Peuplement automatique d'ontologie : l'IA générative est-elle plus efficace qu'une approche sémantique ?

Aya-Nour-Elimane Sahbi*, Céline Alec*
Pierre Beust**

*Normandie Univ, UNICAEN, ENSICAEN, CNRS, GREYC, 14000 Caen, France
Prénom.Nom@unicaen.fr
**Univ Rennes, Inria, CNRS, IRISA - UMR 6074, F-35000 Rennes
pierre.beust@univ-rennes.fr

Résumé. Le peuplement d'ontologie consiste à identifier, extraire et intégrer automatiquement des informations provenant de différentes sources afin d'instancier les classes et propriétés d'une ontologie, construisant ainsi un graphe de connaissances sur un domaine. Dans cet article, nous comparons deux techniques de peuplement d'ontologie à partir de textes : KOnPoTe, une approche sémantique qui s'appuie sur une analyse textuelle et une analyse des connaissances du domaine, et une deuxième approche d'IA générative utilisant le modèle Claude, qui repose sur un LLM (large modèle de langage). Des expérimentations, réalisées sur des annonces françaises de vente d'immobiliers, sont présentées. Les points forts et limites des deux approches sont discutés.

1 Introduction

Les ontologies organisent l'information de manière structurée en reliant les entités par le biais de relations sémantiques. Le peuplement d'une ontologie consiste à enrichir cette dernière en y ajoutant des instances concrètes, formellement décrites selon un modèle ontologique. Ces données instanciées peuvent ensuite être exploitées sous forme de graphes de connaissances, offrant une représentation efficace qui permet aux machines de raisonner et de répondre à des questions complexes.

Les approches traditionnelles de peuplement d'ontologies s'inscrivent dans la recherche d'informations et le traitement du langage naturel (TALN). Ces méthodes utilisent des algorithmes d'apprentissage automatique ou des algorithmes basés sur des règles pour identifier les entités et les relations à partir du texte. Cependant, ces approches nécessitent une supervision humaine approfondie ou le développement de règles spécifiques pour chaque application.

Une alternative à ces méthodes traditionnelles est l'utilisation de grands modèles de langage (Large Language Models ou LLM) qui sont entraînés sur de vastes ensembles de données et peuvent apprendre des représentations sémantiques complexes, générant ainsi automatiquement des triplets ou peuplant une ontologie. Cependant, l'utilisation de LLMs dans ce contexte comporte certaines limitations, telles que l'incohérence ou l'incomplétude des triplets générés. C'est pourquoi, il est essentiel d'étudier les avantages et inconvénients de ces modèles et de déterminer s'ils peuvent complètement remplacer les approches sémantiques traditionnelles.

Cet article aborde spécifiquement cette question en comparant une approche sémantique à une approche basée sur un LLM dans le cadre d'une problématique de peuplement automatique d'ontologie. Il commence par présenter l'état de l'art et les travaux connexes dans la première section. La deuxième section expose les deux approches testées, tandis que la troisième détaille les expérimentations et analyse les résultats obtenus. Enfin, la dernière section conclut et présente des travaux futurs.

2 État de l'art

Le peuplement d'ontologie à partir de textes est un champ de recherche largement étudié. Lubani et al. (2019) examinent les différentes méthodes d'extraction et d'intégration de données pour peupler des ontologies. Dans une étude connexe, Asim et al. (2018) classifient les techniques de peuplement d'ontologies en approches linguistiques, statistiques et logiques.

Nous présentons ci-dessous deux catégories d'approches de peuplement d'ontologies. La première regroupe les approches dites sémantiques, tandis que la deuxième englobe les techniques utilisant les modèles d'intelligence artificielle générative. À notre connaissance, la comparaison entre ces deux catégories n'a pas encore été abordée dans la littérature existante.

2.1 Peuplement d'ontologie d'un point de vue sémantique

Dans cette section, nous examinons les méthodes sémantiques qui se concentrent sur le peuplement à partir de documents textuels en utilisant une ontologie de domaine préalablement fournie en entrée. Certaines méthodes reposent sur des techniques de TALN et font appel aux algorithmes d'apprentissage automatique (Machine Learning ou ML) ou profond (Deep learning ou DL) afin d'identifier les entités et les relations à partir du texte, construisant ainsi un graphe de connaissance, comme l'ont démontré (Imsombut et Sirikayon, 2016) et (Sambandam et al., 2023).

Ces approches d'apprentissage ML et DL ont montré des performances remarquables, même lorsqu'elles sont appliquées à des domaines distincts, tels que le domaine biomoléculaire étudié par Ayadi et al. (2019) ou encore la cybersécurité examinée par Gasmi et al. (2019). Cependant, elles exigent une supervision humaine approfondie ainsi que la disponibilité d'un grand nombre de documents pour réussir l'entrainement de ces modèles, comme l'ont souligné Jayawardana et al. (2017) et Suchanek et al. (2006).

