

warning "Mise à jour du backend (octobre 2026)" L'inférence GitHub Models a été retirée le 30 juillet 2026. Les références ci-dessous sont conservées uniquement comme contexte historique pendant la migration de ce lab. Utilisez le Lab 013 ? Backends LLM gratuits pour configurer Foundry Local ou Ollama avant de continuer.

Lab 038 : Optimisation des coûts de l'IA

Niveau : L300 Parcours : ? Pro Code Durée : ~45 min ? Coût : Azure Free Tier ? La plupart des stratégies sont gratuites à mettre en ?uvre

Ce que vous apprendrez

Comprendre les facteurs de coût des API IA (tokens, choix du modèle, fréquence des requêtes) Appliquer 5 stratégies clés pour réduire les coûts : mise en cache, traitement par lots, routage de modèles, compression de prompts et streaming Implémenter un cache sémantique pour éviter les appels LLM en double Construire un routeur de modèles qui utilise des modèles économiques pour les requêtes simples et des modèles coûteux pour les requêtes complexes Surveiller et alerter sur les coûts avec Azure Monitor

Introduction

Un agent IA en production peut facilement coûter 50?500 \$/mois même avec une utilisation modeste s'il n'est pas optimisé. La bonne nouvelle : la plupart des stratégies de réduction des coûts améliorent également la latence et l'expérience utilisateur.

Les principaux leviers de coût :

Facteur | Impact | Optimisation | Choix du modèle | Différence de coût 10?50× | Router vers un modèle moins cher quand c'est possible | Nombre de tokens | Coût linéaire | Compresser les prompts, réduire l'historique | Hits de cache | Élimine entièrement le coût | Cache sémantique pour les requêtes répétées | Volume de requêtes | Coût linéaire | Grouper, temporiser, dédupliquer | Longueur de la sortie | Coût linéaire | Utiliser max_tokens, sortie structurée |

Partie 1 : Mesurer votre référence

Avant d'optimiser, mesurez ce que vous dépensez :

[diagram or code sample]

Partie 2 : Stratégie 1 ? Cache sémantique

Un cache sémantique stocke les réponses du LLM et les réutilise pour des requêtes sémantiquement similaires, même lorsque la formulation diffère :

[diagram or code sample]

Partie 3 : Stratégie 2 ? Routeur de modèles

Utilisez des modèles économiques pour les requêtes simples, des modèles coûteux pour les requêtes complexes :

[diagram or code sample]

Partie 4 : Stratégie 3 ? Compression de prompts

Réduisez l'historique de conversation en ne gardant que les échanges les plus récents + un résumé :

[diagram or code sample]

Partie 5 : Surveillance des coûts avec Azure Monitor

En production, suivez les coûts avec Azure Monitor :

[diagram or code sample]

Dans le portail Azure, créez une alerte :

[diagram or code sample]

? Quiz de connaissances

question "1. Qu'est-ce que le cache sémantique et en quoi diffère-t-il du cache par correspondance exacte ?" Le cache par correspondance exacte ne renvoie des résultats mis en

cache que lorsque l'entrée est identique octet par octet. Le cache sémantique utilise des embeddings pour trouver des réponses en cache pour des entrées sémantiquement similaires ? « Montrez-moi les tentes » et « Quelles tentes avez-vous ? » obtiennent la même réponse en cache même si la formulation diffère. Le cache sémantique est bien plus efficace pour l'IA conversationnelle.

question "2. Quand ne devriez-vous PAS utiliser un modèle économique dans un routeur de modèles ?" Utilisez un modèle premium pour : le raisonnement en plusieurs étapes (chaîne de pensée), les comparaisons entre des options complexes, les recommandations personnalisées qui nécessitent de pondérer de nombreux facteurs, les tâches où les erreurs sont coûteuses (médical, financier, juridique). Utilisez des modèles économiques pour : les recherches, les listes, le formatage, les questions-réponses simples où la tolérance aux erreurs est plus élevée.

question "3. Quels sont les risques d'une compression agressive de l'historique des prompts ?" L'agent peut perdre le contexte dont il a besoin : un utilisateur a mentionné une allergie 5 tours plus tôt et le résumé compressé l'a manqué. Un historique mal résumé peut causer des réponses contradictoires dans les longues conversations. Gardez toujours les N tours les plus récents textuellement ; ne résumez que les tours plus anciens. Testez avec des scénarios qui nécessitent un contexte à longue portée.

Résumé

Stratégie | Économies potentielles | Complexité | Cache sémantique | 40?80 % des appels API | Moyenne | Routage de modèles | Réduction de coût 10?40× | Faible | Compression de prompts | Réduction de 20?60 % des tokens | Faible | Groupement de requêtes | 10?30 % (compromis latence) | Moyenne | Limitation de la sortie (max_tokens) | 5?20 % | Très faible |

Prochaines étapes

Observer ce que vous avez optimisé : ? Lab 033 ? Observabilité des agents Évaluer la qualité après l'optimisation : ? Lab 035 ? Évaluation des agents Déploiement complet en production : ? Lab 028 ? Déployer MCP sur Azure

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: CHEATSHEET.md

Fichiers d'accompagnement: Lab 038 : Optimisation des coûts de l'IA

Commencez par le parcours local ou gratuit documenté. Consignez les hypothèses et les preuves de validation. Gardez les services hébergés facultatifs sauf exigence explicite du lab.

Supporting file: README.md

Lab 038 : Optimisation des coûts de l'IA

Utilisez ces fichiers avec le guide du lab et le kit généré.