AfIA

Association française
pour l'Intelligence Artificielle

IC

Journées francophones d'Ingénierie des Connaissances

PFIA 2024

Table des matières

Éditorial

Journées francophones d'Ingénierie des Connaissances

Les journées francophones d'Ingénierie des Connaissances (IC) fêtent leur 35es édition et sont hébergées par la plateforme PFIA, conjointement avec d'autres conférences francophones dans le domaine de l'intelligence artificielle (IA).

IC est un lieu d'échanges et de réflexions, de présentation et de confrontation des théories, pratiques, méthodes et outils autour de l'ingénierie des connaissances :

— Représentation des connaissances, ontologies
— De la donnée à la connaissance
— Qualité des données et des connaissances
— Raisonnement et apprentissage
— Ingénierie des connaissances pour le Web
— Applications de l'Ingénierie des Connaissances et retours d'expérience

Cette communauté prend désormais en compte l'essor des algorithmes d'apprentissage automatique et leurs retombées sur les pratiques individuelles et collectives, tout en conservant l'humain au centre des systèmes de décision exploitant les données et les connaissances.

Cette année, la conférence IC a reçu 25 soumissions d'articles. 20 ont été acceptés répartis dans les catégories suivantes : 8 articles longs, 2 articles courts, 2 posters, 1 démonstration et 7 articles déjà publiés dans une conférence ou revue internationale de renom. Un travail conséquent a été mené par les membres du comité de programme, chaque article a reçu entre 3 et 4 relectures comportant des critiques argumentées et constructives pour les auteurs.

Le programme de la conférence, réparti sur 3 jours, suit un programme découpé en 8 sessions. Ces sessions portent sur des thèmes qui sont au cœur de l'ingénierie des connaissances tels que «Ontologies», «Extraction d'information, Annotation», «Découverte de connaissances» et «Fusion et intégration d'ontologies».

Deux sessions ont été organisées conjointement avec deux autres conférences de PFIA : la conférence sur les Applications Pratiques de l'Intelligence Artificielle (APIA) et la Conférence Nationale en Intelligence Artificielle (CNIA). Ces deux sessions ont été possibles grâce à une pré-sélection de Céline Rouveirol et Thomas Guyet et à un travail de synchronisation avec Catherine Roussey, Ghislain Atemezing et Nathalie Aussenac-Gilles.

Une table ronde commune à IC, CNIA, et RJCIA, a aussi eu lieu pour alimenter les réflexions de la communauté de recherche en IA sur les espoirs, enjeux et limites des LLMs. C'était l'occasion d'éclairer les complémentarités entre approches symboliques, représentation des connaissances et apprentissage automatique.

Merci aux intervenants de cette table ronde : Cassia Trojahn (IRIT, Université Toulouse 2 Jean Jaurès), Davide Buscaldi (LIPN, université Sorbonne Paris Nord), Pierre Zweigenbaum (CNRS LISN et université Paris-Saclay) ainsi que Eric Gaussier (LIG et Université Grenoble - Alpes).

Pour cette édition 2024 de la conférence, nous avons l'honneur d'accueillir Enrico Motta – Professor of Knowledge Technologies at the Knowledge Media Institute (KMi) of the UK's Open University – dont la conférence invitée est intitulée *Enabling sensemaking by integrating large-scale text mining and knowledge-based models : Case studies in research and news analytics.*

Je profite de cet éditorial pour remercier chaleureusement les membres du comité de programme de leur très forte implication. J'adresse également mes remerciements à l'ensemble des acteurs de la communauté francophone d'Ingénierie des Connaissances qui ont contribué au succès d'IC 2024, ainsi que le comité d'organisation de la PFIA 2024 qui a été d'une grande efficacité.

Haïfa Zargayouna

Comité de programme

Présidence

— Haïfa Zargayouna LIPN, Université Sorbonne Paris Nord.

Membres

— Nathalie Abadie LASTIG, Univ. Gustave Eiffel, IGN-ENSG ;
— Xavier Aimé Cogsonomy / LIMICS UMRS 1142 Inserm ;
— Yamine Ait-Ameur IRIT, Université de Toulouse, Toulouse INP ;
— Nathalie Aussenac-Gilles IRIT, Université de Toulouse, CNRS ;
— Bruno Bachimont Sorbonne Université ;
— Nathalie Bricon-Souf IRIT, Université de Toulouse, UT3 ;
— Sandra Bringay LIRMM ;
— Patrice Buche INRA ;
— Davide Buscaldi LIPN, Université Sorbonne Paris Nord ;
— Sylvie Calabretto LIRIS ;
— Pierre-Antoine Champin LIRIS, Université Claude Bernard Lyon1 ;
— Jean Charlet AP-HP & INSERM UMRS 1142 ;
— Victor Charpenay Mines Saint-Etienne, Univ Clermont Auvergne, INP Clermont Auvergne, CNRS, UMR 6158 LIMOS ;
— Jérôme David INRIA ;
— Sylvie Despres Laboratoire d'Informatique Médicale et de BIOinformatique (LIM&BIO) ;
— Gayo Diallo ISPED & LABRI, University of Bordeaux ;
— Gilles Falquet University of Geneva ;
— Catherine Faron Université Côte d'Azur ;
— Béatrice Fuchs LIRIS, université de Lyon ;
— Frédéric Fürst MIS - Université de Picardie - Jules Verne ;
— Jean-Gabriel Ganascia Pierre and Marie Curie University - LIP6 ;
— Mounira Harzallah LS2N, University of Nantes, France ;
— Nathalie Hernandez IRIT, Université de Toulouse, UT2 ;
— Liliana Ibanescu Université Paris-Saclay, AgroParisTech, INRAE, UMR MIA Paris-Saclay, 91120, Palaiseau, France ;
— Sébastien Iksal LIUM - Le Mans Université, France ;
— Antoine Isaac Europeana & VU University Amsterdam ;
— Clément Jonquet MISTEA (INRAE) and LIRMM (U. Montpellier) ;
— Gilles Kassel University of Picardie Jules Verne ;
— Michel Leclère University of Montpellier (LIRMM/INRIA), France ;
— Maxime Lefrançois MINES Saint-Etienne ;
— Dominique Lenne Heudiasyc, Université de Technologie de Compiègne ;
— Jérôme Nobécourt LIMICS ;
— Nathalie Pernelle LIPN, Université Sorbonne Paris Nord ;
— Yannick Prié LINA - University of Nantes ;
— Cédric Pruski Luxembourg Institute of Science and Technology ;
— Sylvie Ranwez LGI2P / Ecole des mines d'Alès ;
— Catherine Roussey INRAE ;
— Fatiha Saïs, LISN, CNRS & Université Paris Saclay
— Karim Sehaba LIRIS CNRS ;
— Konstantin Todorov LIRMM / University of Montpellier ;
— Raphaël Troncy EURECOM.

Ontologies

Une ontologie pour modéliser les bioagresseurs des plantes

F. Amardeilh[1], S. Bernard[2], R. Bossy[3], M. Courtin[3],
M. Hirschy[4], P. Larignon[5], C. Roussey[6], N. Sauvion[7].

[1] Elzeard, Bordeaux, France
[2] LISC, INRAE, Aubière, France
[3] MaIAGE, INRAE, Jouy-en-Josas, France
[4] Acta, Paris, France
[5] IFV, Occitanie, RODILHAN, France
[6] MISTEA, INRAE, Montpellier, France
[7] PHIM, Univ Montpellier, INRAE, CIRAD, IRD, Institut Agro, Montpellier, France

florence.amardeilh@elzeard.co, matthieu.hirschy@acta.asso.fr, philippe.larignon@vignevin.com,
prenom.nom@inrae.fr

Résumé

Le projet ANR "Des Données aux Connaissances en Agronomie et Biodiversité" (D2KAB) met à disposition une archive de bulletins agricoles publiée sur le Web. Pour annoter les bulletins à l'aide des maladies et des bioagresseurs des culture, nous avons besoin d'une nouvelle ressource sémantique. Plusieurs ontologies et graphes de connaissances existent déjà sur le sujet mais ne couvrent pas l'intégralité de nos besoins. Nous avons donc développé une nouvelle ontologie "BioAGgressor Ontology" (BAGO) en réutilisant le plus possible des éléments d'ontologies existantes. Cette nouvelle ontologie a été développée en utilisant la méthodologie LOT en partenariat avec 4 experts en agriculture, entomologie et maladie des plantes.

Mots-clés

ontologie, Web de données, données liées, bioagresseurs des cultures, organismes nuisibles, maladie des plantes, agriculture.

Abstract

The French ANR project "Data to Knowledge in Agronomy and Biodiversity" (D2KAB) builds an archive of French agricultural alert newsletters. In order to annotate plant disease and pest, we need a new semantic resource. Several ontologies and knowledge graphs already exist on the subject but do not cover all of our needs. We have therefore developed a new ontology "BioAGgressor Ontology" (BAGO) by reusing elements of existing ontologies as much as possible. This new ontology was developed using the LOT methodology in partnership with 4 experts in agriculture, entomology or plant disease .

Keywords

ontology, Web of Data, Linked Open Data, pests, harmful organisms, plant disease, agriculture.

1 Introduction

L'agronomie et l'agriculture sont confrontées à plusieurs défis sociétaux, économiques et environnementaux majeurs, nécessitant des innovations technologiques. Le projet ANR *Des Données aux Connaissances en Agronomie et Biodiversité* (D2KAB) [1] illustre comment la science des données contribue au développement d'applications agricoles innovantes. L'objectif de D2KAB est de créer un cadre pour transformer les données d'agronomie et de biodiversité en connaissances interopérables, exploitables et ouvertes. Pour construire un tel cadre, nous nous appuyons sur des ressources (par exemple, des thésaurus ou des ontologies) pour décrire nos données et les publier en tant que données ouvertes liées. Nous utilisons le portail web AgroPortal [2] [7] pour trouver, publier et partager des ressources puis nous les exploitons dans des applications dédiées à l'agriculture ou l'environnement.

L'un des scénarios agricoles de D2KAB consiste à construire un navigateur web augmenté pour les bulletins officiels d'alertes agricoles français, appelés *Bulletins de Santé du Végétal* (BSV). Le prototype interrogera une archive de BSV disponible sous forme de fichiers PDF. Chaque bulletin sera annoté et ses annotations seront publiées sur le Web de données liées. Les annotations seront produites, entre autre, à partir de techniques de traitement automatique de la langue appliquées sur les contenus textuels des BSV. Des mentions de maladies, de ravageurs, d'agents pathogènes, de vecteurs [3], de symptômes

1. www.d2kab.org
2. http://agroportal.lirmm.fr
3. Un vecteur est soit un organisme vivant (biotique : arthropodes, nématodes, oiseaux, humains, ...) soit un facteur non vivant (abiotique : vent, eau, véhicule de transport,...) qui est capable de transporter avec succès un bioagresseur d'un organisme affecté vers un organisme sain, ou dans l'environnement ou à des aliments de cet organisme sain. Le transport sera considéré comme efficace s'il permet le maintient du bioagresseur dans l'environnement considéré. Dans le cas des arthropodes vecteurs, pour

visibles et de plantes cultivées sont toujours présentes dans les BSV, pour informer les lecteurs de l'état sanitaire des cultures. Afin d'annoter les observations des bioagresseurs des cultures et de leurs vecteurs, nous avons besoin d'une ressource spécifique adaptée aux vocabulaires francophones utilisés dans les BSV. Un bioagresseur se définit comme un organisme vivant non désirable dans une parcelle agricole et qui impacte négativement la production agricole. Un bioagresseur peut être un organisme nuisible (agent pathogène ou ravageur) ou une plante adventice [4]. Notre nouvelle ontologie s'intitule "BioAGgessor Ontology" (BAGO). Elle sera instanciée ultérieurement pour construire des graphes de connaissances représentant les bioagresseurs connus d'une culture donnée.

Le reste de l'article est organisé comme suit : La section 2 passe en revue les ontologies existantes pour décrire les bioagresseurs des cultures ; La section 3 illustre la méthodologie utilisée pour construire l'ontologie BAGO ; La section 4 présente deux design patterns de BAGO. La section 5 discute les difficultés rencontrées. La section 6 propose des perspectives.

2 État de l'art sur les ontologies des bioagresseurs des plantes

Nous nous sommes intéressés aux ontologies décrivant les maladies et les bioagresseurs des plantes cultivées. Un bioagresseur se décrit en fonction de son impact sur la plante : un organisme nuisible (agent pathogène ou ravageur), une plante adventice. Le bioagresseur est introduit dans l'environnement de la plante par le biais d'un vecteur biotique ou abiotique (cf. note de bas de page 3). Nous avons dans un premier temps cherché des articles décrivant la conception d'une telle ontologie dans Google Scholar puis dans le portail Agroportal. Le tableau 1 indique pour chaque référence sa couverture en termes de : maladies (mal.), organismes nuisibles (nuis.), vecteurs (vec.), taxonomie scientifique utilisée (tax.), plantes cultivées (cult.) et symptômes (smp.). Dans les lignes du tableau, le caractère 'o' signifie oui, 'n' signifie non, ' ?' pas d'avis, 'pat.' signifie pathogène, 'ins.' signifie insecte ravageur.

La communauté BFO a produit une ontologie décrivant les processus biologiques impliqués dans des maladies infectieuses : "Infectious Disease Ontology (IDO)". Les travaux de [21] spécialisent IDO pour le cas des maladies des plantes en produisant une nouvelle ontologie intitulée IDOplant. A notre connaissance, IDOplant est la seule ontologie qui modélise les maladies comme un processus. Dans IDOplant les organismes vivants sont représentés par l'ontologie NCBITaxon décrivant la taxonomie du NCBI[19].

ref.	mal.	nuis.	vec.	tax.	cult.	smp.
[21]	o	pat.	n	ncbi	o	n
[1]	o	ins.	o	UniProt	o	o
[3]	?	?	?	?	vigne	?
[8]	n	o	n	ncbi	o	n
[9]	o	ins.	?	?	o	o
[15]	o	ins.	n	Agrovoc	arbo	o
[2]	n	o	n	eppo	o	n

TABLE 1 – Analyse des ontologies existantes

MedISys est une plate-forme d'épidémiosurveillance végétale qui suit l'évolution des bioagresseurs dans les journaux et les blogs. Ce système utilise l'ontologie "Core Plant Plant Health Threat" [1]. Malheureusement cette ontologie n'est pas disponible. Nous avons trouvé intéressant le patron qui décrit les symptômes en précisant l'organe de plante atteint et le type de symptômes. Ces organes sont importés depuis l'ontologie "Plant Ontology (PO)" [6]. Le graphe de connaissances décrivant la taxonomie scientifique UniProt [11] est utilisé pour décrire les organismes vivants.

Les travaux de [3] proposent une base de connaissances et des règles d'inférences associées pour l'aide à la décision dans la lutte intégrée des bioagresseurs de la vigne de table. Les règles implémentent les seuils de la régulation espagnole. Malheureusement cette base n'est pas accessible.

Les travaux de [8] ont produit une ontologie des bioagresseurs des plantes cultivées et des traitements par apprentissage des règles à partir de textes rédigés en espagnol. L'ontologie produite s'intitule "Pests in Crops and their Treatments (PCTO)". PCTO modélise les épidémies comme une relation N-aire entre une culture et un organisme nuisible. PCTO intègre aussi les traitements de luttes associées. Elle utilise NCBITaxon [19] pour décrire les bioagresseurs. Elle est disponible en téléchargement depuis l'article.

AgriEnt est un système d'aide à la décision de diagnostic et de luttes des insectes ravageurs des cultures principales d'équateur [10]. Ce système utilise une ontologie "AgriEnt ontology" et un ensemble de règles SWRL [9] pour déterminer l'insecte à partir des symptômes sur la culture. Cette ontologie n'est pas disponible. Elle s'inspire d'ontologies déjà publiées comme : Plant Ontology [6] et IDOplant [21].

Les travaux de [16] développent un système d'aide à l'identification des bioagresseurs des cultures à partir des descriptions textuelles de symptômes décrits en espagnol. Une fois que l'agriculteur a identifié le bioagresseur le système lui propose un ensemble de traitements. Les cultures considérées sont l'amandier, l'olivier et la vigne. Ce système intègre une ontologie CropPestO [15] et une base de connaissances associée. L'ontologie utilise les concepts du thésaurus Agrovoc pour identifier les cultures et les bioagresseurs. CropPestO est disponible par téléchargement à partir de

être considéré comme vecteur avéré, l'organisme vivant doit non seulement être porteur de l'agent pathogène mais aussi ensuite être capable de l'inoculer dans un nouvel hôte sain. Ainsi, un arthropode peut se retrouver porteur d'un agent pathogène en se nourrissant de la sève d'une plante infectée ou de sang contaminé, mais ne pas avoir la capacité vectorielle à inoculer cet agent pathogène. A sein d'un agrosystème, le transport ne se fait pas nécessairement uniquement entre parcelles cultivées. Les plantes du milieu sauvage (non cultivé) peuvent être réservoir de bioagresseurs et/ou de vecteurs biotiques. Un vecteur biotique peut aussi être en plus un ravageur des cultures (ex. puceron). Les vecteurs biotiques en santé du végétal sont principalement des insectes piqueurs-suceurs (hémiptères) ou des nématodes (vers ronds).

4. Une plante adventice est un bioagresseur végétal qui apparaît dans une parcelle agricole, sans être cultivée. Elle se développe en concurrence avec les plantes cultivées.

l'article [16].
L'entreprise Bayer a transformé la base de données de l'Organisation Européenne et Méditerranéenne pour la Protection des Plantes (OEPP) (plus connue sous le nom de EPPO Global database [5]) en ontologie pour des besoins internes d'échange de données [2]. L'ontologie définie une classe pour chaque bioagresseur et chaque culture.
Une recherche sur Agroportal nous a donné une ontologie supplémentaire : Crop Disease Ontology [6] publiée en 2020. Elle définie 9 classes et 12 propriétés objets sans aucun commentaire. De plus, elle n'est associée à aucune publication pour comprendre sa modélisation et son usage.
En résumé, IDOPLant [21] est la seule ontologie qui représente une agression comme un processus. PCTO [8] modélise une agression d'une plante cultivée par un bioagresseur sous la forme d'une relation N-aire. Les autres ontologies représente les interactions plantes bioagresseurs sous forme de relations binaires. Plusieurs ontologies décrivent les symptômes et nous avons apprécié le patron de "Core Plant Plant Health Threat" [1] qui indique dans l'expression de symptôme l'organe atteint. Chacune de ces ontologies utilise une diversité de taxonomie scientifique, mais NCBITaxon [19] est la plus souvent utilisée.

3 Développement d'ontologies

Linked Open Terms (LOT) [7] est une méthodologie utilisée pour le développement d'ontologies [14]. Cette méthodologie se concentre sur (1) la réutilisation d'éléments (classes, propriétés et attributs) existant dans des ontologies déjà publiées et (2) la publication de l'ontologie selon les principes du Web de données liées. Elle réutilise trois activités d'ingénierie des connaissances définies dans la méthodologie NeOn [20]. Cette méthodologie définit les itérations sur les quatre activités suivantes : (1) spécification des besoins ontologiques, (2) implémentation de l'ontologie, (3) publication de l'ontologie et (4) maintenance de l'ontologie.

3.1 Spécification des besoins ontologiques

Les besoins ont été spécifiés à l'aide de questions de compétences illustrées d'exemples. Ces questions ont été mises à jour au cours de la phase d'implémentation de l'ontologie. Dans l'état actuel BAGO répond à 15 questions de compétences :

1. Quels sont les taxons scientifiques qui caractérisent l'organisme vivant ? L'organisme vivant est un insecte, un champignon, ...
2. Quelle est l'espèce qui caractérise l'organisme vivant ? Ce plant de vigne appartient à l'espèce "Vitis vinifera".
3. Quels sont les noms vernaculaires décrivant l'organisme vivant ? Ce plant est une vigne cultivée.
4. Quel est le rôle d'un organisme vivant impliqué dans une attaque de bioagresseur ? Ce plant de "Vitis vinifera" est l'hôte. L'insecte "Daktulosphaira vitifoliae" est l'agresseur.
5. Quel est le type du bioagresseur ? L'insecte "Daktulosphaira vitifoliae" est un ravageur.
6. Quelles sont les plantes attaquées par le bioagresseur ? L'insecte "Daktulosphaira vitifoliae" attaque les plants de l'espèce "Vitis vinifera".
7. Quelles sont les maladies associées à une plante ? Les maladies de la vigne sont le phylloxera, le mildiou, etc...
8. Quel est l'agent pathogène qui provoque cette maladie ? Le mildiou est provoqué par le champignon "Plasmopara viticola".
9. Quels sont les symptômes de cette maladie ? Les symptômes du phylloxera sont la présence de gâlles sur les feuilles et l'apparition de tubérosités sur les racines.
10. Quelle partie de la plante présente des symptômes de l'attaque ? Une attaque de phylloxera apparaît sur les racines et les feuilles des vignes.
11. Quels sont les symptômes directs et indirects d'un organisme vivant ? L'apparition de tubérosités sur les racines sont des symptômes directs du phylloxera.
12. Quelle est la classe de maladie par type d'organes atteints ? Le phylloxera est une maladie des racines.
13. Quelle est la classe de maladie par type d'agents pathogènes ? Le mildiou est une maladie fongique.
14. Quels sont les stades de développement où la plante hôte est sensible à une attaque du bioagresseur ? Les baies de raisin sont sensibles au mildiou jusqu'au stade véraison.
15. A quels stades de développement le bioagresseur attaque une plante ? La forme gallicole de l'espèce "Daktulosphaira vitifoliae" attaque les feuilles de vigne.

