

Extraction de connaissances de résolution de problèmes à partir de LLM dotés de capacités de raisonnement

Maxime Haurel^{1, 2}

¹ Université de Lorraine, CNRS, LORIA, F-54000 Nancy, France

² Datanello, Nancy, France

maxime.haurel@loria.fr

Résumé

Les grands modèles de langue (LLM) ont permis de faciliter la création automatique de bases de connaissances (KB). Nous nous intéressons aux LLM dotés de capacités de raisonnement (LRM) pour construire des KB automatiquement sur la base des traces de raisonnement qu'ils nous fournissent. Dans ce travail, nous évaluons la qualité syntaxique de la connaissance générée par des LRM. C'est la première étape vers l'évaluation rigoureuse de connaissances générées puis de la construction automatique de KB fiables et à large échelle.

Mots-clés

acquisition automatique de connaissances, grands modèles de langue, modèles de raisonnement, prolog.

Abstract

Large Language Models (LLM) facilitate the automatic construction of knowledge bases (KB). We take interest on automatically building KB through LLM with reasoning abilities (LRM) by leveraging the reasoning traces these models output. In this work, we evaluate the syntactic quality of knowledge generated by LRMs. This is the first step towards the rigorous evaluation of generated knowledge and then automatic construction of reliable and large-scale KBs.

Keywords

automatic knowledge acquisition, large language models, reasoning models, prolog.

1 Introduction

L'acquisition automatique de connaissances (AKA) est une tâche chronophage. En effet, les experts du domaine et les ingénieurs en connaissance collaborent pour extraire les connaissances du domaine et les représenter dans une base de connaissances (KB) de manière structurée [23]. La littérature a récemment montré que la construction de KB peut être soutenue par les grands modèles de langue (LLM), à la fois de manière semi-automatique [14] et automatique [5, 18]. Dans les études utilisant les LLM pour l'AKA, la majorité des approches utilisent des LLM sans capacités de raisonnement [14, 5]. En effet, les LLM dotés

de capacités de raisonnement (ou modèles de raisonnement, ou LRM pour Large Reasoning Model) ne sont apparus que récemment et très peu de travaux les ont jusqu'à présent exploités pour la construction de KB [18].

Dans ce travail, nous nous intéressons spécifiquement à l'utilisation des capacités de raisonnement des LLM dans l'AKA, c'est-à-dire à l'extraction de connaissances de résolution de problèmes à partir de ces modèles. Par capacités de raisonnement, nous désignons la production de traces de raisonnement en langage naturel encapsulées entre des balises `<think></think>`. Les LLM dotés de ces capacités (LRM) sont ré-entraînés avec l'apprentissage par renforcement (RL) [21]. Ces derniers ont reçu une attention particulière depuis la publication de DeepSeek-R1 en janvier 2025 [10] car ils affichent des résultats remarquables sur de nombreux benchmarks. L'entraînement par RL du LLM a été conçu pour récompenser les sorties dans lesquelles la section de raisonnement inclut un texte représentant une "chaîne de pensée" (chain-of-thought) menant à la réponse. Un exemple est présenté dans le tableau 1.

Ce travail s'inscrit dans le cadre d'une thèse de doctorat. Nous émettons l'hypothèse que ce raisonnement textuel et non formel peut être analysé pour extraire des connaissances de domaine de résolution de problèmes sur lesquelles le LRM s'appuie pour construire sa réponse. Sans surprise, compte tenu de l'apparition récente des LRM, ils ont été jusqu'à présent rarement utilisés. Ce travail vise à explorer la viabilité de la création automatique de KB fiables en tirant parti de ces LRM. Le but final est de rendre un LRM capable de gérer automatiquement le cycle de vie d'une KB (affiner, réutiliser, raisonner). Atteindre cet objectif contribuera au développement de l'AKA (donc faciliter la construction de KB plus grandes et plus fiables) et des LLM (donc rendre les LLM plus interprétables). Dans les sections suivantes, nous décrivons plusieurs approches pour vérifier nos hypothèses. À la fin de ce document, nous énumérons les prochaines étapes de ce travail.

Ce travail est une traduction d'un article déjà publié dans le cadre du *doctoral consortium* de la conférence internationale ISWC 2025 [11].

2 Motivation

L'AKA est une question cruciale dans la communauté de l'intelligence artificielle se concentrant sur les systèmes à base de connaissance. En effet, la transformation de connaissances exprimées dans des sources non structurées en informations structurées est un processus qui engendre à la fois des coûts humains et des coûts en temps. Ce challenge limite notre capacité à construire des KB fiables à grande échelle [12]. Nous faisons l'hypothèse que l'utilisation de LRM pour acquérir automatiquement des connaissances et gérer le cycle de vie d'une KB limitera ce problème.

