

Ontologie de Maintenance des Bâtiments et Capacités des Larges Modèles de Langage (LLM) pour le Peuplement

J. Mba Kouhoue^{1,3}, M. Lefrançois², A. Lesage³, J. Lonlac¹, A. Doniec¹, S. Lecoeuche¹.

¹ IMT Nord Europe, Institut Mines-Telecom, Univ. Lille, Centre for Digital Systems, Lille, France.

² Mines Saint-Etienne, Univ Clermont Auvergne, INP Clermont Auvergne, CNRS, UMR 6158 LIMOS, F-42023, Saint-Étienne, France.

³ Intent Technologies, Toulouse, France.

{joel.mba-kouhoue, Jerry.lonlac, Arnaud.doniec}@imt-nord-europe.fr, maxime.lefrancois@emse.fr, alexis.lesage@intent-technologie.fr, stephane.lecoeuuche@mines-ales.fr

Résumé

Les données de maintenance des bâtiments proviennent de diverses sources, notamment de prestataires de services tels que les ascensoristes, les chauffagistes, ou des professionnels multiservices, ainsi que des clients pouvant être des gestionnaires immobiliers, des villes ou des acteurs du secteur tertiaire. La nature hétérogène de ces données, en raison de la diversité des sources, complique le processus de partage des données. Cet article propose une ontologie de domaine pour représenter ces données, explorant l'utilisation des LLMs pour peupler automatiquement l'ontologie. Les résultats indiquent une bonne performance de ChatGPT et TextCortex dans la génération d'instances à partir de données CSV semi-structurées. Cette approche vise à améliorer l'efficacité du peuplement de l'ontologie malgré la diversité des données de maintenance.

Mots-clés

Ontologies, Graphes de Connaissance, Maintenance des bâtiments, LLM, ChatGPT, TextCortex.

Abstract

Building maintenance data comes from various sources, including service providers such as elevator technicians, heating engineers, or multi-service professionals, as well as customers who can be property managers, cities, or actors in the tertiary sector. The heterogeneous nature of such data, due to the diversity of sources, complicates the data exchange process among building stakeholders. In this paper, we present our initial efforts to establish a domain ontology for representing building maintenance data, and explore the use of Large Language Models (LLMs) to automatically populate the ontology. The results indicate ChatGPT and TextCortex perform well in generating instances from semi-structured CSV data. This approach aims to enhance the efficiency of ontology population despite the diversity of maintenance data.

Keywords

Ontology, Knowledge graph, Building maintenance, LLM,

ChatGPT, TextCortex.

1 Introduction

Le secteur immobilier est en pleine transformation numérique, et plusieurs normes d'échange de données ont émergé, tant pour la phase de construction que pour la gestion et l'exploitation. L'écosystème de la maintenance des bâtiments rassemble différents types d'acteurs, notamment des prestataires de services tels que les ascensoristes, les chauffagistes, les professionnels multiservices, ainsi que les clients, qui peuvent être des bailleurs sociaux, des collectivités territoriales ou des clients du secteur tertiaire. La diversité de ces parties prenantes complique le processus d'échange de données en raison de l'hétérogénéité des systèmes utilisés. Pour répondre à cette problématique, nous proposons l'ontologie BM2O (Building Maintenance Operations Ontology) pour la représentation sémantique des données de maintenance des bâtiments, dont le principe de conception repose sur la réutilisation et l'enrichissement de ressources, suivant la méthodologie *Ontology development 101* [15]. L'ontologie se compose d'un ensemble de termes issus de plusieurs ontologies adaptées à nos besoins :

- L'ontologie Brick [2], qui modélise la hiérarchie des équipements d'un bâtiment,
- L'ontologie de référence en matière de maintenance proposée par [20].

Pour répondre aux exigences de complétude, l'ontologie proposée définit de nouveaux concepts (classes et propriétés) basés sur les connaissances d'experts et l'analyse d'un large historique d'opérations de maintenance de bâtiments produites par la société *Intent Technologies*¹.