3.2 Implémentation de l'ontologie

Avant de commencer le développement de notre ontologie, nous avons étudié plusieurs ontologies du domaine agricole en plus de celles de la section 1 pour identifier des patrons de conception ontologique ou les éléments d'ontologie à réutiliser. Ainsi nous avons repris les éléments suivants :

— le patron d'expression de symptômes de l'ontologie "Core Plant Plant Health Threat" [1] ;
— le patron de description des organismes vivants et de leurs ressources génétiques de "Ontology for Experimental Scientific Objects Core" (OESO-CORE) ;
— le patron de spécification des connaissances versus observations d'évènements réels de "Crop Planification and Production Process Ontology" (C3PO) [4] ;
— les classes des organes des plantes définie dans la "Plant Ontology" (PO) [6] ;

5. https://gd.eppo.int/
6. https://agroportal.lirmm.fr/ontologies/CD
7. https://lot.linkeddata.es/

— la classification des types de maladies, des types de bioagresseurs et des rôles dans IDOPlant [21] ;
— les stades phénologiques des graphes issus de "BBCH-based Plant Phenological Description Ontology" (PPDO) [17] ;
— la hiérarchie de classes décrivant les taxonomies scientifiques des ontologies NCBITaxon [19] et TAXREF-LD [12] ;
— la classe *fcuo :Crop* de l'ontologie "French Crop Usage Ontology" (FCUO) [8] ainsi que le thésaurus FCU [9] décrivant l'organisation des cultures en France.

Nous avons commencé par construire un premier diagramme à l'aide du language CHOWLK et de l'outil collaboratif draw.io en utilisant les patrons précédents. Ensuite ce diagramme a évolué en fonction des commentaires des experts. Nos 4 experts (Florence Amardeilh, Mathieu Hirschy, Philippe Larignon, Nicolas Sauvion) ont été interviewés séparément pour modifier le diagramme CHOWLK en fonction de leur compréhension du domaine. Ainsi, nous avons construit en 5 itérations le modèle associé à l'ontologie.

Dès les premières itérations, nous avons proposé des définitions pour les classes et les propriétés d'objets. Ces définitions ont été partagées avec nos experts par le biais d'un fichier tabulé partagé sur le Web. Nous demandions aux experts s'ils avaient des définitions issues de sources de référence à nous proposer.

Nos définitions ont été inspirées des sources publiées [10] par des organismes de recherche en agronomie comme : le site web INRAE Ephytia [11], le thesaurus de INRAE [12], les glossaires des livres tel que [18].

3.3 Encodage des ontologies

Une fois le modèle finalisé et les définitions validées, notre ontologie a été implémentée dans le langage OWL en utilisant Protégé (v5.1.0)[13] et son plugin Cellfie. Cellfie [13] permet de transformer le contenu des fichiers tabulés en axiomes pour enrichir l'ontologie. Ainsi, nous avons utilisé le fichier tabulé des définitions pour documenter les classes.

La mise en œuvre de l'ontologie comprenait la déclaration de métadonnées telles que les contributeurs, les dates et la licence, selon [5]. L'implémentation actuelle de BioAGgressor Ontology (BAGO) contient :

— 67 classes, dont 27 sont des classes définies ;
— 52 propriétés objet (object properties) dont 4 sont définies par des chaînes de propriétés.

Ces éléments sont documentés par des propriétés d'annotation SKOS (skos :definition, skos :note, skos :prefLabel, skos :altLabel). Elle réutilise certaines classes de TAXREF-LD, NCBITaxon, FCUO, et PPDO.

3.4 Évaluation et Publication de l'ontologie

Cette ontologie n'est pas encore finalisée au moment où nous écrivons cet article. Nous avons besoin de la peupler pour vérifier la cohérence et valider des règles d'inférences. Un premier test a été réalisé sur une agression de plant de vigne par un agent pathogène.

Pour maintenir cette ontologie et la faire évoluer, nous avons besoin d'obtenir les commentaires de la communauté d'utilisateurs potentiels. Le code OWL de l'ontologie est disponible dans un dépôt git hébergé sur la forge MIA de INRAE [14]. Ainsi les utilisateurs peuvent écrire des commentaires et rendre compte des problèmes rencontrés en déclarant des issues. La documentation du répertoire fournit aussi les adresses e-mail des responsables des ontologies. BAGO est publiée sur le portail Agroportal : `https://agroportal.lirmm.fr/ontologies/BAGO`. L'ontologie BAGO sera publiée sur le Web, avec un identifiant pérenne : `https://opendata.inrae.fr/bag-def`.

4 Présentation du modèle de BAGO

Vu sa construction, BAGO représente le point de vue des experts francophones dans le domaine des maladies des plantes. Nous avons fait le choix dans BAGO de modéliser une situation d'agression d'un bioagresseur sur une plante par une relation N-aire comme présenté dans la figure 1.

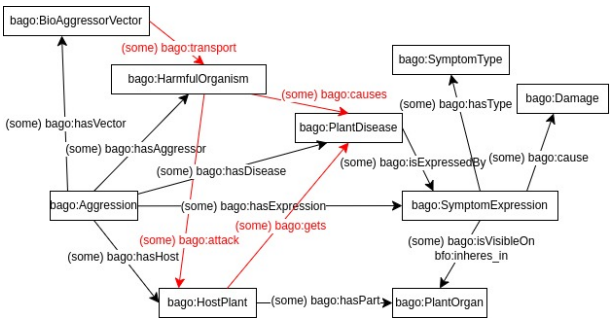


FIGURE 1 – modèle d'une agression

Ainsi, nous pourrons inférer les relations binaires d'interaction entre organismes à partir de cette relation N-aire. Dans la figure 1, les propriétés en rouge sont définies par des chaînes de propriétés (property chains). Nous pourrons ainsi aligner les données issues d'autres modèles comme la base d'épidémio surveillance végétale de l'INRAE.

Pour différencier les connaissances par rapport aux observations, nous avons repris le patron de C3PO spécification / réalisation. Une spécification agrège l'ensemble des connaissances connues sur une agression donnée alors que la réalisation indique uniquement ce qui a été observé en champs. La figure 2 présente le modèle d'une spécification d'agression indiquant les stades de développement où

8. `https://agroportal.lirmm.fr/ontologies/FCUO`
9. `https://agroportal.lirmm.fr/ontologies/CROPUSAGE`
10. La liste complète est décrite dans le readme du répertoire git
11. `https://ephytia.inra.fr/fr/`
12. `https://thesaurus.inrae.fr/thesaurus-inrae/fr/`
13. `https://github.com/protegeproject/cellfie-plugin`
14. `https://forgemia.inra.fr/bsv/bio-agressor-ontology`

la culture est sensible au bioagresseur. Ce modèle indique les stades de développement où le bioagresseur est en situation d'attaquer une culture. Pour ce faire, il faudra que l'ontologie PPDO soit étendue aux organismes zoologiques ou complétée par une autre ontologie.

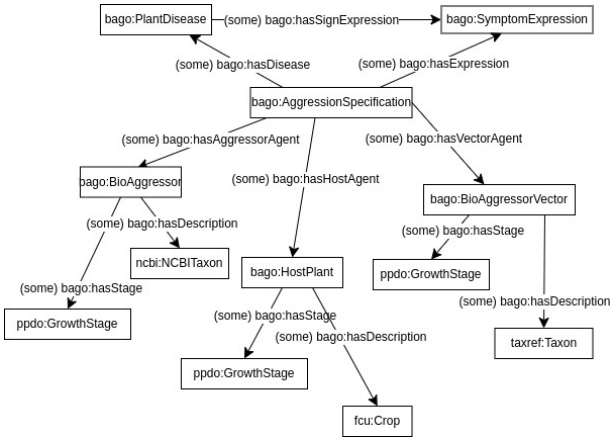


FIGURE 2 – modèle de spécification d'agression

5 Discussion

5.1 Méthode

Concernant la méthodologie de développement utilisée, les définitions en langue naturelle et leurs sources jouent un rôle important lors de la conceptualisation. Par exemple, trois termes proches ont été identifiés pendant la spécifications : symptômes, dégâts et dommages. Dans un premier temps aucune définition n'a fait consensus, jusqu'à ce qu'un des experts propose une définition issue d'une source de référence. L'usage des sources de référence permet de stabiliser les discussions entre experts, qui peuvent ne pas être d'accord entre eux.

Cellfie présente des limites, il ne travaille pas avec des propriétés objets. Ce qui nous a posé problème pour documenter les propriétés.

5.2 Modèles

IDOPLant modélise les rôles par des classes. Nous avons reproduit cette modélisation mais n'avons pas encore compris son utilité autre que documenter les classes d'organismes nuisibles en indiquant leur rôle dans une situation d'agression.

Nous avons choisi de travailler avec deux taxonomies scientifiques : celle de TAXREF-LD qui représente les organismes vivants présents sur le territoire français et NCBITaxon qui a une couverture plus large. Ainsi nous sommes sûrs d'être à jour sur l'évolution des connaissances taxonomiques des organismes vivants. TAXREF-LD ne couvre pas les micro-organismes de type levure ou bactérie. Les plantes cultivées sont aussi décrites par le thésaurus FCU. Les deux taxonomies ne sont pas complètement identiques, et peuvent présenter des formes d'incohérence sur le type de rang taxonomique indiqué pour un taxon. Il existe des alignements entre ces trois ressources sémantiques (TAXREF-LD, NCBITaxon et FCU). Pour le moment nous avons choisi de ne pas utiliser les alignements connus entre ces sources, mais il serait intéressant dans des travaux futurs d'inclure et de faire évoluer ces alignements.

6 Synthèse et Conclusion

L'ontologie BAGO représente l'expertise française sur les bioagresseurs des plantes. Elle modélise une agression par une relation N-aire dans le but de pouvoir ajouter de nouvelles informations, comme les conditions climatiques qui favorisent l'apparition d'une maladie. Pour être compatible avec les modèles qui représentent les interactions entre organismes sous forme de relation binaires, cette information est dupliquée par d'autres propriétés objet.

Dans un future proche, BAGO sera instanciée et enrichie pour représenter les bioagresseurs de la vigne sous la forme d'un graphe de connaissances intégrant une partie des taxons de TAXREF-LD et NCBITaxon. Les identifiants des maladies et des bioagresseurs de la vigne ont été déclarés dans un fichier tabulé et utilisés pour annoter des BSV viticulture de la région Alsace. Comme perspectives nous avons également prévu de publier les alignements de BAGO avec l'ontologie IDOplant qui nous a paru la plus complète.

Remerciements

Ces travaux ont été financés par projet ANR Data to Knowledge in Agronomy and Biodiversity (ANR-18-CE23-0017) et par le Plan de Relance et le Programme d'Investissements d'Avenir «i-Nov» du gouvernement français. Nous tenons à remercier la DIPSO INRAE, Sophie AUBIN et François-Xavier SENNESAL, pour leur aide dans les alignements avec le thésaurus INRAE et la publication des URIs. L'équipe Agroportal nous a aussi aider à publier BAGO.

Références

[1] Oscar Alomar, Assumpció Batlle, Josep Maria Brunetti, Roberto García, Rosa Gil, Toni Granollers, Sara Jiménez, Amparo Laviña, Carme Reverté, Jordi Riudavets, et al. Development and testing of the media monitoring tool med is ys for the monitoring, early identification and reporting of existing and emerging plant health threats. *EFSA Supporting Publications*, 13(12) :1118E, 2016.

[2] Aarón Ayllón-Benitez, José Antonio Bernabé-Diaz, Paola Espinoza-Arias, Iker Esnaola-Gonzalez, Delphine SA Beeckman, Bonnie McCaig, Kristin Hanzlik, Toon Cools, Carlos Castro Iragorri, and Nicolás Palacios. Eppo ontology : a semantic-driven approach for plant and pest codes representation. *Frontiers in Artificial Intelligence*, 6 :1131667, 2023.

[3] Joaquín Cañadas, Isabel M del Águila, and José Palma. Development of a web tool for action thre-

shold evaluation in table grape pest management. *Precision agriculture*, 18 :974–996, 2017.

[4] Baptiste Darnala, Florence Amardeilh, Catherine Roussey, Konstantin Todorov, and Clement Jonquet. C3PO : a crop planning and production process ontology and knowledge graph. *Frontiers in Artificial Intelligence*, 6 :1187090, October 2023.

[5] Daniel Garijo and M. Poveda Villalon. A checklist for complete vocabulary metadata. Technical report, WIDOCO, April 2017.

[6] Pankaj Jaiswal, Shulamit Avraham, Katica Ilic, Elizabeth A Kellogg, Susan McCouch, Anuradha Pujar, Leonore Reiser, Seung Y Rhee, Martin M Sachs, Mary Schaeffer, et al. Plant ontology (po) : a controlled vocabulary of plant structures and growth stages. *Comparative and functional genomics*, 6(7-8) :388–397, 2005.

[7] Clement Jonquet, Anne Toulet, Elizabeth Arnaud, Sophie Aubin, Esther Dzalé Yeumo, Vincent Emonet, John Graybeal, Marie-Angélique Laporte, Mark A. Musen, Valeria Pesce, and Pierre Larmande. AgroPortal : a vocabulary and ontology repository for agronomy. *Computers and Electronics in Agriculture*, 144 :126–143, January 2018.

[8] Javier Lacasta, F. Javier Lopez-Pellicer, Borja Espejo-García, Javier Nogueras-Iso, and F. Javier Zarazaga-Soria. Agricultural recommendation system for crop protection. *Computers and Electronics in Agriculture*, 152 :82–89, 2018.

[9] Katty Lagos-Ortiz, José Medina-Moreira, César Morán-Castro, Carlos Campuzano, and Rafael Valencia-García. An ontology-based decision support system for insect pest control in crops. In *International Conference on Technologies and Innovation*, pages 3–14. Springer, 2018.

[10] Katty Lagos-Ortiz, María del Pilar Salas-Zárate, Mario Andrés Paredes-Valverde, José Antonio García-Díaz, and Rafael Valencia-Garcia. Agrient : A knowledge-based web platform for managing insect pests of field crops. *Applied Sciences*, 10(3) :1040, 2020.

[11] Michele Magrane and UniProt Consortium. Uniprot knowledgebase : a hub of integrated protein data. *Database*, 2011 :bar009, 2011.

[12] Franck Michel, Olivier Gargominy, Sandrine Tercerie, and Catherine Faron Zucker. A Model to Represent Nomenclatural and Taxonomic Information as Linked Data. Application to the French Taxonomic Register, TAXREF. In *ISWC 2017 Workshop on Semantics for Biodiversity (S4Biodiv 2017)*, volume CEUR Vol. 1933, pages 1–12, Vienna, Austria, October 2017.

[13] Mark A. Musen. The protégé project : a look back and a look forward. *AI Matters*, 1(4) :4–12, June 2015.

[14] María Poveda-Villalón. A reuse-based lightweight method for developing linked data ontologies and vocabularies. In Elena Simperl, Philipp Cimiano, Axel Polleres, Oscar Corcho, and Valentina Presutti, editors, *The Semantic Web : Research and Applications*, pages 833–837, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[15] Miguel Ángel Rodríguez-García and Francisco García-Sánchez. Croppesto : An ontology model for identifying and managing plant pests and diseases. In Rafael Valencia-García, Gema Alcaraz-Marmol, Javier Del Cioppo-Morstadt, Néstor Vera-Lucio, and Martha Bucaram-Leverone, editors, *Technologies and Innovation*, pages 18–29, Cham, 2020. Springer International Publishing.

[16] Miguel Ángel Rodríguez-García, Francisco García-Sánchez, and Rafael Valencia-García. Knowledge-based system for crop pests and diseases recognition. *Electronics*, 10(8), 2021.

[17] Catherine Roussey, Xavier Delpuech, Florence Amardeilh, Stephan Bernard, and Clement Jonquet. Semantic Description of Plant Phenological Development Stages, starting with Grapevine. In Emmanouel Garoufallou and María-Antonia Ovalle-Perandones, editors, *14th International Conference on Metadata and Semantics Research (MTSR 2020)*, volume 1355 of *Metadata and Semantic Research. MTSR 2020*, pages 257–268, Madrid, Spain, December 2020. Springer International Publishing. The final authenticated version is available online at https ://doi.org/10.1007/978-3-030-71903-6_25.

[18] Nicolas Sauvion, Paul-André Calatayud, Denis Thiery, and Frederic Marion-Poll. *Interactions insectes-plantes*. Editions Quae, IRD, 2013.

[19] Eric W Sayers, Jeffrey Beck, Evan E Bolton, Devon Bourexis, James R Brister, Kathi Canese, Donald C Comeau, Kathryn Funk, Sunghwan Kim, William Klimke, et al. Database resources of the national center for biotechnology information. *Nucleic acids research*, 49(D1) :D10, 2021.

[20] Mari Carmen Suárez-Figueroa, Asunción Gómez-Pérez, and Mariano Fernandez-Lopez. The neon methodology framework : A scenario-based methodology for ontology development. *Applied ontology*, 10(2) :107–145, 2015.

[21] Ramona Walls, Barry Smith, Elser Justin, Goldfain Albert, W Stevenson Dennis, and Pankaj Jaiswal. A plant disease extension of the infectious disease ontology. In *Proceedings of the International Conference on Biomedical Ontology (ICBO-2012)*, Graz, Austria, 2012.

Variété d'objets physiques

G. Kassel

Laboratoire MIS, Université de Picardie Jules Verne
33 rue Saint-Leu, 80039 Amiens Cedex 1

Gilles.kassel@u-picardie.fr

Résumé

Dans cet article, nous poursuivons l'étude d'un type d'ontologie que nous avons récemment défini et qualifié d'« épistémique ». Une ontologie épistémique est un système de catégories représentant indirectement le monde : les catégories correspondent à des objets généraux mentaux permettant à un sujet de se référer à des entités du monde. Nous complétons une ontologie épistémique fondatrice en focalisant notre étude sur la notion d'objet physique pensé, ce qui nous conduit à analyser trois variétés de tels objets : des artefacts techniques, des objets physiques spatio-temporalisés et des objets physiques fictifs. Nous en profitons pour comparer nos traitements de ces espèces d'objets avec ceux proposés dans BFO et DOLCE.

Mots-clés

Ontologie appliquée, métaphysique, objet de pensée, objet physique, artefact, fiction, BFO, DOLCE

Abstract

In this article, we continue our study of a type of ontology that we have recently defined and described as "epistemic". An epistemic ontology is a system of categories indirectly representing the world: categories correspond to mentally general objects allowing a subject to refer to entities in the world. We complete a foundational epistemic ontology by focusing our study on the notion of a thought physical object, which leads us to analyze three varieties of such objects: technical artefacts, spatio-temporalized physical objects and fictitious physical objects. We take this opportunity to compare our treatment of these types of objects with that proposed in BFO and DOLCE.

Keywords

Applied ontology, metaphysics, object of thought, physical object, artifact, fiction, BFO, DOLCE

1 Introduction

Récemment, nous avons défini un nouveau type d'ontologie qualifié d'« épistémique » et prôné son utilisation en Ontologie appliquée, et donc en Ingénierie des Connaissances [25]. À l'instar des ontologies couramment développées en Ontologie appliquée [4], une ontologie épistémique est un système de catégories d'objets, structuré principalement au moyen de relations de généralisation. À la différence des ontologies courantes telles BFO [30] et DOLCE [3], les catégories correspondent à des objets généraux mentaux référant au monde et ne représentent donc qu'indirectement le monde. Il en est de même des instances, correspondant à des objets singuliers mentalement pensés, représentant des entités individuelles (des particuliers) du monde. Une ontologie épistémique est de fait une espèce d'ontologie conceptuelle. À l'origine de la notion d'ontologie *épistémique* sont les travaux menés au tournant du 20ème siècle au sein de l'école de Franz Brentano [6], à la frontière de la psychologie et de la métaphysique. Au sein de cette école, nous nous fondons plus particulièrement sur les travaux de Kazimir Twardowski [41,42] et sa théorie de la représentation. Dans une première partie de l'article, nous rappelons le cadre métaphysique nous servant de référence (§ 2) et notre notion d'ontologie épistémique (§ 3).