Dans certains domaines, en adoptant des approches de peuplement supervisées, on peut rencontrer des limitations liées à la disponibilité de données annotées pour l'apprentissage. Pour réduire l'effort de supervision nécessaire par ces approches, des approches de peuplement non supervisées (Ayadi et al., 2019), semi-automatiques (Maedche et Staab, 2001) et hybrides (Castano et al., 2009) ont été implémentées.

En plus de ces méthodes, il existe des systèmes basés sur des motifs lexico-syntaxiques ou des règles, tels qu'ArtEquAKT (Alani et al., 2003), qui se concentre sur le peuplement d'ontologie à partir du Web dans le domaine des artistes ou encore l'approche de Makki et al. (2009), qui se focalise sur les verbes pour peupler l'ontologie. Enfin, l'approche KOnPoTe (Alec, 2023) se base sur une chaîne de traitements pour peupler une ontologie de domaine à partir de descriptions.

2.2 Peuplement d'ontologie d'un point de vue IA Générative

L'émergence des LLMs offre de nouvelles opportunités pour créer des représentations sémantiques ou générer des triplets au format RDF (Resource Description Framework).

Ces modèles se basent sur des requêtes, ou ce que l'on appelle des `prompts`. L'ingénierie de ces derniers implique la conception d'instructions spécialisées pour guider les LLMs dans la génération des sorties selon la spécification demandée.

Plusieurs LLMs ont été introduits (par exemple, GPT d'OpenAI, Claude d'Anthropic, Bard de Google et Llama de Facebook), où chaque modèle est entraîné sur différents ensembles de données et développé pour des tâches spécifiques. Des études comparatives ont été établies pour tester l'efficacité de ces modèles dans différentes applications. Ahmed et al. (2023) expliquent que lorsque l'on compare Bard et ChatGPT, plusieurs différences remarquables apparaissent selon le cas d'utilisation. ChatGPT montre une efficacité dans les tâches de traduction de langues, les descriptions de produits et les résumés de transcriptions. Cependant, Bard est plus efficace en termes d'extraction d'information, de génération et d'optimisation de code. Borji et Mohammadian (2023) ont investigué la capacité de Claude, GPT, Bard et Llama2 dans plusieurs applications, et ont expliqué que dans le contexte de la reconnaissance d'entités nommées et de la compréhension de la langue, les modèles ont presque la même efficacité.

Plusieurs travaux ont analysé la capacité des LLMs à générer une ontologie peuplée. Trajanoska et al. (2023) ont évalué les performances de ChatGPT dans l'extraction de graphes de connaissances à partir de textes. Funk et al. (2023) proposent une méthode pour construire automatiquement une hiérarchie de concepts pour un domaine donné en interrogeant GPT 3.5. Leurs expériences indiquent que les LLMs peuvent être d'une aide considérable pour construire cette hiérarchie.

En général, les LLMs ont le potentiel de contribuer au Web Sémantique en organisant l'information dans des formats structurés, améliorant ainsi l'efficacité et la précision de la représentation des données. Cependant, il est nécessaire d'investiguer à quel point ces représentations sont valides syntaxiquement et sémantiquement.

2.3 Choix des approches comparées

Notre étude s'intéresse à la tâche de peuplement d'une ontologie de domaine à partir de documents textuels décrivant chacun un objet de ce domaine. Ces documents présentent quelques particularités : des phrases contenant peu ou pas de verbes, des propriétés d'objet (object properties) et des propriétés des données (data properties), peu d'entités nommées et des relations n-aires [1]. Parmi les approches sémantiques de l'état de l'art, KOnPoTe (Alec, 2023) semble être la plus adaptée à ce contexte. Concernant l'approche à base de LLM, le modèle Claude [2] paraît plus pertinent, vu l'avantage qu'il offre par rapport aux autres modèles en permettant de joindre un fichier au prompt allant jusqu'à 50 Mo, ce qui est utile puisque nous donnons, en plus du texte, une ontologie en entrée.

1. Relations complexes impliquant trois entités ou plus, par exemple, exprimant qu'un bien se situe à une distance d'un lieu.

2. `https://www.anthropic.com/index/introducing-claude`

3 Peuplement automatique d'ontologie à partir du texte

3.1 Contexte et données

La figure 1 illustre notre problématique de peuplement d'ontologie. L'objectif est de peupler une ontologie de domaine donnée en entrée avec des informations contenues dans des documents textuels où chaque document du corpus contient une description textuelle d'un objet du domaine considéré.