Notre approche vise à construire automatiquement une KB, en utilisant les traces de raisonnement produites par un LRM comme expression de la connaissance. Nous souhaitons obtenir une version formalisée des traces de raisonnement dans l'objectif de construire une KB avec un formalisme qui a fait ses preuves pour cette tâche dans la communauté de la représentation des connaissances et du raisonnement. De plus, les LRM ont récemment émergé, en particulier avec DeepSeek-R1 en janvier 2025 [10]. Bien que ces modèles soient capables de montrer une forme d'explication par rapport aux LLM sans capacités de raisonnement, les LLM en général (c.-à-d. avec ou sans capacités de raisonnement) fonctionnent toujours comme des boîtes noires [31, 22, 27]. Notre approche nous aidera à mieux comprendre ces modèles, contribuant ainsi à leur interprétabilité. En effet, notre approche pourrait expliquer une prédiction par un raisonnement logique, ou même valider/invalidier les résultats obtenus en se basant sur la cohérence du raisonnement logique.

Notre recherche est motivée par un problème rencontré dans le domaine des ressources humaines (RH). Le problème est d'identifier des indicateurs de compétences comportementales (soft skills) qui sont exprimés dans un texte résumant les décisions prises par un individu dans le rôle d'un manager face à une situation difficile (par exemple, faire face à un client en colère en raison d'un problème d'expédition). Cette tâche est hautement intensive en connaissances car elle requiert de solides connaissances en psychologie de la part d'un expert. De plus, aucune donnée étiquetée n'est disponible pour cette tâche de prédiction d'indicateurs de soft skills à partir d'un texte. En outre, nous sommes confrontés au problème de la disponibilité d'experts du domaine qualifiés sur le sujet. C'est pourquoi nous avons besoin d'une méthode qui soit à la fois automatique et explicable. Dans cet article, nous nous limitons à la description du cas d'usage RH au niveau conceptuel ; une illustration détaillée sera présentée dans une contribution dédiée à l'évaluation sur ce domaine.

Par ailleurs, le domaine des RH est un domaine à forts enjeux comme le stipule la loi européenne sur l'IA (EU AI Act) [1]. Plus précisément, le document stipule que : "Les systèmes d'IA destinés à être utilisés pour prendre des décisions affectant les conditions des relations liées au travail, la promotion ou la résiliation des relations contractuelles liées au travail, pour attribuer des tâches sur la base du comporte-

ment individuel ou des traits ou caractéristiques personnels ou pour surveiller et évaluer la performance et le comportement des personnes dans de telles relations" sont classés à haut risque. Nous avons donc besoin d'une méthode qui fournisse des réponses face à un problème complexe (par exemple, prédire des indicateurs de soft skills à partir d'un texte) en utilisant des connaissances expertes formellement définies dans une KB, tout en fournissant des explications concernant le processus de raisonnement utilisé par le modèle pour atteindre la réponse.

Ainsi, construire automatiquement une KB axée sur les connaissances de résolution de problèmes avec un LRM pour ensuite utiliser cette KB afin de raisonner sur des problèmes complexes est prometteur. Bien que notre objectif soit d'utiliser des LRM comme des experts du domaine pour gérer le cycle de vie d'une KB, nous sommes conscients des considérations éthiques [26] et nous traiterons ces considérations lors du développement du framework. Bien que d'autres approches existent, notre approche vise à réduire les obstacles liés au manque d'experts du domaine et au manque de données annotées. Le domaine cible de la thèse est celui des RH, mais l'étude présentée ici s'appuie sur des exemples médicaux issus d'un jeu de données public, utilisés comme banc d'essai exploratoire pour des raisons de disponibilité et de reproductibilité. Le corpus RH, en cours de constitution en français, n'est pas encore exploitable dans le cadre du papier international à l'origine de cette traduction. Le domaine médical reste néanmoins pertinent pour cette étape car il exige un raisonnement solide pour fournir des réponses fiables [6].

3 Travaux antérieurs

Notre travail s'inscrit dans l'AKA, mais avec un ensemble de contraintes. Comme mentionné précédemment, notre objectif est de modéliser des connaissances utiles à la résolution d'un problème spécifique (naturellement intensif en connaissances), comme la prédiction d'indicateurs de soft skills. De plus, les connaissances dont nous avons besoin ne sont stockées dans aucun document. Nous n'avons pas accès à de grandes quantités de cas passés de tels problèmes résolus, et la solution n'est pas automatiquement vérifiable. Notre intérêt pour les LLM vient donc de l'hypothèse que, par l'effet de leur entraînement, ils ont acquis des connaissances spécifiques au domaine, ce qui signifie qu'ils pourraient, au moins en partie, jouer le rôle d'un expert du domaine. L'explicabilité est une considération cruciale, étant donné le niveau de risque élevé énoncé par la loi européenne sur l'IA dans le domaine des RH [1]. S'appuyer sur des connaissances explicitement formalisées dans une KB est donc privilégié ici. Par conséquent, nous passons ici en revue les travaux existants dans l'AKA, en particulier avec les LLM. Nous explorons également la littérature pour passer en revue les formalismes utilisés par la communauté de la représentation des connaissances et du raisonnement.