Une telle espèce d'ontologie, réintégrant le sujet et son monde mental au sein de nos théories métaphysiques, invite à (ré)-analyser la frontière entre entités physiques et entités mentales. C'est à ce travail que l'auteur s'est livré ces dernières années en accordant la priorité aux entités qualifiées d'« occurrentes », processus et événements, pour aboutir à positionner les événements dans la sphère mentale [21,22,23]. Dans cet article, nous choisissons d'analyser plutôt l'objet physique, non pas tel qu'il est en soi, mais tel que nous le pensons, tel que nous le connaissons. Ce changement de perspective « objet pensé *vs* objet en soi » ouvre le champ d'étude à une variété d'objets physiques que nous analysons en seconde partie de l'article : les objets physiques artéfactuels (§ 4), les objets physiques spatio-temporalisés (§ 5) et les objets physiques fictifs (§ 6). Pour chacune de ces variétés d'objets, nous comparons nos traitements à ceux de BFO et de DOLCE.

2 Notre cadre métaphysique de référence

Comme point de départ, nous considérons une bipartition des entités mondaines en *physiques* et *mentales*. Historiquement, cette dichotomie a constitué le cadre majoritaire de pensée des philosophes ontologues depuis René Descartes (1596-1650), jusqu'à ce que Gottlob Frege (1848-1925) introduise des *objets abstraits* platoniciens, dans un premier temps pour rendre compte du caractère apriorique et objectif des vérités des mathématiques, puis par la suite pour caractériser la nature des *pensées* (Ce fait historique est documenté par José Falguera et coll. [13, § 2 *Historical Remarks*]).

Partant donc de cette distinction « physique *vs* mental », nous choisissons, sur un plan méthodologique, d'investiguer en priorité le domaine des entités mentales. Ce faisant, nous nous démarquons de la tendance majoritaire en métaphysique contemporaine consistant à donner la priorité au physique. La raison de cette priorité accordée au physique est compréhensible : les entités physiques sont créditées d'une « véritable » existence, là où, au mieux, une existence « impropre », « diminuée », est accordée aux entités mentales (du fait qu'elles dépendent de la pensée d'êtres humains). Deux raisons guident notre choix. D'une part, la conviction que, faute d'une reconnaissance du mental, des entités sont improprement positionnées dans le physique. C'est un point de vue que nous défendons notamment concernant les *événements* [22,23]. D'autre part, nous tenons compte d'un des objectifs visés en Intelligence Artificielle qui est de représenter nos *connaissances* du monde[1]. Nous visons dès lors à définir les fondements ontologiques de nos connaissances et il est évident que ces fondements sont à rechercher du côté du mental.

Pour développer notre cadre métaphysique, nous nous référons à des travaux, contemporains de ceux de Frege, réalisés par Franz Brentano (1838-1917) et ses disciples, notamment Kazimir Twardowski (1866-1938), relevant de la psychologie, de la logique et de la philosophie. Brentano, dans sa *Psychologie descriptive* [6], désirant fonder scientifiquement la psychologie, caractérise les phénomènes (actes et attitudes) mentaux comme étant dirigés vers un *objet immanent* leur étant propre. Pour Brentano, cette intentionnalité définit l'essence des phénomènes psychiques et les distingue des phénomènes physiques. Il s'avère que l'*objet* en question est un *contenu* correspondant à une combinaison de propriétés référant à un objet transcendant l'acte [37][2]. Twardowski continuera à développer la doctrine de l'intentionnalité de son maître. Dans son [41] *Sur la théorie du contenu et de l'objet des représentations*, Twardowski apporte toutefois un amendement important à la théorie brentanienne de la représentation en distinguant un objet immanent et un contenu immanent à l'acte, posant ainsi un modèle à 4 termes de l'acte psychique : acte / contenu immanent / objet immanent / objet transcendant de référence. Plus tard, dans sa conférence de 1912 [42], Twardowski ira plus loin en accordant une existence à la représentation comme produit continuant de l'acte ayant l'objet et le contenu comme matériaux.

La motivation principale de Twardowski en 1894 est de fournir une explication aux représentations « anobjectuelles » de Bernard Bolzano (1781-1848) ne référant à aucune entité existante, ce qui semble être le cas pour les représentations exprimées par les expressions « la montagne d'or », « le carré rond » ou « l'actuel roi de France ». Selon Twardowski, ces représentations reviennent à penser à des objets auxquels des propriétés sont attribuées, comme le fait d'*être une montagne* et d'*être constitué d'or*, etc., et, même si ces objets n'existent pas réellement, ils existent mentalement : penser à de tels objets, se les représenter, leur confère une existence mentale[3].

Précisons la nature et le rôle joué par l'objet immanent dans l'acte de pensée à un objet. D'après ce que nous venons de voir avec les exemples ci-dessus, l'objet immanent est un objet pensé au sens où cet objet, qui devient représenté, donne au sujet la capacité à penser à un objet doté de propriétés. Un sujet peut ainsi penser au Mont Blanc en lui attribuant telle ou telle propriété dépendant de ses connaissances de cette montagne, ces connaissances pouvant être alimentées à l'occasion par une perception directe du pic montagneux. Mais l'acte de pensée ne s'arrête pas là. Pour le sujet, une différence existe entre 'la montagne d'or' et 'le Mont Blanc' : le premier objet n'« existe pas réellement », au contraire du second (à supposer bien sûr que le sujet le conçoive comme tel). Intervient ici, dans la pensée, un acte psychique que Brentano appelle le « jugement existentiel ». Selon la théorie brentanienne du jugement, reprise par Twardowski, un tel acte consiste à reconnaître ou dénier à un objet une existence réelle[4]. Twardowski apporte l'amendement suivant : même en l'absence de référence, l'objet pensé existe dans tous les cas (dans toute représentation). Prenons l'exemple d'un cueilleur d'un champignon vénéneux qui, l'ayant placé dans son réfrigérateur, continue à penser la nuit à ce champignon alors que quelqu'un de son entourage l'aura sorti du réfrigérateur et détruit, le champignon n'existant plus physiquement[5]. Dans une telle situation, l'objet pensé mental permet au cueilleur de continuer à penser au champignon, qui n'existe plus physiquement.

La Fig. 1 résume les entités impliquées dans un acte de pensée à un objet – l'*objet pensé*. La conscience du sujet se *dirige vers* l'objet pensé immanent (rel *IMM*). L'objet pensé est caractérisé par un ensemble de propriétés – le *contenu* de la représentation. L'objet pensé peut être conçu comme *représentant* une entité (rel *REPR*) à laquelle il se *réfère* (rel *REF*). Cette entité de référence peut être physique ou mentale. Dans le cas où l'objet pensé est conçu comme ne référant à aucune entité, la visée s'arrête à l'objet pensé (les flèches en pointillés dans notre diagramme signifient que les entités correspondantes n'ont pas besoin d'exister). En Fig. 1 apparaissent explicitement 3 des 4

[1] On notera ici une opposition avec le principe de « réalisme ontologique » affiché dans BFO [30] de représenter, non pas nos connaissances du monde, mais le monde selon nos meilleures connaissances.

[2] En tout cas dans la période scientifique de Brentano habituellement qualifiée de « réiste », à partir des années 1900. Selon Arkadiusz Chrudzimski [9, § 4 *The representational theory sensu stricto*], Brentano, dans son habilitation portant sur *La Psychologie d'Aristote* (1867), développa une théorie de la représentation distinguant un objet immanent « analogue » aux objets réels perçus et auquel des propriétés représentant celles des objets réels sont attribuées. Il s'agit là de la théorie que développera Twardowski.

[3] Twardowski [41, § 5, p. 109] : « La confusion commise par les défenseurs des représentations sans objet consiste en ceci qu'ils ont tenu la non-existence [réelle] d'un objet de représentation pour un non-devenir-représenté. Or, toutefois, par chaque représentation, un objet devient représenté, qu'il existe ou non [réellement], de même que chaque nom nomme un objet, sans avoir égard au fait que celui-ci existe ou non [réellement]. »

[4] Cette même conception du jugement sera adoptée par Alexius Meinong dans sa théorie de l'objet [28]. En revanche, contrairement à Twardowski, Meinong dé-pyschologisera son *objet pur* pour en faire une entité apatride en termes de disciplines scientifiques susceptibles de l'accueillir [15].

[5] Nous reprenons là l'exemple inventé par Chrudzimski [*op. cit.*] pour illustrer ce fait psychologique.

termes du modèle twardowskien de l'acte de pensée : l'objet de la représentation, le contenu ainsi que les éventuelles entités de référence. Il convient de considérer que la figure représente globalement le 4ème terme, à savoir l'acte de pensée lui-même. À noter également que les mécanismes de représentation et de référence sont identiques dans le cas où le sujet pense à un objet immanent à son esprit, par exemple lorsque ses représentations deviennent à leur tour objets de (méta)-représentations.

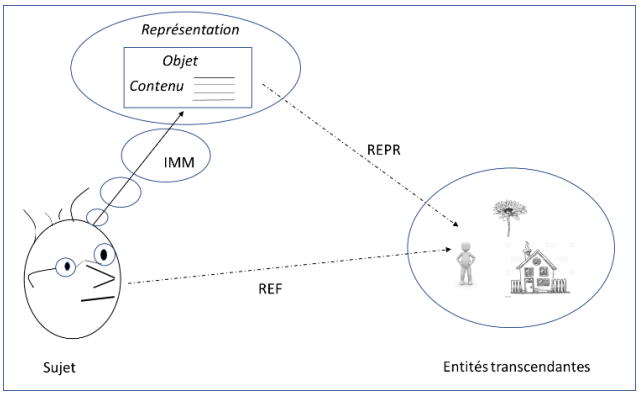


FIGURE 1 – Entités impliquées lors d'une pensée d'un sujet à un objet

Le jugement d'existence d'un objet pensé, en ce qu'il consiste en une mise en correspondance avec d'autres entités, revient à ancrer l'objet dans le système de croyances du sujet. Par le jugement, l'objet devient, pour le sujet, connaissance de quelque chose. Concernant ce statut épistémique de l'objet, apportons quelques précisions.

(i) Dans leurs écrits, Brentano et Twardowski indiquent que, lorsque l'objet pensé représente une entité référente, il la représente selon un certain *aspect* ou *profil* (*Abschattung*). Il convient donc de considérer qu'un sujet entretient généralement plusieurs objets pensés co-référentiels pour rendre compte de ces différents aspects. On trouve cette même conception dans les travaux contemporains en philosophie de l'esprit et du langage portant sur les *dossiers mentaux* [29,32]. Dans ce courant, toutefois, la notion frégéenne platonicienne de *mode de présentation* sert de référence alors que, par « aspect », nous entendons un mode de présentation mental.

(ii) Du fait de relever du mental et d'être une croyance sur le monde, une représentation peut s'avérer erronée. L'origine peut en être une hallucination temporaire ou un engagement vis-à-vis d'une idéologie ou d'une théorie scientifique erronée. Un exemple célèbre est celui de l'astronome français Le Verrier (1811-1877) qui, ayant théorisé l'existence d'une planète nommée Vulcain pour expliquer des perturbations de la trajectoire de Mercure, verra sa théorie infirmée.

(iii) Mentionnons également une possible opacité épistémique de la référence. Durant plusieurs années, les lecteurs et critiques littéraires ont pris Romain Gary et Émile Ajar comme deux écrivains distincts, jusqu'à ce que la mystification entretenue par les deux pseudonymes soit éventée[6]. Ce faisant, des critiques littéraires ont attribué des propriétés contradictoires à cet auteur, voyant dans les écrits post prix Goncourt de Gary un écrivain « has-been » tout en encensant Ajar, qui remportera également le prix Goncourt.

Par la suite, nous revenons au cas général où un sujet, en connaissance de cause, entretient plusieurs objets pensés co-référentiels. Ceci n'empêche toutefois pas que des propriétés contradictoires soient attribuées, toujours en connaissance de cause. Prenons l'exemple cité par Saul Krikpe [26] de Ignacy Paderewski (1860-1941) qui a mené de front deux carrières professionnelles, d'homme politique et de pianiste. On peut s'attendre à ce que des qualités distinctes, voire contradictoires, soient attribuées, d'une part à 'Paderewski le ministre', d'autre part à 'Paderewski le pianiste', sans pour autant que la rationalité du sujet pensant soit questionnée.

Pour clore la présentation de notre cadre métaphysique de référence, nous souhaitons évoquer une critique principale adressée à Twardowski (nous y ferons référence dans la suite du texte), concernant l'existence de deux objets intentionnels, à savoir l'objet pensé immanent et l'objet transcendant de référence. Dès la parution du texte de Twardowski en 1894, Edmund Husserl (1859-1938) contestera violemment (notamment) le fait qu'il puisse exister deux objets intentionnels [17, p. 282-283].

> C'est le même Berlin que celui que je me représente, qui existe aussi, et c'est le même qui n'existerait plus si un châtiment éclatait comme à Sodome et Gomorrhe. C'est le même Centaure Chiron que celui dont je parle à présent et que par là je me représente. Et d'une manière analogue dans chaque cas où la représentation est univalente.

En métaphysique contemporaine, la position dominante est pro-husserlienne. Une stratégie d'analyse de la pensée d'un sujet à un objet (quelconque), connue sous le vocable « adverbialiste », consiste à masquer l'existence de l'objet immanent de pensée en le faisant participer d'une propriété complexe de pensée attribuée au sujet, par exemple : Husserl 'pense de façon berlinoise' (comme on pourrait dire qu'il 'marche de façon chaotique'). Dans cet article, nous ne chercherons pas à dégager les défauts d'une telle stratégie[7]. Tout au contraire, nous tâcherons de continuer à justifier l'existence de l'objet pensé en nous fondant principalement sur des données psychologiques.

3 Notion et esquisse d'une ontologie épistémique

L'objet pensé, tel que le conçoit Twardowski, correspond à l'*ens rationis* scholastique – l'être de raison, par opposition à l'être réel – à ceci près (et la différence est de taille !) que son domaine est étendu aux objets portant des déterminations

[6] Nous empruntons cet exemple à François Recanati, qui l'utilise abondamment dans ses cours sur la référence et les dossiers mentaux au collège de France. https://www.college-de-france.fr/fr/agenda/cours/dossiers-mentaux.

[7] Pour des arguments convaincants soulignant le prix fort à payer d'une telle stratégie pour rendre compte d'une architecture cognitive cohérente, nous renvoyons le lecteur aux analyses de Jacques Dubucs et Wioletta Miśkiewicz [11, § 1.2] et de Chrudzimski [10, § 9 *Problems with the adverbial theory*].

contradictoires pour couvrir finalement le domaine du *représentable* et donc du *pensable*. Cet objet vient en plusieurs espèces « formelles » (on parle ici de catégories de représentations). Notamment une distinction est établie entre des objets *singuliers* et des objets *généraux* : l'objet *singulier* représente une entité individuelle ; l'objet *général*, l'analogue mental d'une idée générale platonicienne, partage avec l'objet singulier le fait d'être une unité et non une pluralité. Sur un plan métaphysique, l'objet général, par exemple *le triangle*, *le lion*, *l'hépatite* ou *la récession économique*, possède des déterminations communes à une pluralité d'objets singuliers qu'il subordonne. Nous engageant vis-à-vis de cette théorie de l'objet pensé, nous formulons la thèse (psychologique) que tout sujet dispose d'un système de catégories correspondant à des objets pensés généraux, autrement dit des catégories représentant non pas le monde directement mais les connaissances du sujet sur le monde – nous avons baptisé en ce sens ce système de catégories « ontologie épistémique » [25].

La Fig. 2 présente une proposition d'ontologie épistémique fondatrice [25]. Celle-ci illustre l'engagement pris, rappelé en § 2, de l'existence d'une bipartition des entités mondaines en physiques et mentales. À noter que la catégorie `Physique` (resp. `Mental`) représente la connaissance d'un sujet portant sur les objets physiques (resp. mentaux) en général. Précisons que les catégories, pour être privées (car présentes dans la tête de sujets), peuvent jouir d'une dimension sociale en étant « partagées » entre plusieurs sujets. Nous aurons, par la suite, l'occasion d'élucider sur un plan ontologique cette dimension sociale permettant à des sujets de penser qu'ils/elles pensent à « la même chose ». À ce stade, nous nous contentons d'indiquer qu'il est possible de considérer des catégories de sens commun ou au contraire relevant de théories scientifiques[8].

Sur un plan métaphysique, et pour s'ancrer dans le physique, nous sommes en présence de deux types d'entités, d'une part l'objet physique *en soi* existant indépendamment de nos pensées, d'autre part l'objet physique *pensé* correspondant à notre connaissance de l'objet physique, à la manière dont nous nous représentons le monde physique.

Le principe d'un réalisme physique, retenu par l'ensemble des ontologies fondatrices développées en Ontologie appliquée, constitue notre engagement de départ. Toutefois, lorsqu'il s'agit de préciser l'ameublement du monde physique que l'on retient, des divergences importantes se font jour. Un exemple de divergence concerne le statut à accorder aux universaux aristotéliciens – ces entités censées exister à l'identique dans des objets particuliers – postulés pour rendre compte de la ressemblance entre objets physiques. Dans cet article, nous choisissons de laisser de côté cette question, les considérations qui suivront pouvant être acceptées aussi bien par les défenseurs des universaux physiques que par les détracteurs de ces entités (dont l'auteur fait partie). Nous nous focalisons sur les particuliers physiques. Plusieurs théories en rendent compte, celle de la substance aristotélicienne, celle plus récente des tropes, par ailleurs celle des niveaux ontologiques consistant à considérer qu'un objet particulier physique est constitué (à l'image de poupées russes) d'objets physiques existant à différents niveaux [31,27]. Concernant cette dernière théorie, un consensus existe sur un ensemble de niveaux : *atome-molécule-cellule-organisme*, la question se posant sur la façon de les compléter en positionnant les artefacts, les entités mentales, les organisations.

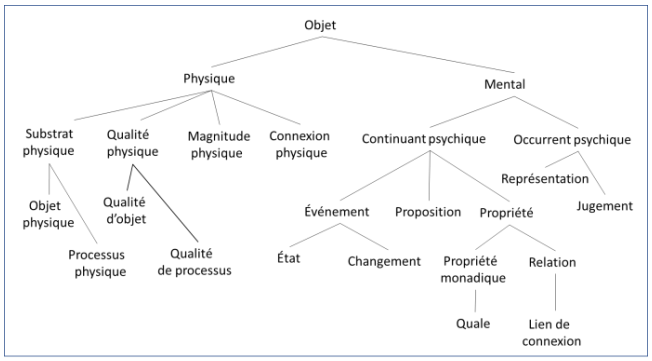


FIGURE 2 – Esquisse d'une ontologie épistémique fondatrice

Dans la suite de l'article, nous commençons par nous focaliser sur les objets physiques artéfactuels et défendons la thèse selon laquelle l'artefact est une espèce d'objet pensé et non d'objet physique, en contradiction avec les traitements des artefacts dans DOLCE et BFO (§ 4). En prolongement de cette thèse, nous mettons en avant deux autres espèces d'objets physiques pensés : d'une part, des objets physiques ayant existé dans le passé ou étant susceptible d'exister dans le futur (§ 5), d'autre part des objets auxquels nous attribuons des propriétés physiques alors qu'ils ne jouissent pas d'une existence physique réelle (§ 6).

4 Objets physiques artéfactuels

Nous débutons notre analyse de la variété des objets physiques par les artefacts, compte tenu du rôle stratégique qu'ils jouent pour positionner la frontière entre le physique et le mental (et le social). Dans la littérature philosophique, deux propriétés essentielles caractérisent les artefacts techniques, à savoir le fait d'être des entités *fonctionnelles* et d'être *intentionnellement produites* [20]. Suivant John Searle [33], nous considérons que la notion d'artefact repose sur l'attribution d'une fonction à une entité, une telle attribution correspondant à un premier pas dans la construction de la réalité sociale. De fait, deux catégories d'entités sociales sont à distinguer[9]. En premier lieu, on trouve des entités sociales *concrètes* (ex : une bouteille, un presse-papier, un tournevis) : ces entités jouent le rôle du *Y* dans la règle constitutive « *X* compte pour *Y* dans le contexte *C* » ; le *X* est l'entité concrète à laquelle une fonction est attribuée, constituant ainsi l'entité *Y*. Par exemple, un galet peut se voir attribuer la fonction de presser du papier et constituer ainsi un

[8] Dans [25], nous explorons la possibilité de faire cohabiter des objets de sens commun et théoriques, de même que des objets pensés par différents sujets individuels. Dans ce texte, nous n'abordons pas cet aspect de cohabitation pour nous focaliser sur des connaissances de sens commun.