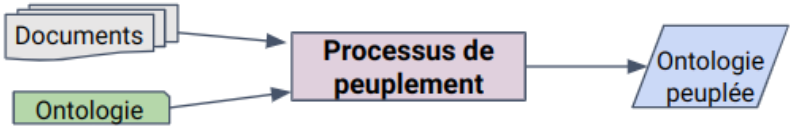

FIG. 1 – *Présentation de la problématique de peuplement d'ontologie*

On désigne par ontologie toute représentation formelle et structurée de connaissances qui définit des classes et les relations entre ces dernières. Elle permet de spécifier ainsi les classes, les propriétés (attributs ou relations entre ces classes), les instances (exemples spécifiques de ces classes) et les axiomes (règles logiques) qui définissent la structure et la sémantique du domaine concerné. Plus formellement, elle peut être définie dans notre contexte comme un tuple (C, P, I, A, R) où C est un ensemble de classes, P un ensemble de propriétés (propriétés d'objet et propriétés des données) caractérisant les classes, I un ensemble d'individus et d'assertions (potentiellement vide), A un ensemble d'axiomes représentables en OWL (Web Ontology Language) et R un ensemble de règles SWRL (Semantic Web Rule Language). On désigne par peuplement d'ontologie le processus d'ajout d'instances spécifiques à une ontologie déjà créée. Cela permet de lier les classes définies dans l'ontologie à des individus concrets du domaine d'application. Autrement dit, cela correspond dans notre contexte à ajouter des éléments à I.

Nous appliquons notre étude au domaine de la vente d'immobilier. Ainsi, notre corpus est composé des annonces de ventes de maisons et l'ontologie décrit les connaissances sur ce domaine. Celle-ci contient des classes telles que la classe principale `Bien` (désignant un bien immobilier), `Maison`, `Chambre`, etc. Elle contient aussi des propriétés. Par exemple, on peut citer la propriété d'objet `contient` reliant un bien à ses parties ou une maison à ses pièces, ou la propriété des données `surfaceEnM2` reliant une partie de bien (terrain, maison, etc.) ou une pièce à une valeur numérique. L'ontologie contient également des axiomes, comme le fait qu'un bien ne peut être situé que dans au maximum une Commune. Enfin, certaines règles SWRL sont définies. Le corpus de documents exploité est composé de 20 annonces françaises de ventes de maison. La suite de cette section décrit les deux approches testées.

3.2 L'approche KOnPoTe

L'approche KOnPoTe permet de peupler une ontologie de domaine à partir de descriptions textuelles d'éléments de ce domaine. La figure 2 résume les grandes étapes de l'approche. Dans cette section, nous nous contenterons d'expliquer brièvement le principe de KOnPoTe. Les détails des différents traitements sont décrits dans (Alec, 2023).

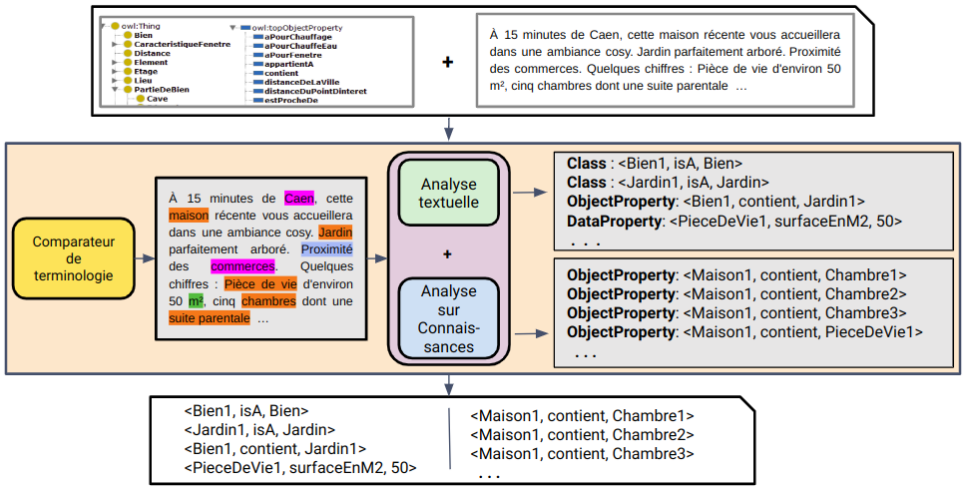

FIG. 2 – *Les grandes lignes de l'approche KOnPoTe, appliquée sur un document exemple*

Tout d'abord, un comparateur de terminologie est appliqué à la fois sur les textes du corpus d'entrée et sur l'ontologie. Celui-ci a pour but d'établir des correspondances entre les mentions de textes et les entités ontologiques (classes, propriétés et individus), en prenant en compte les variations syntaxiques des mots-clés. Par exemple, celui-ci va permettre, entre-autres, de détecter dans le document de la figure 2 les classes `Maison`, `Jardin`, etc. Ensuite, un algorithme de peuplement est appliqué, qui se décompose en deux étapes principales : une analyse textuelle et une analyse basée sur les connaissances du domaine.

L'analyse textuelle étudie les correspondances obtenues et se base sur des indicateurs textuels pour effectuer le peuplement. Dans un premier temps, elle peuple l'ontologie en fonction des types d'entités ontologiques concernées par les correspondances. Par exemple, une correspondance avec une classe va permettre d'instancier cette classe, tandis qu'une correspondance avec une propriété va permettre d'instancier la propriété, en exploitant le texte pour trouver les sujets et objets des assertions les plus adaptés. Ensuite, pour compléter les assertions de propriétés d'objet, d'autres traitements sont effectués, en cherchant notamment à associer les individus issus de correspondances qui se succèdent au sein d'une même phrase, ou encore les individus trop isolés (n'étant jamais objet d'une assertion) avec ceux issus d'une correspondance provenant de la même phrase. Les propriétés des assertions ajoutées dans ce cadre sont les plus adaptées en fonction des connaissances ontologiques. Dans l'exemple proposé en figure 2, cette étape va permettre, entre-autres, de créer les instances des classes `Maison`, `Jardin`, `PièceDeVie` et une assertion de la propriété des données `surfaceEnM2` pour cette pièce avec la valeur 50.