Acquisition automatique de connaissances sans LLM
Plusieurs travaux se sont concentrés sur la construction automatique de KB sans avoir recours aux LLM. Nous avons

<think> Okay, so I'm trying to figure out the possible heart condition based on the symptoms described. The user mentioned sudden weakness in the left arm and leg, which makes me think of a stroke or possibly a TIA (transient ischemic attack). But then there's also recent long-distance travel and a swollen, tender right lower leg. First, the left-sided weakness points towards neurological issues, likely related to blood flow. A stroke or TIA is a common cause for sudden weakness in one side of the body. However, the user also provided other symptoms that I need to consider together. The long-distance travel makes me think about prolonged sitting or physical activity during the trip. This could contribute to deep vein thrombosis (DVT), especially if there's a history of blood clots or risk factors like obesity, age over 40, or being sedentary for long periods. Now, the right lower leg symptoms—swollen and tender—are classic signs of DVT. DVT occurs when blood clots form in the legs and can travel to the lungs (causing a pulmonary embolism) or other organs. If someone experiences sudden weakness on one side along with DVT, it might be related to an embolism blocking a blood vessel in the brain, leading to a stroke. So, putting this together, the person might have experienced a pulmonary embolism from DVT, which then caused a stroke-like syndrome. This would explain both sets of symptoms : the left-sided weakness from the stroke and the right leg issues from DVT. **</think>**

<answer> The most likely cardiac abnormality is a pulmonary embolism secondary to deep vein thrombosis (DVT). The recent long-distance travel increases the risk of DVT, which can lead to a pulmonary embolism. This condition can cause sudden neurological symptoms like left-sided weakness due to an embolism blocking a cerebral artery, mimicking a stroke or TIA.**</answer>**

TABLE 1 – Un exemple de la sortie brute obtenue à partir de DeepSeek-R1 dans sa version 8B. Les traces de raisonnement sont montrées entre les balises `<think>` (en bleu). Le problème donné au LRM est : *Given the symptoms of sudden weakness in the left arm and leg, recent long-distance travel, and the presence of swollen and tender right lower leg, what specific cardiac abnormality is most likely to be found upon further evaluation that could explain these findings ?*

identifié deux approches dans la littérature : l'approche par traitement de document (langage naturel) et l'approche par apprentissage (statistique). Par exemple, [24] propose de construire automatiquement une KB par l'analyse syntaxique de documents, ce qui nécessite donc l'accès à un corpus de documents pertinents. Concernant l'approche par apprentissage, [15] compare un système d'apprentissage inductif à un système d'apprentissage bayésien naïf. Le premier génère des connaissances sous forme d'arbre de décision, tandis que le second génère une table de probabilités conditionnelles et a priori. Les deux systèmes (inductif et bayésien naïf) sont basés sur une collection de problèmes concrets, résolus par des experts dans le passé. Ici aussi, l'applicabilité de la méthode n'est possible que dans le cas où une base de problèmes précédemment résolus est disponible, ce qui n'est pas le scénario sur lequel nous nous concentrons.

Acquisition automatique de connaissances avec LLM

Les LLM sont entraînés sur un vaste corpus de données, ce qui les rend capables de tirer parti d'une grande quantité de connaissances inter-domaines. Ils surpassent même les humains dans certains domaines tels que la production de code [13], la génération d'histoires [32] et la réponse à des questions liées à la génétique [9]. Les travaux connexes suivants s'appuient sur l'idée que les LLM peuvent être exploités comme source de connaissances, à l'instar des experts du domaine. Récemment, [14] a proposé une approche semi-automatique (humain dans la boucle) pour construire une ontologie puis un graphe de connaissances (KG) à partir de questions de compétence (CQ). D'autres travaux, comme [5], visent une construction automatique de KG

avec un pipeline en trois modules : (i) un Générateur qui propose des triplets candidats, (ii) un Vérificateur qui filtre les triplets erronés, et (iii) un Élagueur (Pruner) qui décide de poursuivre ou non l'expansion du KG. Cette approche améliore la qualité du KG par rapport aux méthodes précédentes et confirme l'intérêt des LLM pour l'AKA. Cependant, ces travaux ciblent principalement des connaissances ontologiques ou relationnelles générales, alors que notre objectif porte sur l'extraction de connaissances de résolution de problèmes.