[9] Nous nous référons ici à la distinction établie par Amie Thomasson [38].

presse-papier. En second lieu, on trouve des entités sociales *abstraites* (ex : une loi, une monnaie, un syndicat) pour lesquelles la règle précédente ne s'applique pas, faute de pouvoir exhiber un *X* sur lequel surviendrait (directement) un fait social. Suivant par ailleurs Amie Thomasson [40], nous considérons qu'un artefact est intentionnellement produit et que « l'intention de produire un artefact de type K doit impliquer un concept substantiel (et substantiellement correct) de ce qu'est un K, incluant une compréhension des types de propriétés étant K-pertinentes et une intention de réaliser plusieurs de ces propriétés dans l'objet créé » [*ibid.*, p. 59].

Compte tenu de notre propos dans cet article, nous nous focalisons sur les entités sociales concrètes. Nous visons à rendre compte de la façon dont des propriétés telles des fonctions et des intentions de création surviennent sur des objets physiques, ce qui est le cas pour les artefacts techniques. Plus particulièrement, en guise d'objets physiques nous considérons les objets de notre quotidien avec lesquels nous interagissons et que nous utilisons pour réaliser des actions – ou « objets-Spelke » [36][10].

Pour illustrer les engagements ontologiques que nous nous apprêtons à prendre vis-à-vis des artefacts, considérons comme exemples un bistouri et un scalpel. Bien que le même terme puisse être utilisé (suivant les lieux) pour référer à ces artefacts, sachant qu'il s'agit d'objets physiques strictement identiques, la distinction conceptuelle que nous retenons est fonctionnelle et dépend du contexte d'utilisation : les bistouris servent à inciser la peau des vivants tandis que les scalpels servent à disséquer les morts.

Sur un plan ontologique, précisons quelles entités existent dans la situation, par exemple, où un soignant utilise un bistouri. La théorie des objets pensés mentaux nous conduit à distinguer deux entités : d'une part, l'objet physique et, d'autre part, l'objet pensé correspondant à la conceptualisation de l'objet physique par le soignant. L'objet physique et l'objet pensé ne « portent » pas des propriétés en un même sens : les propriétés de l'objet pensé sont attribuées par un sujet à l'occasion d'un acte psychique de représentation. En l'occurrence, un objet pensé singulier $Bistouri_{\#i}$ est pensé par le soignant comme étant à la fois un objet physique et un artefact (il cumule les propriétés) et cet objet représente un objet physique. Si le soignant devait utiliser dans un contexte différent le même objet comme scalpel, nous aurions un nouvel objet pensé singulier $Scalpel_{\#j}$ représentant le même objet physique.

Dans une ontologie épistémique, sachant qu'une catégorie correspond à un objet pensé général, nous trouvons les catégories `Bistouri` et `Scalpel` comme plus spécifiques de la catégorie `Objet Physique` (cf. Fig. 3). Rappelons que cette dernière représente notre connaissance en général des objets physiques.

`Objet Physique`

`Objet-Spelke` *(Cohésion, solidité, continuité, contact)*

`Artefact technique` *(Fonction, création intentionnelle)*

`Bistouri` *(Incision, vivant)*

`Scalpel` *(Dissection, mort)*

FIGURE 3 – Ontologie des artefacts techniques

La situation du soignant concevant un même objet physique tour à tour comme un bistouri et un scalpel est illustrée en Fig. 4. Les objets singuliers $Bistouri_{\#i}$ et $Scalpel_{\#j}$ sont des instances des objets généraux *Bistouri* et *Scalpel*.

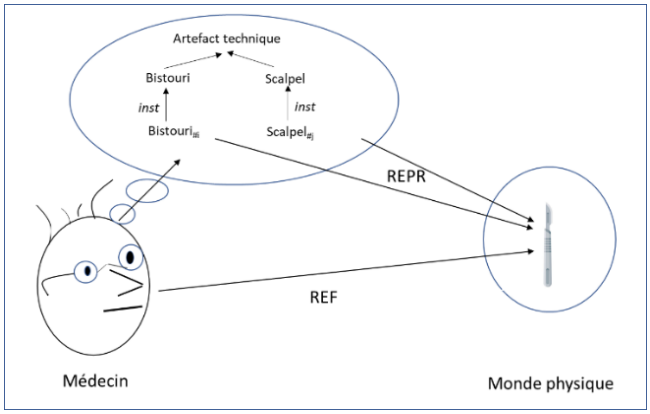


FIGURE 4 – Les objets singuliers $Bistouri_{\#i}$ et $Scalpel_{\#j}$ représentent un même objet physique.

Un premier point important à noter concernant nos engagements ontologiques est que nous ne distinguons pas, du côté du monde physique, de sous-domaine des artefacts. En effet, nous considérons que les artefacts se distinguent des non-artefacts, non par des propriétés physiques mais par des propriétés purement mentales (la fonction et l'intention de création de l'artefact). La catégorie `Artefact Technique` rend compte de cette distinction au niveau mental. Un second point important est le fait de considérer que l'existence d'artefacts, autrement dit d'objets pensés comme étant des artefacts, ne dépend pas de l'existence concomitante d'objets physiques de référence (à l'instar du cueilleur d'un champignon vénéneux, continuant à penser à ce champignon alors que ce dernier a été détruit)[11]. En conséquence, et pour se référer à la théorie des niveaux évoquée *supra*, l'artefact ne vient pas compléter la chaîne des niveaux *atome-molécule-cellule-organisme* selon une même relation de *constitution*. La relation existante entre des objets mentaux conçus comme des artefacts

[10] Un consensus existe en psychologie cognitive sur la nature de ces objets, même si différentes caractérisations ont été données dans la littérature. À titre d'exemple, nous mentionnons celle donnée par Roberto Casati [7]. Selon Casati [*ibid.*, p. 574], 4 principes déterminent les Spelke-objets : « 1. Cohesion. Objects are connected masses of stuff that move as a whole (…); 2. Solidity. Objects are not easily permeable by other objects (…); 3. Continuity. Objects move in continuous paths (…); 4. Contact. Objects move through contact (…) ».

[11] Sur ce point, nous nous distinguons de l'ontologie des objets culturels développée par Roman Ingarden [18] mais aussi des engagements ontologiques de Thomasson, ces deux auteurs supposant une relation de dépendance rigide entre l'artefact et sa réalisation physique [39]. Thomasson en tire prétexte pour identifier un artefact à une entité hybride dépendant à la fois du mental et du physique.

et des entités matérielles de cette chaîne est la relation de *représentation*.

Cette question de savoir ce qui relève du physique ou du mental dans le cas des artefacts se trouve au cœur d'un échange entre Smith et Searle [34] concernant la théorie de la réalité sociale de Searle. Searle précise ainsi sa conception des objets sociaux en général [*ibid.*, p. 302] :

> The notion of a social object seems at best misleading because it suggests that there is a class of social objects as distinct from a class of non-social objects. But if you suppose that there are two classes of objects, social and non-social, you immediately get contradictions of the following sort: In my hand, I hold an object. This one and the same object is both a piece of paper and a dollar bill. As a piece of paper it is a non-social object; as a dollar bill, it is a social object. So, which is it? The answer, of course, is that it is both. But to say that is to say that we do not have a separate class of objects that we identify with the notion of social object. Rather, what we have to say is that something is a social object only under certain descriptions and not others (…)

Les engagements que nous venons de prendre sont proches de la position exprimée ici par Searle et en même temps différents dans la mesure où nous rendons compte du social par un objet mental complétant l'objet physique. Effectivement, dans notre main, nous ne tenons qu'un seul objet – l'objet physique – que nous considérons être à la fois un morceau de papier et un billet d'un dollar. Qui plus est, effectivement, dans le monde physique, il n'y a pas lieu de distinguer une classe d'objets sociaux et une classe d'objets non-sociaux. En revanche, cette distinction vaut du côté des objets pensés car ce qui distingue conceptuellement l'objet physique stricto sensu de l'artefact est l'attribution de propriétés non physiques. La notion d'artefact devient une sous-classe d'objets pensés. L'attribution d'une fonction à un objet physique ne modifie pas ce dernier. En revanche, du côté des objets pensés, quelque chose devient un objet social sous une certaine description

En Ontologie appliquée, si nous prenons comme références le traitement des artefacts par DOLCE et BFO, nous constatons que la dimension non-physique et purement mentale de la fonction n'est pas prise en compte, au point d'identifier l'artefact, comme nous le proposons, à une entité mentale. Dans DOLCE, l'artefact est identifié à un objet physique auquel un agent attribue une capacité spécifique[12]. Pour autant, l'artefact demeure un objet physique, la catégorie `Physical Artefact(_DOLCE)` étant subsumée par la catégorie `Physical Object(_DOLCE)`. L'artefact physique $\texttt{Dollar Bill}_{\#i}\texttt{(\_DOLCE)}$ est distinct de l'objet physique $\texttt{Piece Of Paper}_{\#j}\texttt{(\_DOLCE)}$ dont il est constitué et avec lequel il est co-localisé. Dans BFO, nous notons un choix délibéré de dénier la dimension sociale de l'artefact et de la fonction[13]. Notre point de vue est que ces analyses conduisent à positionner dans la strate physique des entités relevant de nos connaissances, faute d'avoir sérieusement envisagé un cadre métaphysique ménageant une place pour les entités mentales.

5 Objets physiques spatio-temporalisés

Dans cette section, nous nous intéressons à des objets physiques que nous pensons comme spatio-temporalisés. De telles expériences de pensée peuvent survenir à la consultation d'un album de photos ou à la lecture d'un document historique, nous faisant penser à 'Paul jeune enfant', 'Paul effectuant son service militaire', 'Paris au début du 20ème siècle', 'Paris le jour de la libération', etc. (le lecteur n'aura aucun mal à trouver d'autres exemples)[14]. Il s'agit bien d'objets pensés selon un certain aspect, l'information privilégiée étant une localisation spatio-temporelle. Peu importe que ces objets soient ou non existants actuellement, ce qui importe est qu'ils existent (ou aient existé) réellement (nous traitons le cas des objets fictifs en section suivante) et de les penser dans diverses régions spatio-temporelles, pas uniquement passées (ex : 'Paul au moment de prendre sa retraite'). Dans la localisation spatio-temporelle, nous privilégions le temps. En parlant d'existence passée, actuelle et future, il convient de noter que nous avons pris des engagements ontologiques vis-à-vis du temps. Nous commençons par les expliciter.

De fait, nous avons pris un engagement correspondant à une théorie *présentiste* du temps selon laquelle seuls les objets présents existent et ces objets changent dans le temps [19]. Ceci vaut aussi bien pour les objets physiques que mentaux, y compris les objets pensés. Pour être plus précis, nous considérons que : « toutes les entités, excepté l'espace et le temps, sont *dans* le temps » [25]. Tous les objets viennent à exister, cessent d'exister et subsistent dans l'intervalle. Concernant les modes de persistance, BFO et DOLCE adhèrent à une théorie *endurantiste* pour ce qui concerne les objets

[12] En exposant la conception de l'artefact physique dans DOLCE, Stefano Borgo et Laure Vieu [5] envisagent l'exemple d'un caillou dont un agent vise à se servir comme presse-papier [*ibid.*, p. 21] : "The paperweight is the result of some agent intentionally selecting a pebble and attributing to it certain capacities. The artefact itself is the new entity whose physical realization is the selected object and which has attributed capacities. In particular, the paperweight is a selected pebble together with the attributed capacity to stand firm and hold down paper without damaging it". La capacité attribuée [*ibid.*, p. 23] "is an intentional quality as it is dependent on the intentions of the creator at the time of the creation". De façon étonnante, bien que le caractère de dépendance de la capacité vis-à-vis d'un agent soit reconnu comme étant l'essence de l'artefact, ce dernier est considéré dans DOLCE comme un objet physique co-localisé avec l'objet physique dont il est constitué.

[13] La remarque suivante concernant les références retenues pour définir la notion de fonction en BFO est sans équivoque [35, fn 4] : "Note that all these [references] are realist views: they hold that functions exist, that they are ingredients of being. We do not adress those accounts – maintained for example by Searle (1995) – according to which function talk is a mere façon de parler about things and thus in principle eliminable". Contrairement à cette position, nous considérons que la fonction d'un objet physique est une propriété attribuée mentalement à l'objet et qu'elle n'est donc pas un « ingrédient de l'être » de l'objet. Elle ne relève pas de la dimension physique de l'objet. Par ailleurs, notre démarche consiste tout au contraire à mettre en avant la propriété conceptuelle plutôt que de chercher à l'éliminer.

[14] Récemment lors de conférences IC ont été évoqués le besoin de pouvoir suivre l'évolution d'un patient en vue de retracer son histoire médicale, convoquant des objets tel 'le tableau clinique de Paul en fin de semaine dernière' [16], et le fait que ce besoin ne trouve pas de réponses évidentes dans les ontologies courantes et les langages de représentation disponibles [8].

physiques – ceux-ci existent pleinement (dans leur pleine identité) à tout moment de leur existence – et à une théorie *perdurantiste* pour ce qui concerne les processus et événements (états et changements d'états) – ces derniers sont des entités étendues dans le temps (4D) et tiennent leur identité du fait de gagner dans le temps des parties. Dans cet article, nous nous intéressons prioritairement à la façon dont nous pensons les objets physiques et, nous fondant sur une connaissance de sens commun, nous les considérons comme des entités 3D.

En évoquant en début de section le fait que nous (êtres humains) pensons à des objets physiques occupant différentes régions spatio-temporelles, nous (l'auteur) nous sommes engagés vis-à-vis d'une théorie endurantiste des objets physiques les assimilant à des entités 3D. Notre cadre métaphysique nous invite à considérer autant d'objets pensés singuliers distincts que nous avons de façons de penser à un même objet localisé dans des régions spatio-temporelles distinctes. Les représentations de ces objets pensés sont alors coréférentielles. Une telle conception soulève toutefois potentiellement une question : s'il paraît conceptuellement cohérent que nous disposions de différents objets pensés se référant à un objet physique actuellement présent (une situation illustrée en Fig. 4 avec nos artefacts), comment parler de représentations coréférentielles en cas d'absence actuelle de référence ? À quel même objet un sujet pense-t-il lorsque l'objet physique n'est pas présent ? En réponse à cette question, il peut suffire de dire que le sujet pense à des objets pensés comme étant coréférentiels sans juger de l'existence actuelle d'un référent. Le sujet se contente de considérer que 'Paul enfant' et 'Paul à l'anniversaire de ses vingt ans' réfèrent à une même entité sans accorder de statut ontologique particulier à cette « même » entité. D'une certaine façon, cette « même » entité est juste une entité *idéale* posée par la théorie de l'endurantisme. Précisément, tâchons d'aller plus loin sur cette notion d'entité « idéale ».

La question que nous posons maintenant est de savoir comment nous pensons qu'entre sujets nous pensons à une « même » chose. Etendons l'expérience de pensée du cueilleur d'un champignon vénéneux (que nous devons à Chrudzimski) en considérant que le cueilleur *A* fait part de sa cueillette à un ami *B*. Notre cadre métaphysique nous invite à considérer que *B* entretient deux objets pensés : 'le champignon cueilli par A' – cet objet réfère à un objet physique – et 'le champignon pensé par A comme cueilli' – ce second objet réfère à un objet pensé par *A* se référant au même champignon. De la sorte, *B* pense que *A* pense au même champignon que lui et, réciproquement, *A* pense que *B* pense au même champignon que lui, le tout indépendamment de l'existence actuelle du champignon. Etendons les échanges sociaux et notre analyse à un club de mycologie et nous obtenons finalement un objet idéal pensé par *A, B, C,* etc., une communauté de sujets.

Cette analyse est en substance celle qu'effectue Twardowski dans son [42] *Fonctions et formations.* Une des motivations de Twardowski dans ce traité est de répondre aux critiques de Husserl concernant son psychologisme notamment ontologique (la thèse du doublon d'objets intentionnels que nous avons évoquée au § 2) et de proposer une théorie sémantico-ontologique faisant abstraction d'objets abstraits platoniciens. Dans ce traité, Twardowski propose une théorie sémantique faisant appel à ce qu'il appelle un *abstractum* – un objet pensé déterminé collectivement, obtenu à partir d'une série d'interactions entre sujets [*ibid.*, § 39, p. 374].

> (…) pour parler de façon exacte, le mot ou la phrase éveille autant de pensées qu'il y a d'auditeurs ou de lecteurs, ces pensées n'étant, de surcroît, pas même pareilles. Pourtant, nous faisons abstraction des éléments divers de ces pensées et considérons seulement comme signification du mot ou de la phrase les éléments dans lesquels ces pensées concordent chez les auditeurs et les lecteurs aussi bien que chez celui qui parle ou écrit. Nous parlons donc (…) seulement d'une signification de la formation psychophysique, et non d'autant de significations qu'il s'en éveille, ou peut s'en éveiller, chez les individus sur lesquels elle exerce son effet. Ainsi comprise, la signification n'est donc nullement une formation psychique concrète individuellement déterminée, mais un *abstractum* obtenu à partir d'une série de telles formations concrètes.

L'*abstractum* joue le rôle de l'objet abstrait platonicien, en expliquant que plusieurs sujets puissent penser à la « même » chose, mais Twardowski nous en livre une figure mentale[15]. Les interactions sociales qu'il évoque pour justifier le caractère collectivement déterminé de l'objet pensé sont celles que décrira Kripke [26], d'un premier baptême consistant à nommer une personne (ou un lieu, un monument ou toute autre chose) puis à se transmettre cette référence de générations en générations. Du fait de son statut mental, l'abstractum reste privé et numériquement distinct dans la tête des sujets, mais il est pensé par eux comme un objet idéal devenu atemporel suite aux interactions précitées.

Nous adoptons donc cette théorie de l'*abstractum* – objet mental social, collectivement déterminé. La Fig. 5 illustre la situation d'un sujet entretenant plusieurs objets pensés se référant à Aristote, le représentant selon différents aspects, par exemple : 'le disciple de Platon', 'le précepteur d'Alexandre Le Grand', 'l'auteur de la Métaphysique'. Pour ce sujet, ces objets pensés réfèrent à l'objet idéal $Aristote_{Idéal}$ pensé par toute une communauté de sujets et qui préexistait avant même que lui-même ne pense à ce philosophe. D'une part, pour le sujet, les objets $Aristote_{\#i}$, $Aristote_{\#j}$ et $Aristote_{\#k}$ sont co-référentiels et représentent donc une même entité, $Aristote_{Idéal}$. D'autre part, les expériences de pensée étant semblables pour d'autres sujets, cette même entité acquiert une dimension sociale en étant considérée comme une unique entité de référence pour une communauté.

[15] Le fait que Twardowski ait réussi dans ce traité à justifier la nature mentale de cet *abstratum* divise les commentateurs et reste notoirement une question ouverte [14]. Pour notre part, nous nous rangeons à la prise de position de Denis Fisette [*ibid.*] : « [Twardowski] affirme que le sens ainsi compris n'est pas quelque chose de transcendant par rapport aux fonctions psychiques, quelque chose qui appartient à un troisième monde, mais un *abstractum* résultant d'un processus de formation de concepts qui opère sur le contenu même des fonctions (...) Il ne peut exister nulle part ailleurs que dans l'intellect qui l'a produit ».

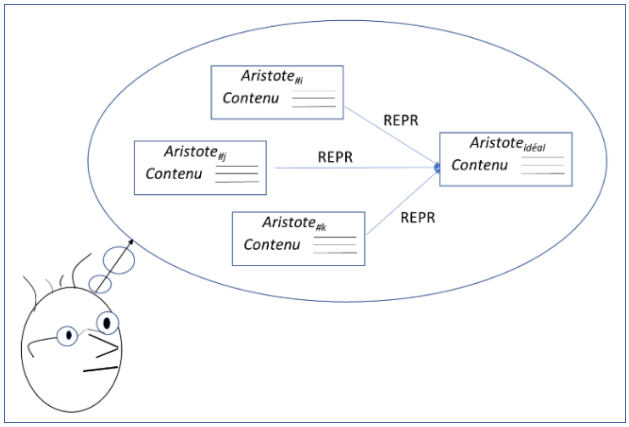


FIGURE 5 – Différents objets pensés se référant à un objet pensé idéal

6 Objets physiques fictifs

Dans cette section, nous nous intéressons à une catégorie d'objets physiques – les dieux de l'Olympe, des figures mythologiques tel Pégase le cheval ailé ou Chiron le centaure, des personnages d'œuvres littéraires comme Anna Karénine, Emma Bovary ou encore Sherlock Holmes – qualifiés de « fictifs » car réputés ne pas exister (ou avoir existé) dans le monde réel spatiotemporel.