L'analyse basée sur les connaissances n'exploite plus aucun indicateur textuel, mais se base uniquement sur les connaissances de l'ontologie et l'ensemble des assertions déjà créées. Elle a pour but de faire en sorte que le graphe représentant les assertions de propriétés d'objet d'un document du corpus se rapproche le plus possible de la forme d'une étoile, où le centre correspond à l'individu représentant le sujet du document traité (par exemple, le bien immobilier

décrit dans le document, autrement dit, `Bien1` dans l'exemple de la figure 2). Là aussi, les propriétés choisies sont les plus adaptées en fonction des connaissances ontologiques. Dans l'exemple proposé en figure 2, cette étape va permettre d'ajouter les assertions de la propriété d'objet `contient` reliant l'instance de la classe `Maison` aux différentes instances des sous-classes de `PièceDeMaison` (les chambres, la pièce de vie, etc.).

3.3 L'approche basée sur LLM

Cette méthode exploite le LLM Claude, qui fait partie de la famille des LLMC (Large Language Models Clara) développés par Athropic, une start-up spécialisée dans l'IA. Claude a été entraîné sur des dialogues humain-humain ainsi que sur des documents et données diversifiées. Il utilise l'apprentissage profond, en particulier l'architecture Transformer, qui a démontré son efficacité dans le traitement du langage naturel. Les capacités conversationnelles de Claude couvrent un large éventail de sujets et de domaines. Ce modèle d'IA se distingue notamment par sa capacité à gérer de longues requêtes et à prendre, en plus de la requête, un fichier en entrée.

Nous utilisons Claude dans un pipeline zéro-shot, ce qui signifie que nous exploitons la compréhension linguistique pré-acquise de ce modèle, obtenue pendant son entraînement général par Athropic, sans lui fournir un entraînement spécifique pour la tâche de peuplement. L'approche, illustrée par la figure 3, prend en entrée un prompt contenant une explication textuelle de la tâche à exécuter (en l'occurrence, peupler l'ontologie), ainsi qu'un document du corpus et l'ontologie. Ces derniers sont fournis en pièces jointes. Il est à noter que cette approche est équivalente à KOnPoTe en termes d'entrées (ontologie et documents du corpus, exploités un par un), et de sortie (l'ontologie peuplée).

Le LLM génère des triplets en fonction du prompt donné en entrée. Nous tenons à préciser qu'une étape d'ingénierie des prompts, détaillée dans la section 4.1, est établie pour déterminer le prompt le plus efficace pour notre tâche. Cette étape est jugée indispensable dans le cas d'une telle approche, car la réponse du modèle dépend fortement du prompt d'entrée. Nous présentons dans le protocole expérimental les différents prompts utilisés. L'ingénierie des prompts implique l'utilisation de schémas définis par l'utilisateur en tant que structures directrices pendant le processus de prompting et intègre des connaissances spécifiques au domaine ainsi qu'une orientation explicite du schéma souhaité en sortie. Elle vise à améliorer la fidélité des triplets générés par le LLM.

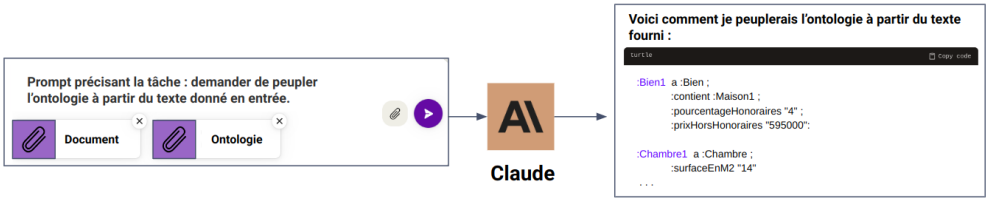

FIG. 3 – *Peuplement d'ontologie en utilisant Claude*

4 Expérimentations et comparaison

Cette section présente nos expérimentations sur le corpus d'annonces de vente de maisons. L'approche KOnPoTe est testée via l'exécutable fourni dans le fichier ZIP disponible [3]. Le modèle Claude est testé directement via l'interface graphique de l'API Claude [4].

