilise un fichier CSV de trace de recherche simulé. Les données représentent une exécution de recherche approfondie réaliste avec 14 étapes réparties sur 4 rôles d'agents, incluant la planification, la recherche multi-sources, la rédaction et la révision itérative.

Prérequis

Exigence | Pourquoi | Python 3.10+ | Exécuter les scripts d'analyse | Bibliothèque pandas | Manipulation des données |

[diagram or code sample]

tip "Démarrage rapide avec GitHub Codespaces" Open in GitHub Codespaces

Toutes les dépendances sont pré-installées dans le devcontainer.

? Fichiers de support

note "Téléchargez ces fichiers avant de commencer le lab" Enregistrez tous les fichiers

dans un dossier lab-079/ de votre répertoire de travail.

Fichier | Description | Téléchargement | [broken_research.py](#) | Exercice de correction de bugs (3 bugs + auto-tests) | ? Télécharger | [research_trace.csv](#) | Trace de recherche en 14 étapes avec rôles d'agents, tokens et qualité | ? Télécharger |

Étape 1 : Comprendre le format de la trace

Chaque ligne de la trace représente une étape du pipeline de recherche :

Colonne | Description | step_id | Numéro séquentiel de l'étape (1?14) | agent_role | Quel

agent a exécuté cette étape : planner, researcher, writer, reviewer | action | Ce que

l'agent a fait (ex. : decompose_query, search_sources, write_section) | tokens_used |

Nombre de tokens consommés à cette étape | sources_cited | Nombre de sources citées dans la sortie de cette étape | quality_score | Évaluation de la qualité de la sortie de cette

étape (0.0?1.0) | duration_sec | Temps pris pour cette étape en secondes |

Étape 2 : Charger et explorer la trace

[diagram or code sample]

Sortie attendue :

[diagram or code sample]

Étape 3 : Analyser l'utilisation des tokens par agent

[diagram or code sample]

[diagram or code sample]

tip "Piste d'optimisation" Le Researcher consomme généralement le plus de tokens car il traite plusieurs sources par sous-question. Pour réduire les coûts, envisagez de mettre en cache les extractions de sources et de limiter le nombre de sources par sous-question.

Étape 4 : Analyser le flux de citations

[diagram or code sample]

Étape 5 : Analyse de la qualité

[diagram or code sample]

warning "Variance de qualité" Surveillez les baisses de qualité dans les étapes Researcher ultérieures ? cela indique souvent un épuisement des sources (l'agent trouve des sources de moindre qualité pour les sous-questions plus difficiles). Envisagez d'ajouter un seuil de qualité qui déclenche une nouvelle recherche avec des requêtes alternatives.

Étape 6 : Construire le rapport d'analyse de recherche

[diagram or code sample]

? Exercice de correction de bugs

Le fichier lab-079/broken_research.py contient 3 bugs qui produisent une analyse de recherche incorrecte. Pouvez-vous les trouver et les corriger tous ?

Exécutez les auto-tests pour voir lesquels échouent :

[diagram or code sample]

Vous devriez voir 3 tests échoués. Chaque test correspond à un bug :

Test | Ce qu'il vérifie | Indice | Test 1 | Total des sources citées | Devrait sommer

sources_cited, pas compter les lignes | Test 2 | Nombre de tokens du Writer | Devrait

filtrer agent_role == "writer", pas "researcher" | Test 3 | Nombre d'étapes du Researcher |

Devrait compter les lignes où agent_role == "researcher", pas sommer les tokens |

Corrigez les 3 bugs, puis relancez. Quand vous voyez All passed!, c'est terminé !

? Vérification des connaissances

question "Q1 (Choix multiple) : Quel est le principal avantage d'un pipeline multi-agents par rapport à une approche LLM unique pour la recherche ?"