Dans cet article, nous évaluons également la capacité des modèles GPT-3 et TextCortex (Sophos-2) à générer des graphes de connaissances à partir de données CSV. Pour évaluer notre approche, nous avons créé une base de connaissances contenant des instances (assertions) ajoutées manuellement avec l'aide d'experts du domaine, qui sont ensuite comparées aux assertions produites par les modèles

1. <https://intent.tech/>

GPT et TextCortex. Les résultats montrent qu’avec de bons prompts, des résultats satisfaisants sont obtenus en termes de précision, de rappel et de F-mesure.

Dans la suite de cet article, nous allons présenter le procédé de construction de l’ontologie BM2O en Section 2 et la stratégie de construction des graphes de connaissances à partir de LLM en Section 3.

2 Mise en place de l’ontologie BM2O

2.1 Etat de l’art

Dans cette section, nous présenterons à la fois quelques ontologies pour les bâtiments ainsi que les ontologies de maintenance.

Les IFC (Industry Foundation Classes) [3] sont un format d’échange de données ouvert pour faciliter la transmission d’informations dans un projet BIM, couvrant divers aspects comme le site, le bâtiment, et les équipements. Cependant, ils présentent des limites, notamment dans la représentation des espaces. Les ontologies sémantiques, utilisant un langage formel pour représenter des concepts et leurs relations, offrent une alternative. Diverses ontologies, telles que DogOnt pour le contrôle des équipements [5] et SA-REF pour les applications intelligentes [10, 9], ont été proposées, mais elles ne couvrent pas tous les équipements des bâtiments [4]. Brick [2] est une amélioration intégrative de ces standards, représentant tous les composants et leurs relations dans un bâtiment, mais il ne prend pas en compte la dynamique de la maintenance.

En ce qui concerne la maintenance, l’automatisation du traitement des données de maintenance et de défaillance des équipements par des ontologies suscite un intérêt croissant dans divers secteurs. Pour standardiser cette approche, l’ISO (Organisation internationale de normalisation) et la CEI (Commission électrotechnique internationale) ont publié les normes ISO/CEI 21838, et des initiatives comme l’Industrial Ontology Foundry travaillent sur des ontologies de domaine alignées sur ces normes [8, 18]. Les ontologies de maintenance existantes se concentrent soit sur une vue générale [14, 13], soit sur des processus spécifiques comme la gestion du travail ou l’analyse des défaillances [12, 11, 16]. Cependant, il n’existe pas encore d’ontologie spécifique pour la maintenance des bâtiments.

2.2 Modélisation ontologique

Le principe de conception de BM2O (Building Maintenance Operations Ontology) est basé sur la réutilisation et l’enrichissement des ressources ontologiques existantes, comme préconisé dans [15]. En effet, la réutilisation d’ontologies présente l’avantage d’utiliser des ressources ontologiques matures, éprouvées et validées par leurs applications, y compris certaines par le W3C (World Wide Web Consortium). Par conséquent, BM2O repose sur trois piliers principaux :

- **Réutilisation des ressources de l’ontologie Brick :** *Brick*² est un effort *open source* visant à standardiser les descriptions sémantiques des actifs phy-

2. <https://docs.brickschema.org/>

siques, logiques et virtuels dans les bâtiments ainsi que les relations entre eux. *Brick* se compose d’un dictionnaire extensible de termes et de concepts dans et autour des bâtiments, d’un ensemble de relations pour relier ces concepts, et d’un modèle de données flexible permettant une intégration transparente de *Brick* avec les outils et bases de données existants. Grâce à l’utilisation des technologies du Web sémantique, *Brick* permet ainsi de décrire de manière cohérente, un vaste ensemble de propriétés et d’actifs qu’on retrouve dans un parc immobilier.

- **Réutilisation des ressources de l’ontologie de référence pour la maintenance :** *L’Ontologie de Référence pour la Maintenance*³ prend en charge la modélisation des concepts associés à la fiabilité des actifs et à la gestion de la maintenance. Cette ontologie est conçue pour être minimale et se concentre uniquement sur les concepts fréquemment observés dans les données de maintenance au sein du cycle de vie des actifs. Elle est alignée sur les normes IOF CORE [7] et BFO [13] afin de garantir une cohérence et une compatibilité avec les pratiques industrielles établies.
- **Modélisation métier :** La mise en œuvre de BM2O a été réalisée en collaboration avec la société *Intent Technologies*, qui a développé une plateforme pour collecter, agréger et partager des données entre les acteurs clés de la profession immobilière. Sur la base d’une analyse métier, nous avons défini de nouvelles classes et relations pour aligner les concepts de différentes ontologies avec les spécificités du contexte métier. Nous avons également défini de nouveaux concepts basés sur le traitement NLP appliqué à un vaste corpus de descriptions textuelles d’opérations de maintenance.