Acquisition automatique de connaissances avec LRM

Des approches récentes utilisent le RL (Apprentissage par Renforcement) pour influencer le comportement d'un LLM. L'algorithme Group Relative Policy Optimization (GRPO) [21] offre un moyen efficace d'entraîner le LLM à maximiser une récompense avec l'apprentissage par renforcement. Cet algorithme est la base derrière le modèle DeepSeek-R1 [10] qui a obtenu d'excellents scores sur plusieurs benchmarks et a ouvert la voie aux LRM. Dans ce cadre, la fonction de récompense utilisée pendant l'entraînement de DeepSeek-R1 favorise explicitement la production d'un raisonnement structuré dans les balises `<think></think>`. Publié en avril 2025, le LRM Cogito [8] a attiré l'attention grâce à d'excellentes performances sur les benchmarks. Étant donné que ces modèles dotés de capacités de raisonnement sont très récents, peu de travaux dans la littérature les ont utilisés pour l'AKA. [18, 17] montrent l'utilisation de LLM dotés de capacités de raisonnement pour générer des ontologies. Ils représentent les connaissances dans le langage OWL après avoir fourni des CQ en entrée au LRM. Cependant, ces travaux ne se

servent pas directement des traces de raisonnement générées.

Représentation des connaissances et raisonnement Bien qu’il existe de nombreux formalismes de représentation des connaissances, Prolog est un bon candidat pour modéliser des représentations symboliques de problèmes de raisonnement [19, 29, 7, 16, 30, 28, 3]. Par exemple, [28] améliore le raisonnement logique des LLM grâce à un framework comprenant notamment un module Traducteur (Translator) qui vise à analyser des prémisses données et un énoncé de question avec un LLM vers un format symbolique adoptant la grammaire de Prolog. Une autre approche [3] utilise Vadalog [4], un langage dérivé de Datalog [2]. Dans l’étude, le langage Vadalog est utilisé pour décrire le graphe de connaissances et agir comme une couche intermédiaire entre la question en langage naturel formulée par l’utilisateur et la connaissance réelle représentée sous forme de graphe. Par rapport à OWL, davantage orienté modélisation ontologique et interopérabilité sémantique, Prolog et Vadalog sont ici privilégiés pour leur caractère directement exécutable et leur adéquation à une représentation de raisonnement sous forme de règles.

Dans la littérature scientifique, Prolog a été largement utilisé. Pour l’application que nous abordons tout au long de cette thèse (c.-à-d. les RH), nous choisissons Prolog comme premier candidat pour représenter les connaissances de résolution de problèmes. Ce choix est motivé par sa syntaxe textuelle, compatible avec la génération par LRM, et par son exécution directe sous forme de règles, utile pour vérifier la cohérence d’un raisonnement. Bien que Prolog présente de nombreux avantages, une limitation est qu’il n’est pas possible de représenter des probabilités. Pour surmonter cette limitation et si nous avons besoin d’une représentation probabiliste des connaissances, nous envisagerons d’utiliser ProbLog [20], une extension de Prolog. En complément, nous prévoyons aussi d’expérimenter d’autres formalismes à base de règles, avec le langage Datalog, dans la suite de cette thèse.

4 Questions de recherche et hypothèses

Cette thèse vise à contribuer à l’AKA en représentant les connaissances dans des KB au moyen de LRM. Plus précisément, nous visons à répondre à la question suivante : *Comment tirer parti des connaissances exprimées dans les traces de raisonnement d’un LRM pour créer et gérer automatiquement une KB spécialisée dans un domaine ?*. En développant la question de recherche, nous postulons les hypothèses décrites ci-dessous.

H1 : Étant donné un formalisme de raisonnement et de représentation des connaissances défini, les LRM sont capables, éventuellement avec un entraînement supplémentaire, de générer une trace de raisonnement dans ce formalisme. Les LLM ont été utilisés pour générer des sorties Prolog [30] mais aucun travail sur cette tâche utilisant des LRM n’a été publié, probablement en raison de leur nouveauté. Dans un premier temps, nous visons à évaluer

si des LRM sont capables de générer un code Prolog syntaxiquement valide dans la partie dédiée au raisonnement. L’expérience associée est détaillée dans la Section 5.

H2 : Le renforcement de la capacité du LRM à générer des connaissances formalisées dans ses traces de raisonnement consolide sa fiabilité à générer des réponses valides, et des connaissances valides. Cette hypothèse nécessite un intérêt à la fois pour les traces de raisonnement pour un problème donné et, plus généralement, pour la KB qui est produite en résolvant de multiples problèmes, itérativement. En ce qui concerne les traces de raisonnement, nous commençons par introduire une mesure s qui nous permet de mesurer à quel point un code est syntaxiquement valide dans un langage donné. Nous l’avons initialement appliquée au langage Prolog avec l’idée que d’autres travaux devront être effectués pour évaluer également la validité sémantique du Prolog généré. Plus précisément, nous prévoyons d’ajouter des mesures de cohérence des règles, d’exécutabilité des programmes et d’alignement entre la réponse finale et le raisonnement formalisé. L’évaluation de la qualité d’une KB est beaucoup plus complexe et fera l’objet de travaux ultérieurs. Ce type d’évaluation existe déjà et implique plusieurs étapes [23] :

- Débogage : détection d’incohérences (conflit, redondance, subsomption)
- Vérification de l’exhaustivité : détection d’incomplétudes (règles manquantes)

Dans la section suivante (Section 5), nous commençons par travailler sur la première hypothèse. Nous avons conçu une expérience pour évaluer si chaque LRM testé est capable de générer la trace de raisonnement dans la syntaxe du formalisme Prolog. Pour y parvenir, nous utilisons la mesure s et, dans la suite de cette thèse, nous introduirons d’autres mesures pour évaluer la qualité de la KB créée (cohérence logique, couverture des règles et alignement raisonnement-réponse).

