

LLM pour la manipulation/raisonnement sur structures de graphe (graphe → question/réponse)

Amélioration du raisonnement sur graphes dans les grands modèles de langage via Graph Chain-of-Thought

- Methodology & Work Plan -

Research Approach:

Durant ce travail, nous adoptons une approche expérimentale comparative visant à évaluer l'impact du raisonnement structuré Graph-CoT sur les capacités des LLM à raisonner sur des graphes. En comparant des prompts standards à une stratégie Graph Chain-of-Thought sur des tâches de raisonnement issues du benchmark GraphEval36K, nous analysons de manière quantitative l'apport du raisonnement structuré pour la compréhension et la manipulation de graphes complexes.

Les articles scientifiques :

1. Large Language Models on Graphs : A Comprehensive Survey
2. GraphLLM: Boosting Graph Reasoning Ability of Large Language Models
3. Benchmarking Coding and Reasoning Capabilities of Large Language Models on Graph Datasets
4. Graph Chain-of-Thought: Augmenting Large Language Models by Reasoning on Graphs
5. Reasoning with Graphs: Structuring Implicit Knowledge to Enhance LLMs Reasoning
6. Graph of Thoughts: Solving Elaborate Problems with Large Language Models

Analyse comparative :

L'analyse comparative des travaux montre que chaque approche adresse une partie spécifique du problème du raisonnement sur graphes. Les architectures hybrides permettent une meilleure représentation explicite de la structure des graphes, mais leur capacité de raisonnement multi-étapes reste limitée. À l'inverse, les approches de type Graph Chain-of-Thought et Graph of Thoughts améliorent significativement le raisonnement structuré, mais ne s'appuient pas toujours sur des graphes de données réels ou explicites.

Par ailleurs, les benchmarks existants fournissent des outils d'évaluation robustes, mais ne proposent pas de solutions architecturales complètes. Ainsi, les approches actuelles restent fragmentées et ne traitent pas simultanément la représentation des graphes, le raisonnement structuré et l'évaluation.

- Comment permettre à un LLM de comprendre explicitement la structure d'un graphe et de raisonner dessus pour répondre à des questions ?
- Comment GraphLLM améliore Graph2Text ?
- Pourquoi Chain-of-Thought / Graph-of-Thought sont importants?

Plan de travail de l'équipe (Team Work Plan) :

Le travail a été réparti en plusieurs étapes successives, avec une répartition claire des responsabilités entre les membres de l'équipe.

Phase 1 — Étude bibliographique et cadrage du sujet

Objectif : Comprendre l'état de l'art et définir la problématique de recherche.

Tâches réalisées :

- Recherche et sélection d'articles scientifiques pertinents via Google Analyse approfondie des travaux sur :
 - les modèles de langage appliqués aux graphes,
 - GraphLLM,
 - Graph Chain-of-Thought,
 - Graph of Thoughts,
 - les benchmarks d'évaluation du raisonnement sur graphes.
- Identification des limites des approches existantes, notamment des méthodes Graph2Text et du raisonnement non structuré.

Phase 2 — Sélection et préparation des datasets

Objectif : Construire un jeu de données adapté au raisonnement des LLMs.

Tâches réalisées :

- Sélection du benchmark **GraphEval36K** pour l'évaluation des capacités de raisonnement sur graphes.
- Choix des tâches ciblées :
 - détection de cycles (lc210),
 - recherche de chemins (lc797),
 - reachability.
- Conversion automatique des fichiers JSONL vers des datasets exploitables par les LLMs.
- Transformation des problèmes de type *coding* en problèmes de *raisonnement en langage naturel* (questions, réponses et instructions).

Phase 3 — Conception de la méthodologie proposée (Graph-CoT)

Objectif : Définir une approche de raisonnement structuré sur graphes.

Tâches réalisées :

- Conception d'une approche hybride inspirée de GraphLLM et Graph Chain-of-Thought.

- Définition d'un pipeline en trois étapes :
 1. Encodage textuel de la structure du graphe.
 2. Intégration de la représentation graphique dans le prompt du LLM.
 3. Raisonnement multi-étapes explicite à l'aide de Graph-CoT.
- Conception de prompts structurés incluant des étapes de raisonnement explicites.

Phase 4 — Implémentation technique

Objectif : Mettre en œuvre la méthodologie proposée.

Tâches réalisées :

- Développement de scripts Python pour :
 - la conversion des datasets,
 - l'encodage textuel des graphes,
 - la génération des prompts standards et Graph-CoT.
- Automatisation de la génération des fichiers de prompts.
- Implémentation des scripts d'évaluation des prédictions.

Phase 5 — Expérimentation et évaluation

Objectif : Comparer les performances du prompt standard et de Graph-CoT.

Tâches réalisées :

- Exécution des expériences sur les datasets sélectionnés.
- Collecte des prédictions du modèle pour chaque type de prompt.
- Calcul des métriques d'évaluation :
 - accuracy,
 - success rate,
 - raisonnements corrects et incorrects.
- Génération de tableaux récapitulatifs et de graphiques comparatifs..

Phase 6 — Analyse qualitative et interprétation des résultats

Objectif : Comprendre les raisons des performances observées.

Tâches réalisées :

- Sélection automatique et manuelle d'exemples représentatifs.
- Analyse qualitative de quatre catégories :
 - correct_graph_cot,
 - incorrect_graph_cot,
 - correct_standard,
 - incorrect_standard.
- Identification des schémas de raisonnement réussis et des causes d'erreur.

Phase 7 — Rédition et finalisation du rapport

Objectif : Produire un article scientifique complet.

Tâches réalisées :

- Rédaction de l'article selon une structure académique standard.
- Intégration des résultats expérimentaux et analyses.
- Discussion des limites et des perspectives futures.
- Relecture et harmonisation du document final.

Dataset description :

Dans ce projet, nous utilisons le benchmark **GraphEval36K**, un dataset spécifiquement conçu pour évaluer les capacités de raisonnement des grands modèles de langage sur des graphes. Ce dataset comprend plus de 36 000 instances couvrant diverses tâches de raisonnement graphique, telles que le calcul de chemins, la connectivité et le raisonnement logique. Son choix est motivé par sa capacité à tester le raisonnement multi-étapes et la compréhension explicite de la structure des graphes, en adéquation avec les objectifs de ce travail.

- **Choix recommandé pour notre projet :**

Nous nous concentrons sur des graphes orientés associés à des tâches nécessitant un raisonnement multi-étapes, telles que le calcul de chemins et la détection de cycles.

- **Directed graphs**
- **Problèmes de raisonnement structurel**
 - topological sort (lc210)
 - path finding (lc797)
 - reachability

Nous avons réalisé une **transformation méthodologique du dataset initial**, initialement conçu pour des tâches de coding contenant des **graphes bruts** (des listes de flèches entre des nœuds) qui venaient de problèmes de programmation, en un **dataset de raisonnement destiné à l'évaluation des Large Language Models (LLM)**.

Cette transformation consiste à convertir des instances algorithmiques basées sur du code et des structures brutes en **questions formulées en langage naturel**, accompagnées de **réponses exploitables** sous forme textuelle. Chaque graphe est ainsi représenté de manière structurée, puis associé à une question de raisonnement explicite (par exemple, la détection de cycles ou l'atteignabilité), permettant d'évaluer les capacités de compréhension et d'inférence des modèles. Le dataset résultant, organisé sous la structure GraphEval36K_processed (incluant notamment les sous-ensembles *lc210*, *lc797* et *reachability*), constitue une **contribution méthodologique originale**, facilitant l'analyse du raisonnement des LLM sur des problèmes de graphes abstraits.