Ces entités fictives, comme nous venons de le voir, nous les qualifions (avant tout) d'« objets physiques ». Il s'agit là d'un premier engagement que nous prenons en cohérence avec notre cadre métaphysique, tout particulièrement avec la notion d'*objet pensé*. Nous considérons en effet que de telles entités sont *pensées* comme étant des objets physiques, du fait qu'elles possèdent un corps à l'instar des objets physiques réels. Plus largement, nous considérons que nous *pensons* également les objets les environnant dans leurs mondes fictifs comme des objets physiques, car constitués de matière[16].

Des données psychologiques justifient cet engagement. Umberto Eco, dans son [12] *Quelques commentaires sur les personnages de fiction*, atteste de tels faits psychologiques de pensées. Il souligne qu'à la lecture de romans, nous allons même jusqu'à nous identifier aux personnages de ces romans. La raison en est que, comme dans nos rêves éveillés, nous avons la capacité à nous transporter dans d'autres mondes, le plus souvent proches du monde réel [*ibid.*, § 10] :

> Nous pouvons nous identifier sans problème à des personnages de fiction et à leurs actes parce que, selon une convention narrative, nous nous mettons à vivre dans le monde possible de leur histoire comme s'il était le nôtre.

Nous *pensons* donc ces entités peuplant des mondes possibles comme des entités du monde réel, et les personnages vivants de ces mondes possibles comme des êtres réels vivants. De telles considérations amènent Eco à se poser la question de la *vie* que nous attribuons à ces personnages [*ibid.*].

> De quelle vie particulière vivent les personnages de roman, qui fait que nous sommes capables de les tenir pour plus réels que des personnages réels, et que nous sommes enclins à éprouver les sentiments qu'ils éprouvent, même si nous savons qu'ils n'existent pas ?

Pour répondre à la question, il convient tout d'abord de reconnaître l'homonymie du terme « personnage » qui le fait désigner deux entités distinctes auxquelles nous attribuons deux espèces de vies différentes. Quand on parle du personnage de Sherlock Holmes, il y a tout d'abord ce personnage_1 créé par Conan Doyle, apparu en 1887 dans le roman *Une étude en rouge*, repris ensuite par l'auteur dans d'autres romans et par de nombreux autres auteurs dans différentes œuvres littéraires et films. Ontologiquement parlant, ce personnage_1 est un artefact culturel dont on peut dire qu'il est dans le temps, le temps de notre monde réel spatiotemporel. Ce personnage_1 n'est pas plus fictif qu'une œuvre d'art, une loi, une monnaie, etc., ces entités cataloguées d'entités sociales *abstraites* par Thomasson [38][17]. Par ailleurs, ce personnage_1 a été pensé par Conan Doyle comme étant un détective privé vivant dans le Londres de la seconde moitié du 19ème siècle, violoniste aguerri, féru de médecine et de science. Nous parlons là d'une autre entité, d'un personnage_2 doté d'une autre vie (il serait du reste né en 1850). Entre personnage_1 et personnage_2 existent plusieurs relations que nous ne chercherons pas à élucider dans ce texte. Plutôt, nous revenons aux personnages_2 – autrement dit aux objets physiques fictifs – auxquels Eco fait référence *supra* lorsqu'il indique que nous les tenons pour plus réels que des personnages réels.

Nous venons de voir que nous les identifions à des objets physiques pensés. Cet engagement est cohérent avec la thèse de l'indépendance de l'objet pensé vis-à-vis de l'existence (cf. § 2), ce qui signifie que tant que seul l'objet pensé est pris en compte, la question de son existence ne se pose pas. La propriété d'existence porte techniquement sur la représentation, son attribution relevant d'un jugement de reconnaissance ou de déni de l'objet : l'objet pensé représente-t-il quelque chose d'existant ? En d'autres termes, tant que l'objet pensé est seul concerné, il n'y a pas de « vrai » ou « faux » objet, en l'occurrence physique.

Pour répondre à la dernière question, évoquons une expérimentation menée par Carola Barbero et coll. [2] auprès d'une centaine de sujets profanes en matière de métaphysique. Les sujets devaient évaluer les conditions de vérité de phrases telle « Emma Bovary existe et Barack Obama existe », « Sherlock Holmes existe et Anna Karénine existe » ou « Pénélope Cruz existe et Snazzo existe », faisant varier les termes pour référer à des objets réels, fictifs et non existants.

[16] À ce propos, Mauro Antonelli [1] nous rappelle que pour Brentano « réel » et « exister » sont deux notions distinctes : le *réel* (*Reales*, *Wesenhaftes*) réfère à la substance aristotélicienne et ses accidents, tandis que l'*existence* réfère à ce qui est « correctement affirmé ». Pour Brentano, dès lors, un centaure est une entité réelle car, si elle existait, elle serait un corps.

[17] Rappelons que nous avons fait le choix de ne pas traiter ces entités, pour nous focaliser sur les seules entités sociales *concrètes*.

Selon Barbero et coll. [*ibid.*], la meilleure interprétation psychologique des résultats est que les sujets considèrent que : (i) les termes tel « Barack Obama » et « Emma Bovary » renvoient à des objets existants, au contraire de termes tel « Snazzo » ; par ailleurs, (ii) les objets réels et fictifs ne peuvent toutefois se rencontrer, vivant dans des mondes séparés, ce qui les empêche d'avoir des interactions causales.

En guise d'interprétation de ces résultats, et en complément de nos premiers engagements, nous proposons d'adopter pour les objets physiques fictifs un cadre métaphysique analogue à celui retenu pour les objets physiques réels (cf. §5). Ceci revient à considérer qu'à bien des égards, sur un plan psychologique, les objets fictifs tel Chiron le centaure ou Sherlock Holmes le détective se comportent comme l'objet pensé Barack Obama (existant actuellement) ou Aristote (ayant existé). Une différence est que les objets fictifs sont considérés exister dans des mondes différents du monde réel spatiotemporel. Ces mondes font donc leur entrée dans notre cadre métaphysique. En revanche, une similitude que nous (auteur de l'article) avançons est que nous (sujets pensants) pensons qu'il n'y a qu'un seul Sherlock Holmes, ce détective correspondant au personnage$_1$ créé par Conan Doyle, de même qu'il n'y a qu'une seule Anna Karénine, créée par Tolstoï. Nous tirons prétexte de cette remarque pour conférer une existence à l'objet pensé *idéal*, comme nous l'avons fait en § 5 avec l'objet physique réel, représentant l'objet considéré comme pensé collectivement par une communauté. Une figure similaire à la Fig. 5 est donc à considérer en remplaçant le nom Aristote par exemple par Sherlock Holmes.

7 Conclusion

Dans cet article, faisant suite à notre *Plaidoyer pour des ontologies épistémiques* [25], nous avons continué à promouvoir un cadre métaphysique et une notion d'ontologie dont le trait d'union est l'*objet pensé* mental.

Sur le plan métaphysique, nous avons défendu l'objet pensé en convoquant des théories psychologiques et montré que des données récentes, par exemple concernant les entités fictives, confirment la plausibilité psychologique d'une telle entité. Rappelons à ce propos la proximité de notre cadre métaphysique avec les travaux conduits en philosophie de l'esprit sur les *dossiers mentaux* [29,32].

Sur le plan de l'ingénierie des connaissances, nous aboutissons à deux résultats remarquables. D'une part, avec les objets pensés, nous tenons des objets pour représenter le monde physique auxquels sont attribuées des propriétés à la fois physiques et non physiques. De notre point de vue, ceci vient légitimer une pratique courante en représentation des connaissances et suggère que les instances communément considérées dans les bases de connaissances correspondent à des objets pensés mentaux. De fait, et c'est là un second résultat important, nous avons montré que les objets pensés permettent de tenir compte des différentes perspectives *épistémiques* que nous entretenons, en tant que sujets connaissant, sur des objets physiques *en soi* (principe du réalisme physique) endurant et changeant dans le temps, dont certains ont cessé d'exister. Nous considérons que cette possibilité ouvre de nouvelles voies pour rendre compte aussi bien des objets artéfactuels que des objets que nous nous avons qualifiés de « spatio-temporalisés », en permettant de corriger des problèmes rencontrés dans le traitement de ces objets par les ontologies courantes en Ontologie appliquée.

Remerciements

Nous remercions Adrien Barton ainsi que les relecteurs anonymes de leurs précieux commentaires sur des versions antérieures de l'article.

Références

[1] M. Antonelli, Franz Brentano's Intentionality Thesis. A New Objection to the "Nonsense that was Dreamt up and Attributed to him", *Brentano Studien*, Vol.13, pp. 23-53, 2015.

[2] C. Barbero, F. Domaneschi, I. Enrici & A. Voltolini, What is Existence? A Matter of Co(n)text, *Acta Analytica*, 2023.

[3] S. Borgo, R. Ferrario, A. Gangemi, N. Guarino, C. Masolo, D. Porello, E.M. Sanfilippo & L. Vieu, DOLCE: A descriptive ontology for linguistic and cognitive engineering, *Applied Ontology*, Vol. 17, pp. 45-69, 2022.

[4] S. Borgo, A. Galton, & O. Kutz, Foundational ontologies in action, *Applied Ontology*, Vol. 17, pp. 1-16, 2022.

[5] S. Borgo. & L. Vieu, Artifacts in formal ontology. In A. Meijers (ed.), *Handbook of the Philosophy of the Technological Sciences. Technology and Engineering Sciences* (Vol. 9, pp. 273-307). Elsevier, 2009. Doi:10.1016/B978-0-444-51667-1.50015-X.

[6] F. Brentano, *Psychologie du point de vue empirique*, Aubier, Paris, 1944 ; 2nd éd. revue par J.-Fr. Courtine, Vrin, Paris, 2008 ; trad. fr. par M. de Gandillac de *Psychologie vom empirischen Standpunkt*, vol. I, O. Kraus (ed.), Leipzig: Meiner, 1874.

[7] R. Casati, Commonsense, Philosophical and Theoretical Notions of an Object: Some Methodological Problems, *The Monist*, Vol. 88, N° 4, pp. 571-599, 2005.

[8] W. Charles, N. Aussenac-Gilles & N. Hernandez, Temporalité et graphes de connaissances : analyse théorique et enjeux pratiques, dans C. Trojahn (ed.), *Actes des 34es Journées Francophones d'Ingénierie des Connaissances (IC) Plate-Forme Intelligence Artificielle (PFIA 2023)* (pp. 100-109), 2023.

[9] A. Chrudzimski, Brentano and Aristotle on the Ontology of Intentionality, in D. Fisette & G. Fréchette (Eds.), *Themes from Brentano* (pp. 121-137), Amsterdam: Rodopi, 2013.

[10] A. Chrudzimski, Intentional Objects and Mental Contents, *Brentano Studien*, Vol. 13, pp. 81-119, 2015.

[11] J. Dubucs et W. Miśkiewicz, Logic, act and product, in G. Primiero (ed.), *Knowledge and Judgment* (pp. 209-215) Springer-Verlag, 2009.

[12] U. Eco, Quelques commentaires sur les personnages de fiction, *SociologieS*, 2010 ; trad. fr. par F. Farrugia de On the ontology of fictional characters: A semiotic approach, *Sign Systems Studies*, Vol. 37, N° 1-2, pp. 82-97, 2009.

[13] J.L. Falguera, C. Martínez-Vidal & G. Rosen, Abstract Objects, in Edward N. Zalta (ed.), The Stanford Encyclopedia of Philosophy (Summer 2022 Edition), URL = https://plato.stanford.edu/archives/sum2022/entries/abstract-objects/

[14] D. Fisette, Overcoming Psychologism: Twardowski on Actions and Products, in A. Dewalque *et al.* (eds.), *Philosophy of Language in the Brentano School: Reassessing the Brentanian Legacy*, Palgrave Macmillan, 2021.

[15] G. Fréchette, Homeless Objects, *Grazer Philosophische Studien*, Vol. 100, pp. 207-230, 2023.

[16] J. Hilbey, X. Aimé & J. Charlet, Représentation des connaissances médicales temporelles au moyen d'ontologies, dans F. Saïs (ed.), *Actes des 32es Journées Francophones d'Ingénierie des Connaissances (IC) Plate-Forme Intelligence Artificielle (PFIA 2022)* (pp. 147-152), 2022.

[17] E. Husserl, Objets intentionnels, dans J. English (ed.), *Husserl – Twardowski : Sur les objets intentionnels (1893-1901)* (pp. 279-326), Paris : J Vrin, 1993 ; trad. fr. par J. English de « Husserl Abhandlung "Intentionale Gegenstände", Edition der ursprünglichen Druckfassung », *Brentano Studien*, Vol. 3, 1990/1991, p. 137-176.

[18] R. Ingarden, *Ontology of the Work of Art*, Ohio: Ohio University Press, 1989 ; trad. par R. Meyer et J.T. Goldthwait de *Untersuchungen zur Ontologie der Kunst*, Tübingen: Max Niemeyer, 1962.

[19] D. Ingram & J. Tallant, Presentism, in E. Zalta & U. Nodelman (eds.), *The Stanford Encyclopedia of Philosophy (Winter 2023 Edition)*, URL = https://plato.stanford.edu/archives/win2023/entries/presentism/

[20] G. Kassel, A formal ontology of artefacts, *Applied Ontology*, Vol. 5, N° 3-4, pp. 223-246, 2010.

[21] G. Kassel, Processes Endure, Whereas Events Occur, in S. Borgo, R. Ferrario, C. Masolo & L. Vieu (eds.), *Ontology Makes Sense: Essays in honor of Nicola Guarino* (pp. 177-193), Frontiers in Artificial Intelligence and Applications, 136, IOS Press, 2019.

[22] G. Kassel, Physical processes, their life and their history, *Applied Ontology*, Vol. 15, N° 2, pp. 109-133, 2020.

[23] G. Kassel, Abstract events in semantics, *Philosophia*. Vol. 50, N° 2, pp. 1913-1930, 2022.

[24] G. Kassel, Connexions et relations, in C. Trojahn (ed.), *Actes des 34es Journées Francophones d'Ingénierie des Connaissances (IC) Plate-Forme Intelligence Artificielle (PFIA 2023)* (pp. 133-142), 2023.

[25] G. Kassel, A plea for epistemic ontologies, *Applied Ontology*, Vol. 18, N° 4, pp. 367-397, 2023.

[26] S.A. Kripke, *Naming and Necessity*, Cambridge, MA: Harvard University Press, 1980.

[27] C. Masolo, Understanding Ontological Levels, in F. Lin & U. Sattler (eds.), *Proceedings of the Twelfth International Conference on the Principles of Knowledge Representation and Reasoning (KR 2010)* (pp. 258-268), AAAI Press, 2010.

[28] A. Meinong, La théorie de l'objet, dans *A. Meinong, Théorie de l'objet et Présentation personnelle* (pp. 63-114), Paris : Vrin, 1999 ; trad. fr. par M. de Launay et J.-F. Courtine de Über Gegenstandstheorie, in *Untersuchungen zur Gegenstandstheorie und Psychologie* (pp. 1-51), Leipzig: J.A. Barth, 1904.

[29] M. Murez & F. Recanati, Mental files: an Introduction, *Review of Philosophy and Psychology*, Vol. 7, N° 2, pp. 265-281, 2016.

[30] J.N. Otte, J. Beverley & A. Ruttenberg, BFO: Basic Formal Ontology, *Applied Ontology*, Vol. 17, pp. 17-43, 2022.

[31] R. Poli, Levels of reality and the psychological stratum, *Revue internationale de philosophie*, Vol. 2, N° 236, pp. 163-180, 2006.

[32] F. Recanati, *Mental files*, Oxford: Oxford University Press, 2012.

[33] J. Searle, *The Construction of Social Reality*, New York: The Free Press, 1995.

[34] B. Smith & J. Searle, The Construction of Social Reality: An Exchange, *The American Journal of Economics and Sociology*, Vol. 62, N° 1, pp. 285-309, 2003.

[35] A.D. Spear, W. Ceusters & B. Smith, Functions in Basic Formal Ontology, *Applied Ontology*, Vol. 11, pp. 103-128, 2016.

[36] E.S. Spelke, Principles of Object Perception, *Cognitive Science*, Vol. 14, pp. 29-56, 1990.

[37] M. Textor, Brentano, on Act, Content and Intentionality, *Grazer Philosophische Studien*, Vol. 100, N°1-2, 173-196, 2023. https://doi.org/10.1163/18756735-00000176

[38] A.L. Thomasson, Foundations for a Social Ontology, *Protosociology*, Vol. 18-19, pp. 269-290, 2003.

[39] A.L. Thomasson, Ingarden and the Ontology of Cultural Objects, in A. Chrudzimski (ed.), *Existence, Culture, and Persons: The ontology of Roman Ingarden* (pp. 115-136), Frankfurt: ontos, 2005.

[40] A.L. Thomasson, Artifacts and Human Concepts, in E. Margolis & S. Laurence (eds.), *Creations of the Mind. Theories of Artifacts and Their Representation* (pp. 52-73), Oxford University Press, 2007.

[41] K. Twardowski, Sur la théorie du contenu et de l'objet des représentations, dans J. English (éd.), *Husserl – Twardowski, sur les objets intentionnels (1893-1901)*, Paris, Vrin, pp. 85-200, 1993 ; trad., introduction et notes par J. English de *Zur Lehre vom Inhalt und Gegenstand der Vorstellungen. Eine psychologogische Untersuchung*, Vienne, Hölder, 1894.

[42] K. Twardowski, Fonctions et formations. Quelques remarques aux confins de la psychologie, de la grammaire et de la logique, dans D. Fisette et G. Fréchette (dir.), *À l'école de Brentano. De Würzbourg à Vienne* (pp. 343-383), Paris, J. Vrin, 2007 ; trad. fr. par L. Joumier et J. Plourde de Über Gebilde und Funktionen. Einige Bemerkungen zum Grenzgebiete der Psychologie, Grammatik und Logik, dans A. Ruge (dir.), Die Philosophie der Gegenwart, Heidelberg:Weiss, 1912.

Extraction d'information, Annotation

Extraction automatique de règles pour la détermination de types de relations sémantiques dans les constructions génitives en français

H. Guenoune[1],[2], M. Lafourcade[1],[2]

[1] Université de Montpellier, France
[2] Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier, LIRMM

{hani.guenoune, mathieu.lafourcade}@lirmm.fr

Résumé

Cette étude concerne les relations sémantiques portées par les entités sous forme génitive « A de B ». Après identification des types sémantiques pertinents, nous construisons, à l'aide d'une IA générative, un corpus annoté. Nous proposons un algorithme de découverte des règles permettant de sélectionner la relation entre A et B. Ces règles correspondent à la sélection dans une base de connaissances du voisinage adéquat d'un terme donné. Soit « désert d'Algérie », portant la relation de lieu, le terme désert identifié comme lieu géographique et Algérie comme pays. Ces contraintes aboutissent par calcul à une règle permettant de sélectionner la relation de lieu.

Mots-clés

Génitif - Entités polylexicales - Relations sémantiques.

Abstract

We are interested in the semantic relations conveyed by polylexical entities in the postnominal prepositional noun phrases form "A de B" (A of B). After identifying a relevant set of semantic relations types, using generative AI, we build a collection of phrases, for each relation type identified. We propose an algorithm for creating rules allowing the selection of the relation between A and B in noun phrases of each type. Rules consist in selecting from a knowledge base the appropriate neighborhood of a given term. For the phrase "désert d'Algérie" carrying the location relation, "désert" is identified as a geographical location, and "Algérie" as a country. Constraints are used to learn rules for selecting the location relation for this type of example.

Keywords

Genitive, Postnominal construction, Semantic relations.