5 Résultats préliminaires

Cette section présente un premier protocole expérimental, volontairement simple, afin d’évaluer la faisabilité de la génération de traces en Prolog avec des LRM.

Parmi les LRM disponibles et open-weights (c.-à-d., dont les poids appris par les modèles sont disponibles en accès libre), nous considérons DeepSeek-R1 et Cogito, tous deux dans leurs versions distillées en raison de leurs coûts de calcul. DeepSeek-R1 est un modèle de raisonnement entraîné par RL pour produire des traces `<think></think>` [10], tandis que Cogito est une famille récente de LRM open-weights orientée raisonnement [8]. Concrètement, nous testons les versions distillées 1.5B¹ et 8B² de DeepSeek-R1 et les versions dis-

1. <https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-1.5B>

2. <https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Llama-8B>

tillées 3B³ et 8B⁴ de Cogito. Nous n’avons pas pris en compte QwQ [25] ou tout autre modèle plus grand car ils requièrent plus de ressources que nous n’en disposons à ce stade. Comme nous visons à affiner (fine-tune) le modèle lors de travaux futurs, les LRM propriétaires ne sont pas pris en compte ici.

Prompting Un problème (c.-à-d. une question dont la réponse nécessite un raisonnement) est soumis au LRM, en lui demandant d’exprimer son raisonnement en Prolog. Dans cette expérience préliminaire, ce problème est tiré du jeu de données *medical-ol-reasoning-SFT*⁵ [6]. Ce jeu de données contient, pour chaque entrée, les champs `Question`, `Complex_CoT` et `Response`; le sous-ensemble anglais contient environ 19.7k exemples. Le problème est inséré dans un prompt inspiré de celui utilisé pour entraîner DeepSeek-R1 [10], comme indiqué dans le tableau 3. Nous évaluons uniquement la validité syntaxique du Prolog généré dans la partie raisonnement en utilisant la mesure s définie ci-dessous. Pour cette étude exploratoire, nous répétons le processus 10 fois pour chaque LRM sur un même problème afin d’obtenir un aperçu moyen de la qualité du Prolog généré dans le raisonnement.

Le parseur est utilisé pour analyser chaque trace, détecter les erreurs de syntaxe Prolog et produire un décompte d’erreurs. Étant donné ce parseur⁶, s est défini comme :

$$s = \begin{cases} 1 - (n/m) & \text{if } n > 0 \\ 1 & \text{else} \end{cases}$$

où n est le nombre d’erreurs détectées par le parseur et m est le nombre de lignes des traces de raisonnement. s est égal à 0 si le LRM n’a rien renvoyé dans le raisonnement.

Le tableau 2 présente les résultats obtenus. Les faibles valeurs de s sont cohérentes avec le fait que les LRM testés ne sont pas spécifiquement entraînés pour produire du Prolog. Pour DeepSeek-R1 :8B, les traces de raisonnement sont systématiquement vides sur nos essais (score nul). Plus généralement, malgré plusieurs tentatives et variantes de prompt, nous n’avons pas réussi à obtenir des traces Prolog exploitables avec les versions testées de DeepSeek-R1. Les modèles Cogito obtiennent de meilleurs résultats sur la mesure s , avec une moyenne plus élevée pour Cogito :3B que pour Cogito :8B. Une hypothèse d’explication est que la version 3B suit plus strictement la contrainte de format imposée par le prompt, tandis que la version 8B produit plus souvent du texte libre dans la trace; cette hypothèse devra être vérifiée sur un échantillon plus large. Même si la mesure s est utile pour la syntaxe, elle n’évalue pas dans quelle mesure le code Prolog est porteur de sens. Enfin,

3. <https://huggingface.co/deepcogito/cogito-v1-preview-llama-3B>

4. <https://huggingface.co/deepcogito/cogito-v1-preview-llama-8B>

5. <https://huggingface.co/datasets/FreedomIntelligence/medical-ol-reasoning-SFT>

6. Ici nous utilisons <https://github.com/akvnn/prolog-parser> en tant que base du parseur. Le parseur tente de poursuivre l’analyse et de remonter plusieurs erreurs plutôt que d’échouer à la première.

avec seulement 10 essais par modèle, ces résultats restent exploratoires : ils orientent les étapes suivantes, sans permettre de conclure statistiquement à eux seuls.

| Model | Mean | Std | Min | Max |
|-------------------|--------|--------|--------|--------|
| cogito :3b | 0.3621 | 0.3619 | 0.0500 | 0.8333 |
| cogito :8b | 0.2406 | 0.3115 | 0.0500 | 0.7727 |
| deepseek-r1 :1.5b | 0.0500 | 0.0000 | 0.0500 | 0.0500 |
| deepseek-r1 :8b | 0.0000 | 0.0000 | 0.0000 | 0.0000 |

TABLE 2 – Statistiques descriptives de la mesure s par modèle.