- A) Il utilise moins de tokens au total - B) Chaque agent se spécialise dans une tâche,

permettant une meilleure qualité et traçabilité - C) Il ne nécessite qu'un seul déploiement

de modèle - D) Il élimine le besoin de citations

??? success "?" Révéler la réponse" Correct : B) Chaque agent se spécialise dans une tâche, permettant une meilleure qualité et traçabilité

En divisant la recherche en planification, recherche, rédaction et révision, chaque agent peut être optimisé pour sa tâche spécifique. Le Researcher peut se concentrer sur la qualité des sources, le Writer sur la cohérence de la prose, et le Reviewer sur la précision factuelle. Cette spécialisation produit généralement une sortie de meilleure qualité qu'une génération unique de bout en bout.

question "Q2 (Choix multiple) : Pourquoi le suivi des citations est-il important dans les agents de recherche approfondie ?"

- A) Il réduit l'utilisation des tokens - B) Il garantit que chaque affirmation est reliée à une source, permettant la vérification et la confiance - C) Il rend le rapport plus long - D) Il est exigé par les conditions d'utilisation du LLM

??? success "?" Révéler la réponse" Correct : B) Il garantit que chaque affirmation est reliée à une source, permettant la vérification et la confiance

Le suivi des citations crée une chaîne auditable de chaque affirmation du rapport final jusqu'à sa source. Cela permet aux réviseurs de vérifier la précision factuelle, aux utilisateurs d'explorer les sources primaires et aux organisations de maintenir l'intégrité de la recherche ? critique pour les applications à enjeux élevés comme la recherche juridique, médicale ou financière.

question "Q3 (Exécutez le lab) : Quel est le nombre total de sources citées dans toutes les étapes ?"

Exécutez l'analyse de l'étape 4 sur ? research_trace.csv et summez la colonne sources_cited.

??? success "?" Révéler la réponse" 10 sources

La somme de toutes les valeurs sources_cited sur les 14 étapes est de 10. La plupart des sources sont citées pendant les étapes du Researcher, avec quelques citations supplémentaires ajoutées lors de la synthèse du Writer.

question "Q4 (Exécutez le lab) : Combien de tokens au total l'agent Writer a-t-il consommé ?"

Exécutez l'analyse de l'étape 3 et trouvez le total de tokens pour le rôle writer.

??? success "?" Révéler la réponse" Somme de tokens_used où agent_role == "writer"

Le total de tokens du Writer inclut toutes les étapes d'écriture et de synthèse. Filtrez la trace pour agent_role == "writer" et summez la colonne tokens_used pour obtenir la valeur exacte.

question "Q5 (Exécutez le lab) : Combien d'étapes l'agent Researcher a-t-il exécuté ?"

Comptez les lignes où agent_role == "researcher".

??? success "?" Révéler la réponse" 6 étapes

Le Researcher a exécuté 6 étapes ? le plus de tous les rôles d'agents. C'est logique car le Researcher traite plusieurs sous-questions du Planner, chaque sous-question pouvant nécessiter plusieurs étapes de recherche et d'extraction.

Résumé

Sujet | Ce que vous avez appris | Agents de recherche approfondie | Pipeline multi-agents pour la synthèse de connaissances avec suivi des citations | Architecture du pipeline | Planner ? Researcher ? Writer ? Reviewer avec des rôles spécialisés | Suivi des citations | Chaque affirmation est reliée à une source dans tout le pipeline | Distribution des tokens | Le Researcher utilise le plus de tokens ; le Writer synthétise ; le Reviewer valide | Schémas de qualité | La qualité varie selon les étapes ? les étapes de recherche ultérieures peuvent montrer un épuisement des sources | Optimisation | Mettre en cache les

sources, paralléliser la recherche, ajouter des portes de qualité |

Prochaines étapes

Lab 034 ? Multi-Agent avec Semantic Kernel (construire les agents eux-mêmes) Lab 067 ?

GraphRAG (enrichir la recherche avec la récupération par graphe de connaissances) Lab 033 ?

Observabilité des agents (surveiller les pipelines de recherche approfondie en production)

Lab 076 ? Microsoft Agent Framework (implémenter les pipelines avec les Graph Workflows de MAF)

Phase 3 · Simulation / Parcours Réel

Simulation Parcours Réel

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: Supprimez déploiements, points de terminaison, index, traces et données de test après validation.

Kit d'accompagnement

Le build du site génère des téléchargements PDF et ZIP stables. Le ZIP contient le guide du lab et tous les fichiers d'accompagnement déclarés.

Télécharger le PDF généré Télécharger le ZIP généré

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