1 Introduction

Au-delà de la nécessité d'identification des entités polylexicales pour l'analyse automatisée du langage, il est important pour diverses applications, de cerner la nature des rapports qui lient les différents composants des termes polylexicaux. Nous nous intéressons au cas génitif du complément de nom « de N » (construction « post-nominale » [1]). En d'autres termes, les mots composés construits à travers l'emploi de la préposition "de" introduisant un complément syntaxique à une tête nominale (*« A de B »*, où *A* et *B* sont des noms). Nous cherchons dans ce travail à identifier de manière automatique la relation sémantique entre les termes *A* et *B* dans les formes *« A de B »* (et les variantes, *« A d'B », « A du B », etc.)*. Une telle démarche peut servir à une interprétation plus riche de contenus textuels, et permet de mettre au point des systèmes aboutissant à des représentations sémantiques connexes. Parmi les applications dont bénéficieraient une telle étude, nous pouvons citer la tâche de question-réponse (*Question Answering* ([4, 6]) qui nécessite une représentation sémantique suffisante du texte et des rapports entre les entités qui y sont mentionnées. Ou encore, la tâche de résolution des anaphores déclenchées par un déterminant possessif consistant en une transformation de formes post-nominales en syntagmes anaphorisés (*« le vélo de Julie → son vélo »*) et dont la résolution se base sur des contraintes sur la nature des relations entre l'anaphore et son antécédent [5].

Dans le cadre spécifique à notre projet, ces efforts sont aussi menés dans une perspective de consolidation d'une base de connaissances de sens commun, passant notamment par l'identification des types de relations sémantiques dont l'intégration s'avère la plus intéressante pour nos différents mécanismes d'inférences ainsi qu'aux applications de Traitement Automatique du Langage Naturel (*TALN*) qui exploitent la base de connaissances. Un autre moyen d'améliorer la qualité de la base de connaissances est de développer un système de classification servant d'*outil de contrôle*. L'analyse de la justesse des résultats d'un tel outil apporterait des éclaircissements sur la qualité globale des connaissances utilisées. Pour que cela soit possible, l'accent doit être mis sur *l'explicabilité* des méthodes à utiliser. Il est en effet essentiel de disposer des explications des résultats du système afin d'être en mesure d'identifier les lacunes potentielles et mettre en évidence les moyens appropriés de consolider la base de connaissances.

Dans *« A de B »*, la tête nominale (*A*) joue un rôle important dans le sens qu'entretient le syntagme avec son complément (*B*). Les types de noms liés syntaxiquement par la préposition dans une construction génitive conditionnent

la relation sémantique qui les lie. La distinction citée dans [1, 2, 3], différencie l'usage des noms de *classes d'entités (sortal nouns)* des *noms relationnels (relational nouns)*. La différence dans leurs définitions est analogue à celle des prédicats unaires et binaires de la logique de premier ordre. Certains noms ne prennent leurs sens qu'en étant rapportés à exactement deux arguments. L'exemple cité est celui des noms de famille [*père, mère, sœur, frère...*]. La relation entretenue entre le nom et son complément dans la phrase « la mère de Lucie» se rapporte directement à la sémantique de la tête nominale (le nom *relationnel "mère"*). Un syntagme à tête nominale relationnelle permet donc une *interprétation lexicale* du type de relation sémantique et s'oppose aux cas de lecture pragmatique qui résultent de l'emploi de certains noms de classe (*sortal nouns*) et qui nécessitent de recourir à des informations extra-lexicales afin d'identifier la nature du lien sémantique entre les termes du syntagme. Parler du *« nuage de Marie »* implique d'introduire des éléments d'ordre pragmatique afin d'être en mesure d'interpréter le type de relation (*le nuage qu'elle regardait, qu'elle dessinait, dont elle a rêvé, etc.*).
Les sens portés par ce type de syntagme sont donc variés, quand bien même les efforts d'interprétation automatiques réduisent bien souvent le type de relations sémantiques à la relation d'appartenance/possession (un élément d'explication serait l'importance de la relation de possession et son rôle prépondérant dans les typologies sémantiques standards considérées dans les travaux de *TALN*).

Au-delà du cadre typologique des têtes nominales, les noms utilisés (qu'ils soient relationnels ou de classe) introduisent une multitude de types de relations possibles entre les deux termes. Ce travail cherche à étudier la nature des liens sémantiques dans cette configuration, et vise à proposer une *typologie sémantique* de tels syntagmes polylexicaux. En outre, nous proposons un algorithme d'apprentissage, servant de support à un système de classification des types dans les syntagmes polylexicaux en complément de nom (avec « *de* »).
Les sens figurés étant difficilement accessibles au calcul, ce travail n'aborde pas la détermination du sens global de la forme *« A de B »* quand celle-ci a également acquis un sens idiomatique *(par exemple : « homme de paille / écran de fumée »)*.

Cet article portera donc sur plusieurs points :

Proposition d'une typologie des relations sémantiques dans les syntagmes en complément de nom, puis production d'un corpus associatif entre des exemples de constructions génitives et les types de relations qui y correspondent. Les données sont collectées en exploitant *avec prudence* une IA générative, puis validées manuellement. Une partie de ce corpus servira de données d'entraînement et l'autre d'ensemble de test à un système d'apprentissage de règles de classification ;

Présentation de *GRASP*-it *(Genitive Relations And Semantic Pattern Identification Tool)*, un algorithme d'apprentissage calculant des règles de décision pour le/les types de relations probables ;

Évaluation de la qualité des règles produites en implémentant un deuxième algorithme de classification des types sémantiques dans les formes *« A de B »*.

Nous commençons par une présentation des ressources utilisées dans le cadre de ce projet, en l'occurrence la base de connaissances pour laquelle nous souhaitons apporter des améliorations à travers cette classification des types sémantiques, puis les données ayant servi au développement de *GRASP*-it. Nous donnons, dans cette section, quelques exemples de chaque type de relation résultant de l'emploi des constructions en « de N ». Nous décrivons ensuite le mécanisme d'apprentissage mis en place afin de synthétiser, - automatiquement et sous forme de couples de contraintes -, les rapports sémantiques entre les termes de chaque type présenté dans la section qui précède. Pour finir, nous proposons une évaluation de la qualité des règles produites, réalisée à travers une procédure d'identification des types de relations appliquée à une portion du corpus.

2 Ressources

Ce projet a nécessité l'emploi de ressources externes pour mener à bien l'apprentissage réalisé par l'algorithme *GRASP*-it, mais également pour la mise au point d'heuristiques de classification et l'évaluation du système réalisé.

2.1 Base de connaissances utilisée

L'étude et les algorithmes développés sont soutenus par une base de connaissances issue de la dernière version de la collection de données du projet *JeuxDeMots* (*JDM*) (datée du 11 février 2024).

JeuxDeMots (*JDM*) [9] est un réseau lexico-sémantique représenté par un graphe orienté. Les nœuds du graphe sont des termes ou des informations de nature symbolique, tandis que les arcs indiquent des relations typées, pondérées et potentiellement annotées entre les nœuds. Le graphe aborde la polysémie lexicale en spécifiant des sens hiérarchiques "*raffinements*", où un sens spécifique est affilié au sens général du terme. *JDM* repose sur des outils, principes et concepts pratiques (*e.g.* la notion de raffinement, la diversité des types sémantiques, des relations inverses telles que *r_isa* et *r_hypo*, ainsi qu'une série d'outils collaboratifs).
Le réseau *JDM* est conçu pour être utilisé comme support de connaissances pour les solutions d'IA (analyse sémantique de texte, raisonnement, prise de décision, résumé automatique, etc.). Un système de pondération et d'évaluation symbolique (annotation de méta-informations, *e.g. rare, pertinent, non-pertinent*, etc.) a été mis en œuvre pour faciliter le parcours du graphe et son exploitation.
Au 1er février 2024, *JDM* contenait environ $540\ millions$ de relations entre plus de $9\ millions$ de termes et $24\ millions$ de nœuds [8].

Un des objectifs centraux de ce travail est d'enrichir notre base de connaissances avec des informations sémantiques, en particulier celles concernant les relations dans les syntagmes nominaux génitifs. Cela contribuera principalement aux tâches d'analyse textuelle et à l'extraction de connaissances, en particulier. En effet, lorsqu'on rencontre dans

un texte une forme génitive « A de B », il est souhaitable d'avoir des outils d'identification de la (ou des) relation(s) entre A et B.

2.2 Corpus de constructions génitives

Nous proposons un corpus de petite taille servant à l'apprentissage des règles de détermination de type sémantique et à leur évaluation. Ce corpus, mis à la disposition de la communauté, peut également être vu comme un point de départ à la création de collections de constructions génitives de plus grande envergure. En effet, en dépit de l'importance des petits corpus [10], en deçà de plusieurs milliers d'exemples, il se révèle difficile de destiner ces données à une exploitation par des procédures gourmandes en ressources telles que des algorithmes d'apprentissage neuronaux. Toutefois, nous souhaitons inscrire cet effort dans un projet à plus long terme dans lequel des procédures d'augmentation de données peuvent être mises en place, tels que des mécanismes automatiques d'enrichissement sémantique ou encore une complétion par annotation manuelle.

Dans ce qui suit, nous détaillons le protocole d'acquisition et de validation des données, puis listons les types sémantiques identifiés pour les relations entre les éléments de la paire de syntagmes dans une construction génitive. Afin d'éviter qu'un quelconque biais soit introduit, nous avons choisi de collecter des données à partir d'une source indépendante de la base de connaissances *JDM*.

2.2.1 Typologie sémantique

Dans le Tableau 1, nous listons les types sémantiques que nous avons choisis de considérer. Nous apportons pour chacun d'eux une explication et quelques exemples, ainsi que le type de relation correspondant dans *JDM* (la relation sémantique avec l'orientation appropriée, les relations dont le nom a la forme *'r_x-1'* étant la relation conversive de *'r_x'*).

Type de relation	**Relation *JDM***
Conséquence (Co) : *Le terme A est une conséquence de (est causé par) B.*	*r_has_causatif*
dégâts de la tempête - retards de la circulation	
Caracterisation (Ca) : *Le terme A est une propriété ou le nominalisation d'un adjectif pouvant qualifier B.*	*r_has_property-1*
sournoiserie du politicien - sagesse du viellard	
Matière/composition (M) : *Le terme A est composé de ou est de la matière B.*	*r_objet>matière*
cuillère de bois - trône de fer	
Origine (O) : *Le terme A est originaire de B.*	*r_lieu>origine*
vin de France - café du Brésil	
Topic *(T)* : *Le terme A a pour thème (ou sujet) le terme B.*	*r_topic*
restaurant de sushis - film d'horreur	
Dépiction (D) : *Le terme A est une représentation du terme B.*	*r_depict*
peinture d'un paysage - photo d'une famille	
Holonymie (H) : *Le terme A fait partie de B.*	*r_holo*
coque du bateau - écaille du poisson	
Lieu (L) : *Le terme peut avoir pour lieu le terme B.*	*r_lieu*
tour de Pise - sahara d'Algérie	
Agent (A) : *Le terme A est la nominalisation d'une action dont l'acteur est le terme B.*	*r_processus _agent*
travail de l'ouvrier - cours du professeur	
Patient (P) : *Le Terme A est la nominalisation d'une action que le terme B subit.*	*r_processus _patient*
travail du bois - ouverture de la porte	
Instrument (I) : *Le terme A est instrument de l'action B ou d'une action que le terme B subit.*	*r_processus >instr-1*
clé d'ouverture - clé de la porte	
Possession (Po) : *A est possédé par B*	*r_own-1*
fusil du soldat - vélo du cycliste	
Quantification (Q) : *A sert de mesure à B.*	*r_quantificateur*
brin d'herbe - minute d'attente	
Lien social (LS) : *A tient un rôle de 'A' vis-à-vis de B.*	*r_social_tie*
avocat d'une femme battu - chef du groupe	
Auteur/créateur (AC) : *A est produit par B.*	*r_product_of*
portrait de Van Gogh - gâteau du pâtissier	

TABLE 1 – Liste des types sémantiques considérés dans les entités « *A de B* » et leur correspondances avec les types de relations sémantiques dans le réseau *JDM*.

Il est à noter que cette liste est celle que nous avons arrêtée en tant que typologie de base de notre étude. Les choix concernant la granularité et le nombre de types ont été faits de sorte à aligner cette typologie avec les exigences des ressources et des outils que nous utilisons ainsi que les be-

soins des applications dans lesquelles cette typologie est exploitée. La typologie adoptée en tant qu'état de l'art pour l'étude des modifieurs de noms est donnée dans [7]. Il ne s'agit, dans notre cas, aucunement d'une liste exhaustive de tous les types de relations possibles entre les termes A et B d'une forme « *A de B* ». Quelques cas peuvent s'ajouter à cette liste. Il est également possible de *spécifier/généraliser* certains types de manière à ce qu'ils correspondent plus ou moins précisément à des cadres théoriques différents du nôtre. Notamment dans le cas d'une exploitation avec une autre base de connaissance (que JDM), définissant un ensemble de types de relations différent. Parmi ces cas, nous pouvons mentionner les exemples portant des relations sémantiques temporelles *absolues* (portées par des noms de classe), « *repas de midi - brise du matin - bus de nuit* », ou encore des lien *relatifs* spatiaux et temporels (portés par des noms relationnels) « *milieu/droite/gauche de la pièce - bas de page* ». Un autre cas est celui des nominations : « *Théorème de Pythagore - Râle de Wallace - Kappa de Fleiss* » qui pourrait faire l'objet d'une catégorie à part entière. Nous choisissons pour les cas mentionnés de les inclure dans des types de sémantique similaire, par exemple, les deux premiers cas sont inclus dans le type *topic*, tandis que le cas des nominations, lui, est classé parmi les instances de *auteur/créateur* (même s'il n'est pas systématiquement question de création).

2.2.2 Collecte de données et validation

Pour chaque type de relation sémantique ci-dessus, nous avons fait appel à une IA générative [1]) pour créer un ensemble d'exemples. Nous avons choisi de construire un corpus de forme génitives à partir d'une source indépendante de la base de connaissances JDM. La première raison est de s'assurer de ne pas introduire de biais entre les données du corpus (servant à l'apprentissage et au test de notre algorithme) avec la base de connaissances utilisée pour l'extraction des attributs représentant la sémantique des termes (extraction des *signatures*, voir 3). Une seconde raison est la nécessité d'annoter un syntagme « A de B » issu de JDM avec la (ou les) relations sémantiques attendues entre A et B. Or cette information n'est pas systématiquement disponible dans JDM alors qu'elle peut être demandée à l'IA générative. Un travail manuel important deviendrait donc nécessaire et limitant. D'aucun pourrait penser que la base de connaissances pourrait contenir une ou plusieurs relations sémantiques entre les termes A et B, mais rien ne garantirait qu'il s'agisse des mêmes relations que celles attendues (introduites par le syntagme).

Concrètement, nous avons fait le choix de limiter chaque type considéré à 80 exemples, dont 50 sont consacrés à l'apprentissage et 30 servent à l'évaluation de l'algorithme présenté dans la section 3. La stratégie de construction des requêtes émises à l'agent conversationnel a été différente selon les types de relations. En effet, il s'est avéré plus ou moins difficile, selon les cas, d'obtenir des exemples satisfaisants. Pour les types où les exemples générés étaient peu exploitables, nous avons choisi d'orienter le modèle par l'exemple. Nous avons procédé en donnant une dizaine d'exemples rédigés par nos soins, puis en expliquant les points communs au niveau des relations sémantiques sous-jacentes. Ces explications se sont étalées sur plusieurs *tours de parole* avec l'agent conversationnel.
Quand bien même cette démarche itérative a permis d'obtenir les exemples du type recherché, elle présente néanmoins l'inconvénient « *d'influencer* » excessivement les réponses produites par le *chatbot*, ce qui mène à un ensemble d'entités polylexicales de faible diversité (forte adéquation aux exemples proposés à l'agent). Il a par conséquent été nécessaire, dans un but de diversification, à partir d'une première génération d'exemples, d'insister sur le fait de réitérer la génération en cherchant à la diversifier. Cette stratégie a été répétée jusqu'à ce que nous ayons considéré l'ensemble comme convaincant. Toutefois, les ensembles contenaient environ 10% d'exemples mal classifiés ou dupliqués et sont également restés imparfaits concernant leur diversité. Nous avons donc procédé à une validation manuelle de l'ensemble des exemples produits par le *chatbot*. Précisément, la validation a consisté en un remplacement des cas de duplication et des entités trop similaires, ainsi qu'un reclassement des exemples mal classés.

2.2.3 Mise en forme et post-traitement des données

La collection produite inclut des exemples de morphologie variable. En effet, en ce qui concerne la présence ou non de déterminant du complément du nom (terme B), on pourrait supposer qu'une étape de normalisation morphologique serait intéressante. Il s'avère néanmoins que ce critère constitue un marqueur morpho-syntaxique pouvant se révéler crucial pour une classification, raison pour laquelle nous faisons le choix de ne pas opérer de transformation au niveau morphologique ou lexical. Toutefois, notons qu'une polysémie des termes du corpus peut éventuellement être observée. L'exploitation du corpus nécessitera donc de préparer les données, en l'occurrence une phase de désambiguïsation sémantique devra être menée afin de sélectionner les sens appropriés des termes.

3 Présentation de *GRASP*-it

L'algorithme *GRASP*-it (*Genitive Relations And Semantic Pattern Identification Tool)* vise à produire un ensemble de paires de contraintes pour chaque type de relation en se basant sur les données d'entrée. Ces contraintes sont fondées sur les types sémantiques de la tête nominale et du complément. Elles peuvent être considérées comme une synthèse des attributs sémantiques alignée avec le contenu d'une base de connaissances. Le but de cet ensemble de contraintes est de guider un processus de classification des relations sémantiques dans les syntagmes nominaux génitifs.
Un autre objectif est de produire des contraintes "interprétables" qui peuvent être facilement lues et expliquées. Dans le cas général, la première étape de *GRASP*-it implique de stocker, pour chaque exemple d'un certain type, des informations sémantiques qui pourraient permettre de classer

1. LLM GPT. Version du modèle : gpt-4-0613, datée du 13-06-2023.

l'exemple dans le type pertinent :

- *Hyperonymes des termes A et B* **(H)** : L'objectif est de capturer, aussi précisément que possible, les "types" sémantiques des deux termes. Un hyperonyme est un terme (entité lexicale) dans JDM accessible via la relation *r_isa*.
- *Cible des types de relation* **(TRT)** : Une sélection de types de relation conduisant au terme. Par exemple, un terme fréquemment ciblé par la relation de localisation est considéré, par cette approche, comme une localisation. Cela permet de renforcer la pertinence de cette classe sémantique pour un terme spécifique. La sélection des relations conduisant aux termes peut être vue comme un moyen de compléter la liste des hyperonymes pour un terme donné.
- *Type sémantique standard* **(SST)** : À travers la relation *_INFO-SEM*, le type standard associe un terme lexicalisé à un type ontologique (conceptuel) standard.

Le résultat de cette étape est un ensemble de paires pondérées, appelées ici *signatures* des termes A et B. Le nombre de paires à cette étape correspond au nombre d'exemples pour chaque type, qui, dans le cas de notre corpus (portion d'entraînement), s'élève à 50 unités de la forme « *A de B* ». Une signature est définie comme un ensemble non ordonné de symboles. Chaque symbole prend une valeur d'une entrée spécifique de JDM. Par exemple, la signature s associée au terme « *véhicule* » serait la suivante.

```
s(véhicule) = {
véhicule, transport urbain, partie de
l'espace, Transport urbain , mode de
transport, instrument, lieu, transport,
moyen, machine, moyen de transport,

r_isa, r_hypo, r_has_part, r_holo,
r_agent, r_patient, r_lieu, r_instr,
r_carac-1, r_lieu-1, r_action_lieu,
r_mater>object, r_processus>agent,
r_own, r_is_instance_of,

_INFO-SEM-SUBST, _INFO-SEM-THING-
ARTEFACT, _INFO-SEM-PLACE, _INFO-SEM-
THING-CONCRETE, _INFO-SEM-PLACE-HUMAN
}
```

Pour des raisons de clarté, nous avons divisé les symboles en trois blocs : Hyperonymes, TRT et SST. Il convient de noter que la signature d'un terme contient le terme lui-même, dans le but de capturer des instances qui sont des hyponymes du terme signé. En plus de la nécessité de son explicabilité, cette représentation des termes est conçue pour être contrôlable du point de vue de son contenu et de sa taille. Cela permet à la méthode *GRASP*-it d'être adaptable aux exigences variables de l'application pour laquelle elle est utilisée.