A conversation between User and Assistant. The user asks a question, and the Assistant solves it. The assistant first thinks about the reasoning process in the mind and then provides the user with the answer. The reasoning process and answer are enclosed within `<think>` `</think>`, i.e., `<think>` reasoning process here `</think>` tags. **All reasoning inside `<think>` must be done in Prolog style : define facts, rules, and queries like a Prolog program. Use step-by-step logic and inference. Do not explain in natural language — only use Prolog.** User : **prompt**. Assistant :

TABLE 3 – Prompt pour induire des traces de raisonnement en Prolog. En bleu sont inscrites les modifications effectuées au prompt initial. En rouge est la place du problème que nous injectons dynamiquement sur lequel le LRM doit raisonner.

Affinage du LLM Les résultats obtenus lors de la phase de prompting nous encouragent à poursuivre avec l’affinage du LRM Cogito. Le code Prolog généré dans le raisonnement de Cogito par rapport à DeepSeek-R1 nous fait penser qu’il y a une marge d’amélioration. Nous prévoyons d’utiliser l’algorithme GRPO [21] (donc l’algorithme utilisé pour entraîner DeepSeek-R1 à produire des traces de raisonnement) pour affiner Cogito, en utilisant le score s comme récompense à maximiser. Cet affinage est la prochaine étape de la thèse.

6 Évaluation

Pour répondre à notre question de recherche, nous nous concentrerons d’abord sur le même jeu de données : *medical-ol-reasoning-SFT*, utilisé ici comme cadre expérimental exploratoire. Le transfert vers le domaine RH (domaine cible) sera étudié dès qu’un corpus RH en français sera disponible. Cependant, pour aller au-delà de l’évaluation syntaxique que nous avons réalisée dans la Section 5, une première étape consistera à construire une première KB incrémentalement, en faisant raisonner un LLM sur plus d’un problème. En nous inspirant de la littérature [23], nous introduirons ensuite des mesures de validité structurale (détection de variables inutilisées, détection de termes inutilisés, respect d’un vocabulaire prédéfini), d’alignement

entre réponse finale et raisonnement formalisé et de performance. Pour le domaine RH, une validation par des experts du domaine est prévue afin d'évaluer la pertinence des connaissances acquises et des sorties produites.

7 Travaux futurs

Comme mentionné dans la Section 5, nous prévoyons d'affiner Cogito avec l'algorithme GRPO, en utilisant la mesure s comme récompense pour entraîner le LRM à générer, de manière plus robuste, un code Prolog syntaxiquement valide dans ses traces de raisonnement. Un objectif de nos expériences initiales était de vérifier si les modèles existants avaient un certain niveau de capacité sur lequel un affinage pourrait s'appuyer pour accomplir cette tâche. Comme nous l'avons montré, cela semble être vrai pour Cogito, mais pas pour DeepSeek-R1. La prochaine expérience à réaliser aura donc pour objectif de tester à quel point la capacité de Cogito à générer des traces de raisonnement en Prolog peut être améliorée par l'affinage. Cela constituera la première étape vers une gestion automatique du cycle de vie d'une KB. Nous prévoyons de nous appuyer sur cette capacité pour extraire des connaissances du domaine, en traitant la validité sémantique et la cohérence sur de multiples problèmes en plus de la validité syntaxique.

De plus, comme mentionné précédemment dans cet article, nous avons choisi d'utiliser Prolog comme premier formalisme pour tester les LRM. Une prochaine contribution inclura également l'étude d'autres formalismes, tels que Datalog pour comparer différents compromis d'expressivité et de vérifiabilité. Nous anticipons également le besoin d'un framework plus large et plus robuste, comme illustré dans des études connexes [5, 28, 3]. La construction d'un framework plus large nous permettra d'évaluer toutes les étapes indépendamment afin de parvenir à la construction automatique d'une KB robuste. Ce framework, appliqué au domaine des RH, nécessitera également une formalisation précise, garantissant une grande robustesse.

Comme mentionné ci-dessus, nous devons aller au-delà de l'évaluation de la validité syntaxique de la trace de raisonnement générée, et également aborder la qualité de la KB créée en combinant les connaissances utilisées pour de multiples problèmes liés. Par conséquent, une étude bibliographique sur l'évaluation de la qualité des KB est l'une de nos priorités. Dans la vision à long terme de cette recherche, nous utiliserons la capacité de génération de formalisme du LLM dans le raisonnement pour valider ou invalider sa prédiction finale. Cela agira comme une sécurité pour signaler les réponses invalides et/ou illogiques. Nous anticipons donc que notre méthode contribuera également à l'interprétabilité, en fournissant un lien vérifiable entre la trace de raisonnement et la réponse générée. En effet, alors que le langage naturel dans les traces de raisonnement, tel qu'il est classiquement exprimé dans les LRM, offre un moyen de mieux comprendre si la réponse pourrait ou non être correcte, notre approche pourrait fournir un moyen d'automatiser cette vérification.