4.1 Ingénierie des prompts

Comme précédemment mentionné, le résultat donné par le modèle Claude dépend fortement du prompt donné en entrée. Nous détaillons dans cette section l'ingénierie de prompt utilisée pour déterminer le prompt le plus adéquat. Plusieurs travaux ont discuté l'impact de l'ingénierie des prompts sur la performance du LLM et l'écart des réponses générées en sortie pour la même tâche en utilisant des prompts différents. Gao (2023) et Arawjo et al. (2023) ont montré que la réponse du LLM a une forte corrélation avec le prompt donné en entrée. White et al. (2023) ont proposé un catalogue de techniques d'ingénierie de prompts, présentées sous forme de patterns, qui ont été appliquées pour résoudre des problèmes courants lors de l'interaction avec ChatGPT. Ces techniques sont proposées et testées seulement sur les LLMs GPT et Bard. Cependant, ces patterns d'ingénierie de prompt n'ont pas été, à notre connaissance, testés sur le modèle Claude, ce qui est abordé dans cet article.

Nous nous sommes basés sur ces patterns d'ingénierie et avons pris en compte les éléments suivants qui précisent notre objectif et qui diffèrent des travaux connexes : il s'agit d'une tâche de peuplement d'ontologie à partir d'un texte et d'une ontologie donnée. De plus, la langue utilisée dans le prompt, les entités de l'ontologie et le texte est le français. Finalement, la version du LLM testée est Claude-v1.

Nous avons élaboré le prompt de manière progressive, en étudiant les résultats obtenus au fur et à mesure. La figure 4 illustre les 4 prompts utilisés ainsi que le résultat généré par Claude pour chaque prompt pour peupler l'ontologie à partir du document présenté dans la figure 2.

Le Prompt 1, étant un prompt basique qui précise uniquement la tâche, a généré des triplets de la forme (sujet, prédicat, objet) à partir d'instances de l'ontologie. Cependant, le format du résultat de ce prompt ne peut pas être exploité directement dans l'étape de validation. Pour remédier à cela, nous avons précisé dans le Prompt 2 le format souhaité : Turtle.

En analysant le résultat du Prompt 2, nous constatons que le LLM n'a pas détecté automatiquement la classe principale (`Bien`) de notre ontologie et n'a pas généré les assertions liées à cette classe (les parties du bien, le prix, etc.). À ce stade, le LLM n'est pas en mesure de détecter cette classe, car il n'est pas informé du contexte. Selon White et al. (2023), le modèle a besoin de ce que l'on appelle « The Meta Language Creation Pattern », autrement dit, le contexte des mots utilisés dans le prompt. Nous avons précisé ce contexte dans le Prompt 3.

Le Prompt 3 peuple toutes les classes de notre ontologie, mais il génère aussi des informations qui ne sont pas contenues dans le texte. Par exemple, le résultat du Prompt 3 sur la figure 4 mentionne que le bien a un chauffage au gaz et un chauffe-eau électrique, alors que cela n'est pas indiqué dans le texte de l'annonce. Nous avons donc précisé dans le Prompt 4 de peupler l'ontologie uniquement avec les informations contenues dans le texte, en ajoutant le mot « exclusivement ».

3. `https://alec.users.greyc.fr/research/konpote/`
4. `https://claude.ai/chats`

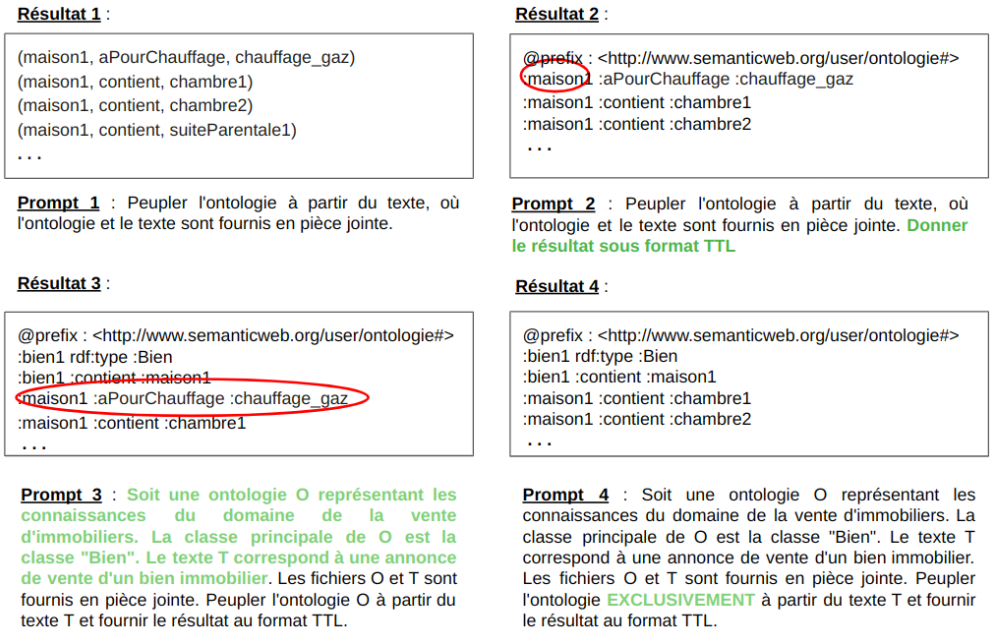

FIG. 4 – *Les prompts testés*

Le dernier prompt (Prompt 4) est le plus adéquat pour la tâche de peuplement automatique d'ontologie, il est ensuite utilisé et évalué selon le protocole expérimental détaillé ci-après.