La deuxième étape vise à agréger les règles de chaque type pour traiter l'ensemble complet par généralisation. Comme le montre (1), nous définissons une règle R comme une paire de contraintes sL et sR (qui sont des signatures, respectivement gauche et droite correspondant aux termes A et B) et un type de relation sémantique rt.

$$R :< s_L, s_R, rt > \tag{1}$$

L'agrégation est une opération de fusion de deux règles et est définie dans (2).

$$Fusion(R1, R2) =< s1_L \cup s2_L, s1_R \cup s2_R, rt > \text{ [2]} \tag{2}$$

Une fusion de deux règles signifie que les contraintes qu'elles associent respectivement sont suffisamment similaires pour être représentées par une seule paire de contraintes. Formellement, comme une signature s peut être vue comme un vecteur, nous avons adopté la similarité cosinus (produit scalaire divisé par le produit des normes), notée sim. Deux signatures sont considérées comme suffisamment similaires lorsque leur valeur sim est supérieure à un seuil de 0.5 (établi empiriquement). La signature fusionnée est la somme vectorielle des deux signatures (ce qui correspond à l'union ensembliste).

Une paire produite par une ou plusieurs fusions successives est considérée comme plus générale et fiable qu'une paire qui n'a pas subi de fusion. La fiabilité est donc une mesure de la couverture des exemples du type et est calculée en attribuant un poids à la paire de contraintes correspondant au nombre de fusions effectuées pour arriver à la forme finale de la paire. À la sortie de cette étape, un ensemble de paires de contraintes plus ou moins agrégées, avec une cardinalité d'au plus deux fois le nombre d'exemples du type considéré, est attribué à chaque type de relation considéré. L'idée derrière la fusion des règles est que le résultat de fusions successives est une règle qui représente de manière appropriée un grand ensemble d'exemples d'un certain type. Une règle fusionnée peut donc être considérée comme un *modèle généralisé* pour un type de relation donné. Un type de relation peut être associé à plusieurs modèles. Un bon modèle associera de manière appropriée le type de relation entre deux termes A et B dans un syntagme génitif.

4 Evaluation de *GRASP*-it

Dans cette section nous présentons les conditions dans lesquelles notre évaluation a été réalisée et finissons par l'analyse des scores de performance de notre système.

4.1 Préparation des données

Une phase minimale de préparation des données a été entreprise avant d'appliquer l'algorithme de classification (détaillé dans la section 4.2). Afin de réussir à prendre en compte l'intégralité du corpus lors de l'entraînement ainsi que de la classification, nous avons identifié deux tâches principales qu'il est nécessaire de réaliser au préalable.

2. Notons que seules des règles ayant le même rt sont à fusionner.

Identification des mots composés : Les instances de syntagmes nominaux contenant plusieurs prépositions "de", comme dans "*lunettes de soleil de marque - détecteur de fumée de protection*", posent le problème du choix de la bonne préposition de séparation. Cela a une influence directe sur l'identification des termes A et B, et par conséquent sur le type de relation approprié à identifier. Nous avons procédé à l'identification de ces instances en vérifiant leur existence dans la base de connaissances. Dans "*lunettes de soleil de marque*", les candidats pour les termes A et B seraient "***(A :** lunettes, **B :** soleil de marque)*" et "***(A :** lunettes de soleil, **B :** marque)*". Dans le premier cas, l'inexistence du terme B dans *JDM* nous permet d'assigner les valeurs du dernier candidat à A et B. Dans le cas où les deux candidats résultent en des termes A et B connus, la préposition de séparation *prévue* [3] est désignée manuellement.

Entités nommées génériques : Les instances contenant des entités nommées telles que le prénom ou le nom de famille d'une personne ne sont bien représentées dans notre base de connaissances que lorsque ces entités sont renommées ou relève de connaissances communes/culture populaire (par exemple, "*Coca-Cola*" - "*Lucie*"). De ce fait, il est important de prendre en compte les instances inexistantes en effectuant des transformations qui remplacent le nom par un autre (du même type) que nous savons bien représenté dans la base de connaissances.

4.2 Algorithme d'application

Afin d'évaluer les couples de contraintes sémantiques produit par l'algorithme d'apprentissage, nous mettons en place un processus de validation qui cherche à vérifier la satisfaction de ces contraintes, l'objectif étant d'identifier les types sémantiques tirés de la portion du corpus n'ayant pas été impliquée dans le calcul des contraintes. Il s'agit donc de 450 exemples répartis équitablement sur les 15 types de relations possibles.

4.2.1 Critères de décision

En théorie, l'approche de validation des contraintes est basée sur une recherche de similarité des types sémantiques des termes du syntagme en entrée et les termes à partir desquels le système *GRASP*-it a été entraîné. L'idée est donc d'induire une *identité de type* [4] si les termes nouveaux sont suffisamment proches de la sémantique synthétisant les entrées du processus d'apprentissage. En pratique, la recherche est menée en calculant une similarité entre les termes A et B de l'entrée et la signature respective dans toutes les règles $< sL, sR, rt >_i$ apprises par *GRASP*-it. Les deux similarités obtenues (pour le terme A avec sL et B avec sR) sont agrégées par une moyenne arithmétique. Par conséquent, la similarité entre une forme « *A de B* » et une règle (paire de contraintes) $< sL, sR, rt >$ est donnée dans la formule 3.

$$1/2(sim(s(A), s_L) + sim(s(B), s_R)) \quad (3)$$

Il convient de noter qu'une signature pour un terme et une contrainte d'une règle partagent une structure identique. Une réponse positive est renvoyée pour le type le mieux classé en termes de valeurs de similarité moyenne avec chaque paire. Notons qu'afin que le classement puisse être fait, la procédure de vérification est menée une fois que les couples de contraintes sont calculés pour tous les types considérés.

Il reste possible d'avoir uniquement recourt aux règles qui sont soit le résultat d'une fusion, soit qui n'ont pas été fusionnées. Autrement dit, nous pourrions exclure les règles qui ont participé à la création d'une nouvelle règle. Nous prévoyons que cette réduction de l'ensemble de règles entraînerait un temps d'exécution plus court sans fortement dégrader les résultats. Cet aspect fait l'objet d'une évaluation (voir *Expérience* 3).

4.2.2 Traits extra-sémantiques

La détection de certains types dépend plus ou moins de marqueurs extra-sémantiques, tels que l'utilisation ou non de déterminant, l'utilisation d'entités nommées, ou encore la définitude des compléments de nom. Un exemple illustrant cette dépendance est le syntagme "*photo de famille*" par opposition à "*photo d'une famille*" qui, en raison de la présence ou non d'un déterminant, conditionne l'interprétation du lien sémantique (resp., *topic* et *dépiction*).

De telles heuristiques ne font pas partie de la composante principale de notre solution, tant nous souhaitons mettre en évidence l'implication des règles sémantiques *spécifiquement*. Néanmoins, étant donné que la représentation des termes (signature) consiste en un ensemble de symboles, la solution proposée semble également adaptée pour prendre en compte des informations extra-sémantiques. Cela revient à inclure les traits pertinents dans les signatures. Par conséquent, et afin de confirmer l'intuition de l'utilité de marqueurs extra-sémantiques, nous avons procédé à l'intégration du trait de définitude des compléments (B) à nos représentations. Cela constitue une étude distincte dans la section suivante [5] (voir *Expérience* 2).

4.3 Protocole d'évaluation

Dans cette évaluation, nous menons trois expériences distinctes, visant à évaluer les aspects suivants.

Expérience 1 : Nous cherchons à évaluer individuellement le gain en performance permis par chaque trait sémantique inclus dans les signatures. Dans cette expérience, nous prenons pour référence de base (*baseline*) l'inclusion des *hyperonymes* (*H*) uniquement. L'idée est de procéder de manière contrastive et d'analyser les cas que nous pourrions classer avec succès en ajoutant séparément les traits *TRT* et *SST*.

Expérience 2 : L'inclusion de marqueurs morphologiques pourrait se révéler bénéfique pour le processus de classification. Sans concevoir d'heuristiques élaborées *ad hoc* pour ces traits, nous expérimentons les effets induits

3. Celle permettant d'identifier des termes A et B ayant le type de relation annoté.

4. Supposer, par induction, que deux termes sont du même type sémantique

5. Une classification aboutie devra néanmoins faire appel à des règles *ad hoc*, éventuellement plus élaborées, traitant le cas des marqueurs relevant de la morphologie, par exemple.

par la construction des signatures avec des traits extra-sémantiques, à savoir le trait de *définitude* et la présence ou non d'un déterminant pour le terme B. Nous utilisons une approche symbolique simple expliquée dans 4.4.2.

Expérience 3 : Ici, nous nous intéressons à estimer la valeur globale d'efforts computationnels supplémentaires (efforts se traduisant par des coûts de calcul par exemple). Le système développé étant également destiné à être utilisé comme composant de tâches spécifiques de *TALN* (les plus critiques d'entre elles étant l'extraction de relations à partir de textes ou la résolution d'anaphores), il est important d'étudier la faisabilité d'intégrer *GRASP*-it dans de tels systèmes. Dans ces systèmes appliqués, les exigences pour les sous-tâches concernent souvent le temps d'exécution. Par conséquent, cette expérience est conçue pour étudier le gain/perte de performance par rapport au temps de calcul dans deux paramétrages différents.

Afin de maintenir une équivalence entre le nombre d'exemples pour chaque classe, nous ne considérons pas les cas de classification multiple dans cette évaluation. Compte tenu de la méthode de création du corpus, un seul type est associé à chaque exemple. Les cas pouvant être classés dans plus d'un type devront par conséquent être renseignés à la main. Indépendamment de la justesse des prédictions supplémentaires (fortuites/incidentelles), ces exemples n'étaient pas prévus comme appartenant à cette classe supplémentaire et ne sont donc pas comptés dans le nombre d'instances. De plus, le nombre de ces cas de classifications multiples n'est pas prévisible, ni uniformément réparti entre les types que nous considérons.

Selon les besoins d'exigence de l'évaluation, une alternative indulgente ne nécessitant pas d'annotations supplémentaires peut être envisagée. Il serait alors possible de considérer comme correctes non seulement les instances classifiées dans le type annoté dans le corpus de test, mais également celles pour lesquelles le type attendu se serait retrouvé en deuxième position (ou en position n) dans le classement des types par valeurs de similarités, si (et seulement si) sa valeur est proche de celle calculée pour le type le mieux classé. En d'autres termes, il s'agirait de considérer la sortie de l'algorithme comme une classification dans n classes différentes lorsque celles-ci auraient obtenu les n meilleures valeurs de similarité et que l'écart maximal entre les n meilleures valeurs aura été jugé négligeable (lorsque la n-ième meilleure valeur de similarité est à moins de 5% de la première, avec $n = 2$, par exemple, serait une prédiction d'appartenance dans 2 classes).

4.4 Scores de performance

Dans ce qui suit, étant donné un type de relation, nous considérons la précision (P) d'une classe comme la proportion d'exemples pour lesquels la classe est correctement prédite par rapport à toutes les instances prédites comme appartenant à cette classe. Le rappel (R) représente le rapport d'exemples pour lesquels la classe est correctement prédite à toutes les instances réelles de cette classe.

4.4.1 Expérience 1 - Traits sémantiques

Une approche basée uniquement sur la sémantique de A et B donne les résultats illustrés dans le Tableau 2. Afin d'évaluer séparément le gain de performance permis par chaque trait inclus dans la signature des termes, nous rapportons les résultats de 4 configurations contrastives des paramètres de *GRASP*-it.

Configuration	P (%)	R (%)	F1
H	67,3	65,9	0,653
H+SST	70,4	69,7	0,691
H+TRT	77,6	77	0,767
H+TRT+SST	**78**	**77,3**	**0,772**

TABLE 2 – Moyennes de précision P (%), rappel R (%), et score F1 permis par les différentes configurations de *GRASP*-it.

Les éléments combinés pour construire chacune des configurations représentent les informations stockées lors du calcul des signatures (donc, lors de l'apprentissage des règles) : Hyperonymes (*H*), Cible pour les types de relation (*TRT*), et Types sémantiques standard (*SST*).

Tout d'abord, en tant que *baseline*, les hyperonymes, étant des traits lexico-sémantiques, conduisent à un score *F1* moyen de $0,653$, ce qui est satisfaisant compte tenu de la rareté potentielle des hyperonymes de certains termes (par exemple, nous attirons l'attention sur les termes A des types *Caractérisation (Ca), Agent (A) et Patient (P)* étant tous des entités abstraites pour lesquelles il est délicat d'identifier un hyperonyme lexical). Deuxièmement, nous observons que les traits *SST* et *TRT* conduisent à des améliorations conséquentes, le gain non linéaire ayant naturellement tendance à devenir moins important à mesure que les scores s'améliorent. Les traits *H* et *TRT* d'une part et les *SST* d'autre part, sont complémentaires car ils répondent chacun à des besoins spécifiques de description. Les *SST* apportent la couverture des typologies standards (par exemple, en fournissant le type *_INFOSEM-THING-ABSTRACT* qui aide dans le scénario discuté ci-dessus), tandis que les *TRTs* et les *H* (étant des entrées terminologiques de nature lexicale) apportent une granularité plus fine rendue possible par l'abondance de la terminologie.

Avec un score *F1* de $0,772$, la configuration la plus favorable est celle combinant tous les traits sémantiques. Le Tableau 3 rapporte les résultats de la configuration *H+TRT+SST* pour chaque type de relation sémantique considéré.

Nous observons des résultats relativement élevés, bien que des disparités existent en fonction du type à identifier. En particulier, le faible rappel pour la relation *Caractérisation* (*Ca*) peut être attribué à sa représentation limitée dans la base de données (environ 10000 relations comparées à la relation *Holonymie* (H), qui en compte plus de 10 millions), entraînant une faible proportion d'exemples d'apprentissage correctement annotés. Il en va de même pour

Type	P	R	F1
Origine (O)	100	86	0,92
Lien social (LS)	83	100	0,91
Holonymie (H)	78	86	0,82
Quantification (Q)	82	80	0,81
Agent (A)	71	93	0,81
Dépiction (D)	88	73	0,80
Matière (M)	78	83	0,80
Instrument (I)	77	80	0,78
Lieu (L)	84	70	0,76
Topic (T)	68	86	0,76
Patient (P)	74	76	0,75
Auteur (AC)	76	73	0,74
Conséquence (Co)	76	63	0,69
Possession (Po)	65	63	0,64
Caracterisation (Ca)	71	50	0,59
Moyenne	**78**	**77,3**	**0,772**

TABLE 3 – Pourcentages (%) de Precision (P), Rappel (R), et scores F1 pour la meilleure configuration sémantique du système (*GRASP-it*$_{(H+TRT+SST)}$), pour chaque type de relation considéré.

les exemples de test (jusqu'à la moitié des cas). De plus, dans le cas de (*Ca*), les cas génériques sont également peu représentés et souvent associés à un sens du terme qui n'est pas une propriété (voir Section 4.5). Il est à noter qu'il y a un rappel maximal pour le type de *lien social* (*LS*) porté par des têtes nominales *relationnelles*, ce qui permet une interprétation lexicale de ce type et est bien synthétisé par les contraintes créées. Le type d'*Origine* (*O*) est précis en raison de son petit ensemble de règles générales (un grand nombre de règles ont pu être fusionnées), mais échoue pour les exemples qui ne sont pas bien représentés dans le corpus. Ce cas est intéressant car les instances du type *Origine* sont presque inexistantes dans *JDM* (29 relations), les hyperonymes (H) ayant facilité la synthèse de règles efficaces.

À bien des égards, le protocole suivi cherche à évaluer le modèle sans complaisance ; en effet, les scores devraient être interprétés comme une limite basse à améliorer à travers divers traitements de processus de classification utilisant les règles de *GRASP*-it. Il convient de noter l'absence d'heuristiques morpho-syntaxiques (extra-sémantiques). De plus, les instances qui ont été considérées comme erronées (selon le corpus) mais qui sont en réalité également valides pour le type prédit sont comptabilisés comme des erreurs. Une évaluation à classes multiples ferait évoluer les scores de chaque type à la hausse [6], elle nécessiterait néanmoins une annotation manuelle.

4.4.2 Expérience 2 - Définitude

Nous cherchons, en plus des traits sémantiques discutés précédemment, à *retenir* le patron morpho-syntaxique du syntagme à décrire afin de prendre en considération le caractère défini ou indéfini du complément de nom (B). Ceci se fait en incluant dans la signature du terme B deux symboles distincts correspondant à la présence (*resp.* absence) d'un déterminant défini ou indéfini (*Det*, *NoDet*, *Def*, *NoDef*). Un cas particulier concerne les entités nommées où, en dépit d'un *NoDet*, l'attribut *Def* est "*forcé*". Quelques exemples :

- *chat du rabbin* => *Det* + *Def*
- *écran de cinéma* => *NoDet* + *NoDef*
- *tableau de Chagall* => *NoDet* + *Def*

Configuration	P	R	F1
H+TRT+SST	78	77,3	0,772
H+TRT+SST+DEF	**80,3**	**79,8**	**0,795**
Origine (O)	100	90	0,95
Lien social (LS)	85	100	0,92
Holonymie (H)	81	100	0,90
Instrument (I)	88	80	0,84
Quantification (Q)	86	83	0,84
Matière (M)	83	83	0,83
Dépiction (D)	85	80	0,82
Lieu (L)	85	76	0,80
Agent (A)	67	96	0,79
Auteur (AC)	79	76	0,77
Topic (T)	70	86	0,77
Patient (P)	72	70	0,71
Conséquence (Co)	76	63	0,69
Possession (Po)	76	63	0,69
Caracterisation (Ca)	71	50	0,59

TABLE 4 – Comparaison des scores de performance entre la configuration exclusivement sémantique et celle incluant le trait de définitude (*GRASP-it*$_{(H+TRT+SST+\mathbf{DEF})}$), pour chaque type de relation sémantique considéré.

Tel que présenté dans le Tableau 4, prendre en compte le trait de définitude améliore le score F1 global (par rapport à la configuration exclusivement sémantique). Néanmoins, nous observons une certaine variabilité dans l'amélioration, et dans deux cas particuliers (agent et patient), une diminution des performances. La raison en est que le trait de définitude n'est pas typique d'un type unique, mais plutôt d'un sous-ensemble de types. Son inclusion aide lors de la décision entre deux types pour lesquels le trait de définitude est décisif. Au contraire, cela apporte une certaine confusion entre les types dans le même sous-ensemble de règles (pour lesquels la définitude n'est pas décisive).

4.4.3 Expérience 3 - Élagage des règles

Dans le tableau 5, nous mettons en évidence les différences en termes du nombre de règles appliquées (**#r**) et des temps d'exécution (T) pour les paramètres *Trim* et *No Trim*, qui correspondent respectivement à un ensemble complet de règles et à un ensemble réduit. L'ensemble de règles réduit contient uniquement les règles qui n'ont pas été utilisées pour une opération de fusion. Notons que (T) est la durée pour 450 instances de test.
Le temps d'exécution dépend linéairement du nombre de règles et de la taille des signatures. Plus il y a de fusions,

6. À titre d'exemple, le score F1 du type le moins bien classé (*Ca*) passerait à $0,76$

Configuartion	P	R	F1	#r	T (s)
Trim	79,6	77,6	0,77	49	***25.42***
No Trim	80,36	80	0,798	1384	*92.78*

TABLE 5 – Effets de la réduction de règles sur les scores de performance et les temps d'exécution.

plus les signatures sont longues avec toutefois une asymptote sur la longueur. La configuration d'élagage (*Trim*) conduit à une amélioration considérable du temps de calcul (temps divisé un peu moins de 4 fois) avec une légère diminution de la qualité. Cela signifie que, conformément à l'intuition de départ, les systèmes qui nécessitent des réponses dans des délais particulièrement courts pourraient bénéficier de la réduction de l'ensemble de règles sans subir une dégradation significative de la performance globale.

4.5 Analyse des cas d'échec

Pour rappel, ce travail fait également office d'outil de contrôle du contenu de la base de connaissances, dans le sens où les manques éventuels sont mis en évidence par les cas d'échec de classification de l'algorithme et permettent de consolider les types de relations appropriés.

Parmi les cas d'échec, nous rapportons qu'autour de 75% des occurrences sont directement dus à la polysémie du terme A et/ou du terme B (confondus). Le terme A dans *« richesse du royaume »* est considéré à tort dans le sens d'un *bien* > *objet* et non de la propriété d'*abondance*, ce qui a mené le système à sa classification (fausse) dans le type *lieu* (un objet pouvant se trouver dans un lieu). La relation sémantique correcte dans ce cas est plutôt *caractéristique*.