Remerciements

L'auteur remercie le Pr. Armelle Brun, professeur d'informatique à l'Université de Lorraine, le Pr. Mathieu d'Aquin, professeur d'informatique à l'Université de Lorraine, en tant que directeurs de cette thèse et le Dr. Yacine Abboud, directeur de Datanello et co-directeur de cette thèse. Cette recherche est financée par une bourse de l'ANRT via le dispositif CIFRE avec Datanello comme collaborateur industriel.

Déclaration sur l'IA générative

Lors de la préparation de ce travail, l'auteur déclare avoir utilisé GPT-4 pour générer des formules LaTeX et Writefull pour revoir la syntaxe du document. Gemini-3-Pro a aussi été utilisé en tant qu'assistant à la traduction de l'article original vers cet article. Après avoir utilisé ces outils, l'auteur a revu et édité le contenu selon les besoins et assume l'entière responsabilité du contenu de la publication.

Références

- [1] Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence and amending Regulations (EC) No 300/2008, (EU) No 167/2013, (EU) No 168/2013, (EU) 2018/858, (EU) 2018/1139 and (EU) 2019/2144 and Directives 2014/90/EU, (EU) 2016/797 and (EU) 2020/1828 (Artificial Intelligence Act)Text with EEA relevance., June 2024.
- [2] Serge Abiteboul, Richard Hull, and Victor Vianu. *Foundations of Databases*. Addison-Wesley, 1995.
- [3] Teodoro Baldazzi, Luigi Bellomarini, Stefano Ceri, Andrea Colombo, Andrea Gentili, and Emanuel Sallinger. "Please, Vadalog, tell me why" : Interactive Explanation of Datalog-based Reasoning. pages 834–837. OpenProceedings, March 2024. Accepted : 2025-02-04T16 :48 :04Z.
- [4] Luigi Bellomarini, Davide Benedetto, Georg Gottlob, and Emanuel Sallinger. Vadalog : A modern architecture for automated reasoning with large knowledge graphs. *Information Systems*, 105 :101528, March 2022.
- [5] Hanzhu Chen, Xu Shen, Qitan Lv, Jie Wang, Xiaoqi Ni, and Jieping Ye. SAC-KG : Exploiting Large Language Models as Skilled Automatic Constructors for Domain Knowledge Graphs, September 2024. arXiv :2410.02811 [cs].
- [6] Junying Chen, Zhenyang Cai, Ke Ji, Xidong Wang, Wanlong Liu, Rongsheng Wang, Jianye Hou, and Benyou Wang. HuatuoGPT-o1, Towards Medical Complex Reasoning with LLMs, December 2024. arXiv :2412.18925 [cs].
- [7] William F. Clocksin and Christopher S. Mellish. *Programming in Prolog*. Springer Science & Business Media, July 2003. Google-Books-ID : VjHk2Cjrti8C.