4.2 Protocole expérimental

Le corpus ainsi que l'ontologie utilisés dans nos tests sont disponibles dans le fichier ZIP de KOnPoTe, à ceci près que nous n'avons considéré que les vingt premières descriptions d'annonces du corpus. Le peuplement est évalué selon le même protocole d'évaluation que (Alec, 2023). Plus précisément, un Gold Standard (GS) est exploité : il s'agit de l'ontologie initiale annotée manuellement avec des assertions représentant le corpus. Nous avons comparé les assertions générées par les deux approches avec celles du GS, en calculant la précision, le rappel et la F-mesure, donnés par les formules suivantes [5] :

$$\textit{Précision} = \frac{VP}{VP + FP} \qquad \textit{Rappel} = \frac{VP}{VP + FN} \qquad \textit{F-mesure} = \frac{2 \times \textit{Précision} \times \textit{Rappel}}{\textit{Précision} + \textit{Rappel}}$$

Chaque métrique est calculée de manière macroscopique (mac) et microscopique (mic). Le calcul macroscopique consiste en la moyenne des métriques pour chaque annonce (toutes les annonces ont le même poids), tandis que le calcul microscopique prend en compte la somme de tous les VP, FP, FN (toutes les assertions ont le même poids). Ainsi, une annonce générant

5. Un vrai positif (VP) est une assertion correcte, un faux positif (FP) est une assertion erronée, un faux négatif (FN) est une assertion manquante.

beaucoup d'assertions aura plus d'impact dans une métrique microscopique, tandis qu'une annonce ne générant que peu d'assertions aura plus d'impact dans le calcul macroscopique. Nous vérifions aussi la cohérence de l'ontologie résultante en utilisant le moteur d'inférence Pellet.

4.3 Résultats et discussion

Le tableau 1 montre les résultats des deux approches. L'approche KOnPoTe est visiblement plus performante que l'approche basée sur le modèle Claude. En effet, KOnPoTe obtient de meilleurs résultats sur toutes les métriques calculées. L'approche utilisant Claude a une précision relativement basse (77%), ce qui signifie qu'un certain nombre d'assertions générées sont fausses (presque un quart). De plus, le rappel est bas (56-59%), ce qui signifie que beaucoup d'assertions sont manquantes (presque la moitié).

	Précision (mac)	Rappel (mac)	F-mesure (mac)	Précision (mic)	Rappel (mic)	F-mesure (mic)
KOnPoTe	**0,94**	**0,90**	**0,92**	**0,93**	**0,87**	**0,90**
Claude	0,77	0,59	0,66	0,77	0,56	0,65

TAB. 1 – *Résultat des deux approches sur 20 annonces*

En analysant finement les résultats obtenus, nous avons déduit les points forts et faibles des deux approches. Ceux-ci sont résumés dans le tableau 2. D'une façon générale, on constate que KOnPoTe performe plutôt bien, mais génère très souvent plusieurs individus équivalents. Par exemple, sur le document de la figure 5, il est mentionné « cinq chambres » puis « quatre chambres ». KOnPoTe génère 9 instances de la classe `Chambre`, sans prendre en compte le fait que les quatre chambres sont incluses dans les cinq chambres mentionnées dans la phrase d'avant. Comme les ontologies n'appliquent pas l'hypothèse de nom unique, cette redondance n'est pas fausse à proprement parler, mais nous comptons tout de même cela comme des « sameAs » manquants. À l'inverse, Claude ne génère pas ce genre de redondance d'individus, mais présente essentiellement des lacunes en termes de raisonnement. En effet, l'ontologie générée est incohérente pour 90% des annonces. Cela est dû au fait que les axiomes de domaine et co-domaine ne sont pas toujours respectés par cette approche. En revanche, ce genre d'erreur n'est pas possible avec KOnPoTe car chaque étape de l'approche s'assure que les assertions candidates ne génèrent pas d'incohérence, et ne les ajoute pas le cas échéant. Au niveau des assertions des propriétés n-aires, aucune approche ne fonctionne parfaitement mais KOnPoTe s'en sort très bien tandis que Claude a plus de mal, notamment à lier les individus entre eux. Enfin, les deux approches échouent sur les cas les plus compliqués, qui nécessitent une forte compréhension du texte.

La figure 5 montre un exemple de texte d'une annonce immobilière du corpus (déjà évoqué dans la figure 2) ainsi que les assertions (partielles) des sorties des deux approches et du GS.