L’ontologie BM2O est publiée en ligne à son URI grâce à OnToology :⁴

<https://w3id.org/def/bm2o>

3 Peuplement à partir de LLM

3.1 Etat de l’art

La montée en puissance des grands modèles de langage (LLMs) a suscité une attention notable. Des études comparatives ont révélé des différences d’efficacité selon les applications. [1] note que *ChatGPT* excelle dans la traduction, les descriptions de produits et les résumés, tandis que *Bard* est supérieur en extraction d’informations, génération de code et optimisation. [6] ont trouvé que *Claude*, *GPT*, *Bard* et *Llama2* sont équivalents pour la reconnaissance d’entités nommées et la compréhension linguistique. [19] combine les LLMs avec le raisonnement sémantique pour créer

3. <https://github.com/iofoundry/ontology/tree/master/maintenance>

4. Note 2024-02-28 : le serveur OnToology ne permet pas la négociation de contenu actuellement. La source en turtle est disponible par exemple ici : <https://joelmba.github.io/testtt/OnToology/bmoo.ttl/documentation/ontology.ttl>

des graphes de connaissances sur la durabilité, enrichis par des données d’actualité. [17] évalue *Claude* pour peupler une ontologie à partir d’annonces immobilières. Cette étude examine les performances de *ChatGPT* et *TextCortex AI* pour peupler une ontologie à partir de données CSV.

3.2 Ingénierie des prompts

Pour construire nos graphes de connaissances, nous avons utilisé des LLM avec une entrée comprenant l’ontologie, des données CSV et un prompt expliquant la tâche de peuplement de l’ontologie. Une étape cruciale a été l’ingénierie des prompts, visant à déterminer le plus efficace pour cette tâche, car la réponse des modèles dépend fortement du prompt utilisé. Plusieurs études ont souligné l’importance de cette ingénierie pour améliorer les performances des LLMs et la variance des réponses générées. Dans le cadre de nos expériences, nous avons défini de manière progressive les trois prompts suivants, adaptés à notre contexte :

- **prompt 1** : *Generate all the data properties and object assertions related to these operations, results in turtle.*
- **prompt 2** : *Generate all the data properties and object assertions related to these operations, results in turtle. Also take as input an example of a knowledge graph with the correct namespaces.*
- **prompt 3** : *Adapted generation involves extracting the relevant equipment mentioned in the description column and mapping it to the correct Equipment or subclass of Equipment in the ontology, as well as extracting the corresponding maintenance activity mentioned in the description column and mapping it to the correct Maintenance Activity class or subclass in the ontology.*

3.3 Méthodologie et résultats

Nous avons mené des expériences sur des opérations de maintenance contenues dans un fichier CSV contenant à la fois des champs de données structurées et non structurées, comme par exemple la description d’une opération de maintenance, la durée, etc. Le protocole expérimental a impliqué un peuplement manuel de l’ontologie avec des experts pour créer une base de référence, comparée ensuite aux résultats de ChatGPT et TextCortex. Les données CSV ont été transcrites en assertions ou triplets. Le Tableau 1 montre le nombre total d’assertions obtenues par chaque méthode, et les résultats ont été évalués en termes de précision, rappel et score F.

$$Precision = \frac{VP}{VP + PF} \quad Rappel = \frac{VP}{VP + FN}$$

$$F\ mesure = \frac{2 * Precision * Rappel}{Precision + Rappel}$$

Nous rappelons ici que : Un vrai positif (VP) est une assertion correcte, un faux positif (FP) est une assertion erronée et un faux négatif (FN) est une assertion manquante.