- [8] DeepCogito. Cogito v1 Preview Introducing IDA as a path to general superintelligence, April 2025.
- [9] Dat Duong and Benjamin D. Solomon. Analysis of large-language model versus human performance for genetics questions. *European Journal of Human Genetics*, 32(4) :466–468, April 2024.
- [10] DeepSeek-AI et al. DeepSeek-R1 : Incentivizing Reasoning Capability in LLMs via Reinforcement Learning, January 2025. arXiv :2501.12948 [cs].
- [11] Maxime Haurel. Extracting problem-solving knowledge from LLMs with reasoning abilities. *ISWC 2025 Companion Volume : Joint Proceedings of Industry, Doctoral Consortium, Posters and Demos of the 24th International Semantic Web Conference (ISWC-C 2025)*, 2025.
- [12] Qiyuan He, Jianfei Yu, and Wenya Wang. Large Language Model-Enhanced Symbolic Reasoning for Knowledge Base Completion, January 2025. arXiv :2501.01246 [cs].
- [13] Wenpin Hou and Zhicheng Ji. Comparing large language models and human programmers for generating programming code. *Advanced Science*, 12(8), December 2024.
- [14] Vamsi Krishna Kommineni, Birgitta König-Ries, and Sheeba Samuel. From human experts to machines : An LLM supported approach to ontology and knowledge graph construction, March 2024. arXiv :2403.08345 [cs].
- [15] Igor Kononenko. Comparison of inductive and naive Bayesian learning approaches to automatic knowledge acquisition. *Current trends in knowledge acquisition*, 8 :190, 1990. Publisher : IOS Press Amsterdam.
- [16] Philipp Körner, Michael Leuschel, João Barbosa, Vítor Santos Costa, Verónica Dahl, Manuel V. Hermenegildo, Jose F. Morales, Jan Wielemaker, Daniel Diaz, Salvador Abreu, and Giovanni Ciatto. Fifty Years of Prolog and Beyond. *Theory and Practice of Logic Programming*, 22(6) :776–858, November 2022.
- [17] Anna Sofia Lippolis, Mohammad Javad Saeezade, Robin Keskisarkka, Aldo Gangemi, Eva Blomqvist, and Andrea Giovanni Nuzzolese. Assessing the Capability of Large Language Models for Domain-Specific Ontology Generation, April 2025. arXiv :2504.17402 [cs].
- [18] Anna Sofia Lippolis, Mohammad Javad Saeezade, Robin Keskisärkkä, Sara Zuppiroli, Miguel Ceriani, Aldo Gangemi, Eva Blomqvist, and Andrea Giovanni Nuzzolese. Ontology Generation using Large Language Models, March 2025. arXiv :2503.05388 [cs].
- [19] Liangming Pan, Alon Albalak, Xinyi Wang, and William Yang Wang. Logic-LM : Empowering Large Language Models with Symbolic Solvers for Faithful Logical Reasoning, October 2023. arXiv :2305.12295 [cs].
- [20] Luc De Raedt, Angelika Kimmig, and Hannu Toivonen. ProbLog : A Probabilistic Prolog and its Application in Link Discovery. January 2007.
- [21] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath : Pushing the Limits of Mathematical Reasoning in Open Language Models, April 2024. arXiv :2402.03300 [cs].
- [22] Chandan Singh, Jeevana Priya Inala, Michel Galley, Rich Caruana, and Jianfeng Gao. Rethinking Interpretability in the Era of Large Language Models, January 2024. arXiv :2402.01761 [cs].
- [23] Motoi Suwa, A. Carlisle Scott, and Edward H. Shortliffe. An approach to verifying completeness and consistency in a rule-based expert system. *Ai Magazine*, 3(4) :16–16, 1982.
- [24] Yuan Yan Tang, Chang De Yan, and C.Y. Suen. Document processing for automatic knowledge acquisition. *IEEE Transactions on Knowledge and Data Engineering*, 6(1) :3–21, February 1994.
- [25] Qwen Team. Qwq-32b : Embracing the power of reinforcement learning. URL : <https://qwenlm.github.io/blog/qwq-32b>, 2025.
- [26] Cunxiang Wang, Xiaozhe Liu, Yuanhao Yue, Xiangru Tang, Tianhang Zhang, Cheng Jiayang, Yunzhi Yao, Wenyang Gao, Xuming Hu, Zehan Qi, Yidong Wang, Linyi Yang, Jindong Wang, Xing Xie, Zheng Zhang, and Yue Zhang. Survey on Factuality in Large Language Models : Knowledge, Retrieval and Domain-Specificity, December 2023. arXiv :2310.07521 [cs].
- [27] Fengli Xu, Qianyu Hao, Zefang Zong, Jingwei Wang, Yunke Zhang, Jingyi Wang, Xiaochong Lan, Jiahui Gong, Tianjian Ouyang, Fanjin Meng, Chenyang Shao, Yuwei Yan, Qinglong Yang, Yiwen Song, Sijian Ren, Xinyuan Hu, Yu Li, Jie Feng, Chen Gao, and Yong Li. Towards Large Reasoning Models : A Survey of Reinforced Reasoning with Large Language Models, January 2025. arXiv :2501.09686 [cs].
- [28] Jundong Xu, Hao Fei, Meng Luo, Qian Liu, Liangming Pan, William Yang Wang, Preslav Nakov, Mong-Li Lee, and Wynne Hsu. Aristotle : Mastering Logical Reasoning with A Logic-Complete Decompose-Search-Resolve Framework, December 2024. arXiv :2412.16953 [cs].
- [29] Sen Yang, Xin Li, Leyang Cui, Lidong Bing, and Wai Lam. Neuro-Symbolic Integration Brings Causal and Reliable Reasoning Proofs, February 2025. arXiv :2311.09802 [cs].
- [30] Xiaocheng Yang, Bingsen Chen, and Yik-Cheung Tam. Arithmetic Reasoning with LLM : Prolog Generation & Permutation, May 2024. arXiv :2405.17893 [cs].
- [31] Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei

Yin, and Mengnan Du. Explainability for Large Language Models : A Survey. *ACM Transactions on Intelligent Systems and Technology*, 15(2) :1–38, April 2024.

- [32] Zoie Zhao, Sophie Song, Bridget Duah, Jamie Macbeth, Scott Carter, Monica P Van, Nayeli Suseth Bravo, Matthew Klenk, Kate Sick, and Alexandre L. S. Filipowicz. More human than human : Llm-generated narratives outperform human-llm interleaved narratives. In *Proceedings of the 15th Conference on Creativity and Cognition*, CC '23, page 368–370, New York, NY, USA, 2023. Association for Computing Machinery.