Dans l'ontologie fournie en entrée, la propriété d'objet `contient` a un domaine et co-domaine relativement larges. Cependant, l'ontologie définit des axiomes qui vont préciser ce domaine et co-domaine en fonction du contexte. Entre autres, il n'y a que les instances de `Bien` qui peuvent contenir des instances de `PartieDeBien`. La classe `Garage` est une

	Points forts	Limites
KOnPoTe	- Raisonnement pris en compte - Détection des propriétés n-aires	- Redondances des individus générés - Confusion dans les cas compliqués
Claude	-Pas de redondances d'individus	- Pas de raisonnement pris en compte - Détection partielle des propriétés n-aires - Confusion dans les cas compliqués

TAB. 2 – *Comparaison des points forts et limites de KOnPoTe et Claude*

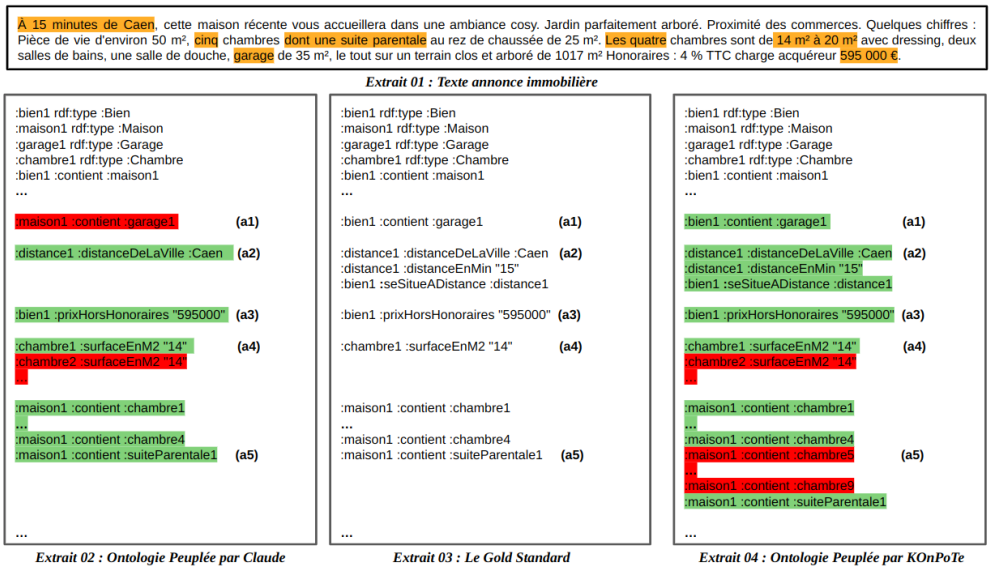

FIG. 5 – *Exemple des ontologies peuplées par Claude et KOnPoTe à partir d'un texte et le Gold standard correspondant*

sous-classe de `PartieDeBien`, un garage ne peut donc être contenu que par un bien. Or, on peut voir que l'individu `Garage1` (a1) est contenu par une instance de `Maison`, et non une instance de `Bien`, ce qui rend l'ontologie incohérente d'après le moteur d'inférence. Au contraire, KOnPoTe a créé une assertion cohérente et correcte.

Au niveau de la propriété ternaire (a2) : disant que le bien se situe à `15` minutes de `Caen`, on peut voir que le peuplement est correctement fait par KOnPoTe : le `bien1` se situe à distance de la `distance1`, et la `distance1` est associée à la notion de `15` minutes et à la ville de `Caen`. Pour Claude, l'instance `distance1` créée concerne bien la ville de `Caen`, mais elle n'est ni associé au `bien1` ni à la notion de `15` minutes.

En ce qui concerne les assertions des propriétés des données, les deux approches parviennent à extraire les informations si elles sont mentionnées explicitement comme le prix ((a3) sur la figure). Cependant, une confusion peut avoir lieu dans certains cas. Par exemple, lorsqu'il est mentionné que les surfaces des chambres varient entre 14 m² et 20 m², les deux

approches instancient la propriété `surfaceEnM2` à 14 m² pour toutes les chambres (a4).

En revanche, KOnPoTe a tendance à générer des individus redondants. Par exemple, elle génère 9 instances de `Chambre` et une instance de `SuiteParentale`, là où Claude génère, à raison, uniquement 4 instances de `Chambre` et une instance de `SuiteParentale` (a5).

5 Conclusion

Cet article a présenté une étude comparative de deux approches différentes pour résoudre une problématique de peuplement automatique d'ontologie : une approche sémantique et une approche à base de LLM. On constate que les approches présentent chacune des qualités et des défauts. Tout d'abord, l'approche sémantique est plus efficace et exploite la capacité de raisonnement des ontologies. Cependant, elle présente des défauts, notamment au niveau de la redondance des individus. L'approche utilisant un LLM est moins efficace, et a du mal avec les notions nécessitant de raisonner. Mais elle n'a pas tendance à créer des redondances d'individus. De plus, l'approche KOnPoTe testée est adaptée au contexte des textes descriptifs (le cas d'utilisation testé). Cependant, un autre contexte nécessiterait une autre approche. L'utilisation de Claude étant simplifiée et directe via une API disponible, elle reste plus immédiate que la conception et l'implémentation d'une approche sémantique adaptée au contexte des documents exploités. Une perspective intéressante consisterait à proposer une approche hybride, adaptée à divers contextes, s'appuyant sur les résultats de l'approche LLM et cherchant à corriger les assertions incohérentes générées en fonction des connaissances de l'ontologie. Cette approche pourrait ensuite bénéficier d'un module inspiré de l'analyse des connaissances de KOnPoTe afin d'ajouter des assertions manquantes cohérentes.