Le Tableau 2 présente les performances des différents modèles de langage en fonction des prompts utilisés pour

TABLE 1 – Nombre d’assertions générées

Operations	Manuel	ChatGPT			TextCortex		
		prompt_1	prompt_2	prompt_3	prompt_1	prompt_2	prompt_3
Op_1	47	20	45	46	26	53	54
Op_2	43	20	45	42	26	31	35
Op_3	38	20	45	41	26	27	21
Op_4	24	20	45	41	26	35	20
Op_5	26	20	45	41	26	18	24
Total	178	100	225	211	130	164	154

chaque opération. Les résultats montrent une nette amélioration lorsque les prompts sont ajustés pour inclure des exemples d’opérations correctement renseignées avec les espaces de noms appropriés, comme illustré par *prompt_2*. Cependant, même avec ces ajustements, ChatGPT semble surpasser TextCortex dans la plupart des cas. En utilisant le dernier prompt (*prompt_3*), visant à identifier les entités nommées dans les descriptions d’opérations, nous constatons une amélioration globale des performances, bien que certaines opérations aient subi une légère dégradation, notamment en raison d’assertions erronées.

TABLE 2 – Résultats

Métrique Operations		ChatGPT			TextCortex		
		prompt_1	prompt_2	prompt_3	prompt_1	prompt_2	prompt_3
Precision	Op_1	0	1	1	0	0.88	0.64
	Op_2	0	0.77	0.68	0	0.70	0.74
	Op_3	0	0.73	0.68	0	0.70	0.71
	Op_4	0	0.55	0.73	0	0.71	0.70
	Op_5	0	0.62	0.70	0	0.77	0.70
Rappel	Op_1	0	0.95	0.97	0	1	0.61
	Op_2	0	0.81	0.67	0	0.51	0.34
	Op_3	0	0.86	0.73	0	0.50	0.20
	Op_4	0	1	1	0	1	0.37
	Op_5	0	1	1	0	0.53	0.43
F-score	Op_1	0	0.97	0.98	0	0.94	0.63
	Op_2	0	0.79	0.68	0	0.59	0.47
	Op_3	0	0.79	0.70	0	0.58	0.32
	Op_4	0	0.71	0.84	0	0.83	0.48
	Op_5	0	0.76	0.82	0	0.63	0.54

4 Conclusion

Dans cet article, nous avons proposé une modélisation ontologique des concepts liés à la maintenance des bâtiments, basée sur la réutilisation et l’enrichissement des ressources ontologiques existantes, ainsi que sur les connaissances du domaine et l’analyse des activités de maintenance à partir d’un ensemble de données de maintenance de bâtiment réel. L’avantage de cette modélisation sémantique est qu’elle permet une représentation unifiée des opérations de maintenance des bâtiments, ce qui est essentiel dans un écosystème où différents acteurs, tels que les techniciens d’ascenseurs, les chauffagistes et les gestionnaires de biens, ont des perspectives différentes.

Cette étude nous a également permis de vérifier des hypothèses concernant la capacité des LLMs ChatGPT et TextCortex à peupler une ontologie de domaine à partir de données CSV semi-structurées. Les expériences montrent des résultats satisfaisants, à condition que de bons prompts d’entrée soient configurés. Cependant, ces modèles fonctionnent moins bien pour l’identification contextuelle des

entités nommées présentes dans les données textuelles. Dans la suite de nos travaux, nous prévoyons d'adopter une méthodologie plus formelle pour évaluer la qualité des graphes de connaissances générés. Cette méthodologie nous permettra de produire des graphes de connaissances sur de grands volumes de données, d'évaluer efficacement la qualité de ces graphes de connaissances tout en fournissant une méthode normalisée pour évaluer leur qualité globale. Nous prévoyons également de définir une approche méthodologique d'utilisation de ces graphes de connaissances pour la résolution des problèmes d'appariements de données de maintenance des bâtiments.