Références

Ahmed, I., M. Kajol, U. Hasan, P. P. Datta, A. Roy, et M. R. Reza (2023). ChatGPT vs. Bard : A Comparative Study. *UMBC Student Collection*.

Alani, H., S. Kim, D. E. Millard, M. J. Weal, W. Hall, P. H. Lewis, et N. R. Shadbolt (2003). Automatic ontology-based knowledge extraction from web documents. *IEEE Intelligent Systems 18*(1), 14–21.

Alec, C. (2023). Peuplement d'ontologie à partir de petites annonces immobilières. *Ingénierie des Connaissances, Jul 2023, Strasbourg, France*, 160–169.

Arawjo, I., C. Swoopes, P. Vaithilingam, M. Wattenberg, et E. Glassman (2023). Chainforge : A visual toolkit for prompt engineering and llm hypothesis testing. *preprint arXiv :2309.09128*.

Asim, M. N., M. Wasim, M. U. G. Khan, W. Mahmood, et H. M. Abbasi (2018). A survey of ontology learning techniques and applications. *Database 2018*, bay101.

Ayadi, A., A. Samet, F. d. B. de Beuvron, et C. Zanni-Merk (2019). Ontology population with deep learning-based NLP : a case study on the biomolecular network ontology. *Procedia Computer Science 159*, 572–581.

Borji, A. et M. Mohammadian (2023). Battle of the Wordsmiths : Comparing ChatGPT, GPT-4, Claude, and Bard. *GPT-4, Claude, and Bard (June 12, 2023)*.

Castano, S., I. S. E. Peraldi, A. Ferrara, V. Karkaletsis, A. Kaya, R. Möller, S. Montanelli, G. Petasis, et M. Wessel (2009). Multimedia interpretation for dynamic ontology evolution. *Journal of Logic and Computation 19*(5), 859–897.

Funk, M., S. Hosemann, J. C. Jung, et C. Lutz (2023). Towards ontology construction with language models. *preprint arXiv :2309.09898*.

Gao, A. (2023). Prompt engineering for large language models. *Available at SSRN 4504303*.

Gasmi, H., J. Laval, et A. Bouras (2019). Cold-start cybersecurity ontology population using information extraction with lstm. In *2019 International Conference on Cyber Security for Emerging Technologies (CSET)*, pp. 1–6. IEEE.

Imsombut, A. et C. Sirikayon (2016). An alternative technique for populating thai tourism ontology from texts based on machine learning. In *2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, pp. 1–4. IEEE.

Jayawardana, V., D. Lakmal, N. de Silva, A. S. Perera, K. Sugathadasa, B. Ayesha, et M. Perera (2017). Semi-supervised instance population of an ontology using word vector embedding. In *Int. Conference on Advances in ICT for Emerging Regions (ICTer 17)*, pp. 1–7. IEEE.

Lubani, M., S. A. M. Noah, et R. Mahmud (2019). Ontology population : Approaches and design aspects. *Journal of Information Science 45*(4), 502–515.

Maedche, A. et S. Staab (2001). Ontology learning for the semantic web. *IEEE Intelligent systems 16*(2), 72–79.

Makki, J., A.-M. Alquier, et V. Prince (2009). Ontology population via nlp techniques in risk management. *Int. Journal of Humanities and Social Science (IJHSS) 3*(3), 212–217.

Sambandam, P., D. Yuvaraj, P. Padmakumari, et S. Swaminathan (2023). Spiking equilibrium convolutional neural network for spatial urban ontology. *Neural Processing Letters*, 1–20.

Suchanek, F., G. Ifrim, et G. Weikum (2006). Leila : Learning to extract information by linguistic analysis. In *Proceedings of the 2nd Workshop on Ontology Learning and Population : Bridging the Gap between Text and Knowledge*, pp. 18–25.

Trajanoska, M., R. Stojanov, et D. Trajanov (2023). Enhancing knowledge graph construction using large language models. *preprint arXiv :2305.04676*.

White, J., Q. Fu, S. Hays, M. Sandborn, C. Olea, H. Gilbert, A. Elnashar, J. Spencer-Smith, et D. C. Schmidt (2023). A prompt pattern catalog to enhance prompt engineering with chatgpt. *preprint arXiv :2302.11382*.

Summary

Ontology population consists of automatically identifying, extracting and integrating information from different sources to instantiate the classes and properties of an ontology, thereby building a knowledge graph of a domain. In this article, we compare two ontology population techniques from texts: KOnPoTe, a semantic approach that relies on textual analysis and domain knowledge analysis, and a second Generative AI approach using the Claude model, which is based on a Large Language Model (LLM). Experiments conducted on French real estate advertisements are presented. The advantages and limits of both approaches are discussed.