Références

- [1] Imtiaz Ahmed, Ayon Roy, Mashrafi Kajol, Uzma Hasan, Partha Protim Datta, and Md Rokonzaman Reza. Chatgpt vs. bard : a comparative study. *Authoria Preprints*, 2023.
- [2] Bharathan Balaji, Arka Bhattacharya, Gabriel Fierro, Jingkun Gao, Joshua Gluck, Dezhi Hong, Aslak Johansen, Jason Koh, Joern Ploennigs, Yuvraj Agarwal, et al. Brick : Towards a unified metadata schema for buildings. In *Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments*, pages 41–50, 2016.
- [3] Vladimir Bazjanac and Drury B Crawley. Industry foundation classes and interoperable commercial software in support of design of energy-efficient buildings. In *Proceedings of Building Simulation'99*, volume 2, pages 661–667. Addison-Wesley Boston, 1999.
- [4] Arka Bhattacharya, Joern Ploennigs, and David Culler. Short paper : Analyzing metadata schemas for buildings : The good, the bad, and the ugly. In *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, pages 33–34, 2015.
- [5] Dario Bonino and Fulvio Corno. Dogont-ontology modeling for intelligent domotic environments. In *International Semantic Web Conference*, pages 790–803. Springer, 2008.
- [6] Ali Borji and Mehrdad Mohammadian. Battle of the wordsmiths : Comparing chatgpt, gpt-4, claude, and bard. *GPT-4, Claude, and Bard (June 12, 2023)*, 2023.
- [7] Milos Drobnjakovic, Boonserm Kulvatunyou, Farhad Ameri, Chris Will, Barry Smith, and Albert Jones. The industrial ontologies foundry (iof) core ontology. 2022.
- [8] Milos Drobnjakovic, Boonserm Kulvatunyou, Simon Frechette, and Vijay Srinivasan. Recent developments in ontology standards and their applicability to biomanufacturing. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, volume 87295, page V002T02A058. American Society of Mechanical Engineers, 2023.
- [9] ETSI TC SmartM2M. SmartM2M; Smart Applications; Reference Ontology and oneM2M Mapping. Technical Specification ETSI TS 103 264 V3.2.1, ETSI, January 2024.
- [10] Raúl García-Castro, Maxime Lefrançois, María Poveda-Villalón, and Laura Daniele. The etsi saref ontology for smart applications : a long path of development and evolution. *Energy Smart Appliances : Applications, Methodologies, and Challenges*, pages 183–215, 2023.
- [11] Melinda Hodkiewicz, Johan W Klüwer, Caitlin Woods, Thomas Smoker, and Emily Low. An ontology for reasoning over engineering textual data stored in fmea spreadsheet tables. *Computers in Industry*, 131 :103496, 2021.
- [12] Melinda Hodkiewicz, Emily Low, and Caitlin Woods. Towards a reference ontology for maintenance work management. In *I-ESA Workshops*, 2020.
- [13] Mohamed Hedi Karray, Farhad Ameri, Melinda Hodkiewicz, and Thierry Louge. Romain : Towards a bfo compliant reference ontology for industrial maintenance. *Applied Ontology*, 14(2) :155–177, 2019.
- [14] Aristeidis Matsokis, Hedi M Karray, Brigitte Chebel-Morello, and Dimitris Kiritsis. An ontology-based model for providing semantic maintenance. *IFAC Proceedings Volumes*, 43(3) :12–17, 2010.
- [15] Natalya F Noy, Deborah L McGuinness, et al. Ontology development 101 : A guide to creating your first ontology, 2001.
- [16] Dnyanesh Rajpathak, Yiming Xu, and Ian Gibbs. An integrated framework for automatic ontology learning from unstructured repair text data for effective fault detection and isolation in automotive domain. *Computers in Industry*, 123 :103338, 2020.
- [17] Aya Nour Elimane Sahbi, Céline Alec, and Pierre Beust. Peuplement automatique d'ontologie : l'ia générative est-elle plus efficace qu'une approche sémantique? In *24ème conférence francophone sur l'Extraction et la Gestion des Connaissances (EGC)*, 2024.
- [18] Barry Smith, Farhad Ameri, Hyunmin Cheong, Dimitris Kiritsis, Dusan Sormaz, Chris Will, and J Neil Otte. A first-order logic formalization of the industrial ontology foundry signature using basic formal ontology. 2019.
- [19] Milena Trajanoska, Riste Stojanov, and Dimitar Trajanov. Enhancing knowledge graph construction using large language models. *arXiv preprint arXiv :2305.04676*, 2023.
- [20] Caitlin Woods, Matt Selway, Tyler Bikaun, Markus Stumptner, and Melinda Hodkiewicz. An ontology for maintenance activities and its application to data quality. *Semantic Web*, (Preprint) :1–34, 2022.