En écartant les cas de classes multiples (comme *« ombre d'un arbre »* ou encore *« travail du réalisateur »* classés respectivement dans *caracteristique* et *auteur*, et prédits par le système comme *dépiction* et *agent*), le reste des erreurs peuvent être expliquées par différentes raisons, parmi elles : *Les défauts de connaissances* (représentant des manques dans la base) comme dans *« restaurant de cuisine végétalienne »* (où le terme *B* n'était pas inclus dans *JDM*). Ensuite, quelques *défauts de compétence* (dûs au manque de traitements) expliquent un petit nombre de cas. En l'occurrence, la gestion de la dispersion des attributs sémantiques dans JDM à travers les différentes variantes morphologiques d'un terme : il arrive, pour des raisons intentionnelles/valables ou suite à un manque de connaissance, que certaines relations n'aient pas été propagées à toutes les dérivations d'un terme donné. En ce qui concerne notre système, ce défaut peut être observé particulièrement au niveau du trait *type sémantique standard* (SST). Dans « *liste de films* » devant être classé dans le type *Quantification*, les types standards du lemme *film* étaient manquants dans sa variante au pluriel. Bien que ce cas d'échec ait été utile en signalant l'état de la base en ce qui concerne ce type de relation, une phase de normalisation (en l'occurrence, une lemmatisation) aurait permis de contourner ce cas de dispersion.

L'analyse des cas d'échecs peut mener à l'identification d'un défaut de connaissance au niveau d'un exemple test ou d'un exemple d'apprentissage. Dans le second cas, l'impact sera plus important dans la mesure où il influencera négativement tous les exemples qui relèvent de ce type de relation.

5 Conclusion et perspectives

Nous avons présenté les résultats d'une recherche portant sur l'analyse automatique des formes *« A de B »*. Les contributions de ce travail ont consisté à définir une typologie non-exhaustive des relations sémantiques entre les termes *A* et *B*, puis à produire un petit corpus d'exemples annotés par ces types. Enfin, nous avons proposé un algorithme d'apprentissage de règles d'ordre sémantique sous la forme de couples de contraintes agrégés. L'objectif est d'une part, de réussir une classification automatique des types pouvant être utilisée lors de l'analyse sémantique de textes, et d'autre part, d'employer l'algorithme dans des démarches de consolidation de la base de connaissances utilisée dans le système présenté. On notera que le système proposé ne dépend pas d'une base de connaissances spécifique.

Les enseignements apportés par ce projet se définissent sur les plans théorique et appliqué. En premier lieu, nous identifions le défi que pose la possibilité d'inclusion des exemples dans plusieurs types différents. Ensuite, même si une identification du type repose fortement sur la sémantique, cette dernière ne constitue pas l'unique critère de décision de classification. Ceci signifie qu'un algorithme performant devra inclure une série de traitements, notamment de préparation des données, mais aussi d'analyse (identification de mots composés, gestion de la polysémie, normalisation des entrées, etc). En outre, comme pour toute construction de corpus, le problème de l'équilibre de la distribution et de la couvertures des traits/modèles se pose. Ce problème est d'autant plus marqué pour une collection relativement petite de données.

La question de la couverture est difficile : intuitivement on pourrait penser qu'il faut augmenter le nombre d'exemples. Cependant la couverture des types de relations dans les formes génitives suit une loi de puissance, c'est-à-dire qu'il y a un grand nombre de cas spécifiques dans la longue traîne. Ces cas sont difficilement calculables car ils peuvent correspondre à des formes figurées et sont du point de vue des signatures souvent des exemples non-prototypiques (des quasi hapax). De plus, les formes prototypiques correspondant au début de la distribution en loi de puissance sont souvent nombreux (par exemple, un nombre impressionnant de « <*animal*> *de* <*lieu*> »). Multiplier les exemples d'apprentissage ayant quasiment les mêmes signatures (représentés par la même règle *modèle*) n'a que peut d'effet, sur le long terme, sur la qualité de l'apprentissage. De surcroît, étant donné la contrainte de coût de calcul posée par le besoin d'intégrer la solution dans des systèmes appliqués, il serait contre-productif de rajouter des exemples relevant d'un modèle déjà connu. On peut cependant envisager une

stratégie qui exploiterait un critère de fusion (règle fusionnable ou pas) via un apprentissage incrémental.

L'approche symbolique adoptée présente l'avantage d'être facilement explicable, ses résultats dépendent aussi bien des types considérés, de la richesse de la connaissance du monde utilisée, que de la qualité des données (justesse et pertinence mais également l'équilibre *signaux faibles/forts* qui sont autant d'enjeux de représentation des cas discutés ci-dessus).

Le fait que les règles ne soient pas mutuellement exclusives (et dans certains cas quasi-identiques) représente une difficulté qui ne pourrait être levée, - même pour un locuteur humain -, à moins de disposer d'un contexte suffisant, par exemple : *présentation de l'élève - portrait de Van Gogh - travail du ciment*, pouvant aussi bien être classés dans les types : **[***Agent (A)* et/ou *Patient (P)***]** - **[***Agent (A)* et/ou *Dépiction (D)***]** - **[***Agent (A)* et/ou *Patient (P)***]**, resp.).

Étant donné la complexité et la difficulté inhérentes à de nombreux cas, la question de la détection automatique des relations sémantiques dans ce type de syntagmes reste ouverte et sujette à une exploration continue. En perspective de ce travail, nous pouvons évoquer les tâches suivantes :

— Enrichissement du corpus par l'application de notre algorithme : Une collection plus grande permettrait d'employer des méthodes diverses notamment celles nécessitant de très grands nombres d'instances telles que des approches d'apprentissage neuronal. L'idée ici est de générer un ensemble plus large d'exemples à partir des termes de la base de connaissances qui sont liés par les relations appropriées. Le résultat devra être validé manuellement afin de garantir, d'une part, la correction du syntagme nominal généré, et d'autre part, sa pertinence. Produire "*atome du garçon*" pour la classe de *holonymie* ou de *composition*, bien que correct en théorie, ne semble pas très pertinent. Les annotations de méta-informations relevant de la *pertinence* des relations dans *JDM* pourraient aider dans cette démarche ;

— Extension de la typologie des relations sémantiques considérées dans le processus d'apprentissage à davantage de types, et leur hiérarchisation afin de permettre des exploitations plus ou moins précises qui répondraient aux exigences des diverses ressources et outils pouvant être utilisés ;

Distribution

Le corpus ainsi que l'algorithme sont accessibles sur le lien en bas de page [7]. À des fins de démonstration et d'expérimentation, un exemple d'implémentation de *GRASP*-it est disponible. Plusieurs paramètres sont proposés, notamment l'inclusion ou non de chacun des traits discutés dans cet article (*H*, *TRT*, *SST* et *DEF*), la réduction des règles et le réglage du seuil de fusion sont également possibles. L'entraînement et le test peuvent être effectués aussi bien avec le corpus proposé, qu'avec une autre collection de données. De plus, il n'est pas nécessaire que les types sémantiques soient restreints à ceux définis dans cette étude (plus ou moins de types peuvent être définis et utilisés sur le démonstrateur web). Cependant, étant donné que les signatures construites pour cette implémentation sont basées sur notre base de connaissances et sa structure (la structure JDM), nous devons noter que si une autre collection de données est utilisée, il est nécessaire de s'assurer que les exemples sont conformes aux exigences discutées dans la section *Préparation des données* 4.1.

7. https ://www.jeuxdemots.org/rezo-GEN1.php

Références

[1] C. Barker, C. Maienborn, K. Von-Heusinger, P. Portner. Possessives and relational nouns. *Semantics-noun phrases and verb phrases*, pp. 177-203, 2019.

[2] S. Löbner. Concept types and determination. *Journal of semantics*, pp. 279–333, 2011.

[3] J. De Bruin and R. Scha. The interpretation of relational nouns. *26th Annual Meeting of the Association for Computational Linguistics*, pp. 25–32, 1988.

[4] A. Ben Abacha. Recherche de réponses précises à des questions médicales : le système de questions-réponses MEANS. *PhD thesis*, Paris 11, 2012.

[5] H. Guenoune. Résolution des anaphores dans la communication électronique médiée : heuristiques et apport d'informations de sens commun. *PhD thesis*, Université de Montpellier, 2022.

[6] P. Kapanipathi, I. Abdelaziz, S. Ravishankar, S. Roukos, A. Gray, R. Astudillo, M. Chang, C. Cornelio, S. Dana, A. Fokoue. Leveraging abstract meaning representation for knowledge base question answering. *Association for Computational Linguistics*, 2021.

[7] V. Nastase and S. Szpakowicz. Exploring noun-modifier semantic relations. *Fifth international workshop on computational semantics (IWCS-5)*, pp. 285–301, 2003.

[8] M. Lafourcade and N. Le Brun. Apport du jeu pour la construction de connaissances : le projet JeuxDeMots. *Technologie et innovation*, Vol. 8, pp. 4, 2023.

[9] M. Lafourcade. Making people play for Lexical Acquisition with the JeuxDeMots prototype. *SNLP'07 : 7th international symposium on natural language processing*, p 7, 2007.

[10] F. Landragin. Étude de la référence et de la coréférence : rôle des petits corpus et observations à partir du corpus MC4. Bases, corpus et langage-UMR 6039. *Corpus*,Vol. 2, pp.18 , 2018.

SemOntoMap : une méthode hybride pour l'annotation sémantique de textes cliniques en psychiatrie

O. Aouina[1], J. Hilbey[1,2], J. Charlet[1,2]

[1] Sorbonne Université, Inserm, Université Sorbonne Paris-Nord, Laboratoire d'informatique médicale et d'ingénierie des connaissances en e-santé, LIMICS, Paris, France
[2] Assistance Publique-Hôpitaux de Paris, Paris, France

ons.aouina@etu.sorbonne-universite.fr

Résumé

Les descriptions en texte libre contenues dans les dossiers patients informatisés (DPI) revêtent un intérêt significatif pour la recherche clinique et l'optimisation des soins. Toutefois, la capacité des ordinateurs à interpréter directement ce texte libre est limitée, réduisant ainsi sa valeur potentielle. Bien que l'annotation sémantique offre une solution pour rendre le texte libre des DPI interprétable par les machines, elle rencontre des obstacles majeurs lorsqu'elle est appliquée aux ontologies de domaine spécifiques, particulièrement en français. Ces difficultés sont encore plus marquées dans le domaine psychiatrique où l'on cherche non seulement à extraire les concepts du domaine mais à les normaliser et à extraire les relations de textes décrivant longuement l'histoire d'une maladie d'un patient et ses ascendants. Face à ces enjeux, nous proposons un système fondé sur des techniques d'apprentissage non supervisé pour extraire les entités et leurs interrelations en utilisant une ontologie de domaine. Ce système est évalué dans le cadre du projet PsyCARE sur un échantillon de 60 comptes rendus analysés par deux évaluateurs.

Mots-clés

annotation sémantique, ontologie, plongement de l'ontologie, apprentissage automatique non supervisé, TALN, BERT, Word2Vec

Abstract

The free-text descriptions contained in Electronic Health Records (EHR) hold significant interest for clinical research and the optimization of care. However, computers' ability to directly interpret this free text is limited, thereby reducing its potential value. While semantic annotation offers a solution to make the free text of EHRs machine-interpretable, it faces major obstacles when applied to specific domain ontologies, particularly in French. These difficulties are even more pronounced in the psychiatric field, where there is an attempt to extract domain concepts relations from texts that extensively describe a patient's disease history and their ancestors. Faced with these challenges, we propose a system based on unsupervised learning techniques to extract entities and their interrelations using a domain ontology. This system is evaluated within the framework of the PsyCARE project on a sample of 60 reports analyzed by two evaluators.

Keywords

Semantic annotation, ontology embedding, unsupervised machine learning, Ontology, NLP, BERT, Word2Vec

1 Introduction

Dans le domaine de la recherche biomédicale et des soins aux patients, les documents techniques, et plus spécifiquement les textes biomédicaux, sont cruciaux. Ces documents, sont indispensables pour faire avancer les pratiques cliniques et stimuler l'innovation en santé. La complexité et la richesse de ces textes ont mené à l'utilisation de diverses ontologies biomédicales, telles que l'Ontologie des Gènes (GO) et la Nomenclature Systématisée de Médecine – Termes Cliniques (SNOMED CT), marquant des efforts significatifs pour structurer cette information vitale et améliorer son accessibilité [34]. L'annotation sémantique, qui associe le texte à des balises significatives issues de ces ontologies, joue un rôle clé dans de nombreuses applications, renforçant l'interopérabilité et l'efficacité de la récupération d'informations [27].

L'effort d'annotation sémantique varie du manuel au totalement automatisé, exploitant les avancées en traitement automatique du langage naturel (TALN) [25]. Notre étude se concentre sur l'annotation automatique de textes cliniques, tâche rendue complexe par la nature du langage médical. Une attention particulière est portée aux sections narratives des dossiers cliniques, surtout dans les résumés de sortie en psychiatrie (dans notre cas des comptes rendus d'hospitalisation ou CRH), qui recèlent des informations sur les événements cliniquement significatifs affectant la trajectoire médicale du patient. Ces informations incluent les antécédents familiaux, l'historique de la maladie, les traitements prescrits, ainsi que les relations temporelles entre ces événements. Des questions comme « comment la maladie a évolué chez le patient ? » ne peuvent être interprétées et on ne peut y répondre que si l'on prend en compte le contexte complet des antécédents du patient et les relations temporelles entre les différents concepts repérés. Ce problème est

abordé en psychiatrie par le projet RHU PsyCARE [1] qui vise à améliorer l'intervention précoce dans la psychose en fournissant des outils pour faciliter l'accès aux soins et offrir des programmes de traitement personnalisés.
Notre travail vise donc à annoter sémantiquement des CRH, en capturant les segments textuels qui correspondent à des ontologies ou des terminologies standardisées mais aussi en déchiffrant les modalités, les relations temporelles et les informations détaillées sur les antécédents et l'évolution de la psychose. Dans cet article, nous proposons une méthode d'annotation sémantique des CRH fondée d'abord sur une ontologie développée dans le cadre de PsyCARE. Cette ontologie est combinée avec des modèles de langue et des algorithmes d'apprentissage pour construire un modèle formel précis du texte [15].

2 Contexte

Les CRH sont rédigés par des professionnels de la santé divers et aux styles d'écriture variés. Le traitement efficace de ces documents est essentiel pour ces derniers qui doivent parcourir d'importants volumes de dossiers médicaux électroniques pour dégager les informations clés. La normalisation des entités nommées joue un rôle crucial dans la réduction de l'ambiguïté des comptes rendus cliniques. Néanmoins, ces étapes peuvent aboutir à l'extraction de concepts redondants ou de faible valeur informative. Face à ce défi, le développement des méthodes d'extraction d'information (EI) non supervisées devient une évidence. Ces algorithmes permettent d'identifier des informations significatives sans dépendre des corpus préalablement annotés, offrant ainsi une réponse efficace aux contraintes des approches traditionnelles [22].
Parallèlement, la normalisation des entités, également connue sous les termes de désambiguïsation ou de liaison d'entités, joue un rôle crucial dans l'extraction d'informations. Cette démarche consiste à associer les mentions d'entités présentes dans le texte avec des catégories ou des concepts issus d'un vocabulaire de référence [28] ou d'une ontologie spécifique, ce qui permet d'uniformiser la représentation de ces mentions. Pour améliorer l'efficacité de la normalisation des entités, certaines recherches proposent d'intégrer des données concernant la structure des graphes de connaissances [24], tandis que d'autres études mettent en avant les bénéfices de combiner les plongements de mots et d'entités afin de créer des connexions significatives entre les entités. Ces approches visent à renforcer les performances de cette tâche en exploitant les relations sémantiques profondes [24]. Dans ce contexte, l'*embedding* (ou plongement) d'ontologies représente un domaine de recherche prometteur. P. Devkota *et al.* [9] montre que l'intégration d'informations issues du plongement d'ontologies peut significativement affiner la détection des concepts ontologiques dans la littérature scientifique, renforçant ainsi la concordance sémantique entre les informations textuelles et les structures ontologiques [3].
Dans cette section, nous explorons les techniques d'EI qui, dans notre contexte, concernent l'extraction de syntagmes, y compris les syntagmes nominaux (SN) et les syntagmes verbaux (SV) ainsi que le plongement d'ontologies pour structurer et enrichir les connaissances extraites.

1. https://psy-care.fr/

2.1 Extraction de syntagmes

Pour l'extraction de syntagmes, nous adoptons l'hypothèse selon laquelle les syntagmes correspondent à une liste de N-grammes, soit des séquences de n mots manifestant une structuration grammaticale particulière. Cette tâche consiste à déterminer un ensemble de séquences de mots qui encapsulent les thèmes centraux ou les idées présentées dans un document, offrant un aperçu de son contenu le plus critique. Ces algorithmes sont classés en méthodes supervisées [32] et non supervisées [31]. Compte tenu de la polyvalence et de l'applicabilité générale des méthodes non supervisées, se concentrant sur les attributs inhérents du texte pour l'extraction de syntagmes, notre proposition se concentre sur l'extraction non supervisée. Trois méthodes principales se distinguent dans ce domaine :

Méthodes fondées sur les Graphes. Ces méthodes convertissent le document en un graphe et classent les phrases candidates dans le graphe [21, 31] . Les nœuds correspondent à des éléments textuels tels que les mots ou les phrases, et les arêtes reflètent les liens entre eux, par exemple la co-occurrence ou la similarité sémantique. Cette approche permet d'évaluer l'importance des phrases candidates en fonction de leur position et de leurs connexions au sein du graphe. En exploitant les relations contextuelles entre les éléments textuels, ces méthodes se distinguent par leur capacité à identifier avec précision les syntagmes clés qui sont directement liés aux thèmes centraux du document. Ainsi, elles améliorent la pertinence et l'efficacité des systèmes de récupération d'information en facilitant l'identification de syntagmes essentiels qui récapitulent de manière efficace le contenu central du texte.

Méthodes Statistiques. Ces méthodes, telles que YAKE [5], sont fondées sur TF-IDF (Fréquence du Terme - Inverse de la Fréquence des Documents), TextRank [21] ou SingleRank [37]. Ces méthodes analysent les propriétés de distribution des mots et des phrases dans un texte par rapport au corpus de travail pour identifier des phrases clés et mettre en évidence des termes qui sont spécifiques et informatifs du contenu du document. L'importance de combiner des analyses statistiques avec des informations contextuelles est spécialement mise en avant par YAKE qui se distingue par son utilisation de métriques statistiques avancées pour saisir le contexte et la dispersion des termes à travers le document. Mais bien que ces méthodes soient efficaces d'un point de vue computationnel et simples à mettre en œuvre, elles ne capturent pas toujours la richesse sémantique du texte, limitant potentiellement leur efficacité dans certains contextes de récupération d'informations.

Méthodes fondées sur l'apprentissage profond. Ces méthodes exploitent les réseaux neuronaux pour apprendre des représentations du texte qui capturent des relations et des motifs sémantiques. Des approches non supervisées

d'apprentissage profond telles que les auto-encodeurs ou les modèles fondés sur les transformateurs comme BERT [10], peuvent modéliser implicitement l'importance des syntagmes. Parmi celles-ci, KeyBERT [12] tire parti des modèles tels BERT, pour identifier de manière efficace les syntagmes clés dans les textes. KeyBERT combine la capacité des transformateurs à comprendre le contexte profond du texte avec une approche ciblée pour l'extraction des phrases clés, permettant ainsi une identification précise et contextuellement riche des informations clés contenues dans les documents. PatternRank [26] s'appuie sur des modèles de langage et des parties de discours (PoS) pré-entraînés pour l'extraction non supervisée de phrases-clés à partir de documents uniques. Cet algorithme représente l'état de l'art dans l'extraction de phrases clés, grâce à son intégration de modèles de partie du discours pour la sélection des phrases candidates, permettant ainsi son adaptation à divers domaines. Cette approche permet une granularité et une précision accrues dans l'extraction des phrases clés, en se basant sur des critères syntaxiques spécifiques pour identifier les éléments les plus informatifs du texte. Dans notre approche, nous avons adapté PatternRank pour améliorer sa capacité à capturer des syntagmes nominaux et verbaux, en exploitant le *part of speech*, augmentant la précision de notre méthode.

2.2 Plongement des ontologies

Les modèles de plongement de graphe de connaissances (Knowledge Graph Embedding ou KGE) sont utilisés pour la transformation des vastes réseaux complexes d'entités et de relations au sein d'un graphe de connaissances en des espaces vectoriels de faible dimension, ainsi gérables [8]. L'essence du KGE réside dans sa capacité à transformer des informations complexes et de haute dimension d'un graphe de connaissances – comprenant diverses entités et les relations à facettes multiples entre elles – en une forme à la fois efficace sur le plan du calcul et sémantiquement riche.

Plusieurs modèles pour KGE, tels que DistMult [39] et RotatE [33], ont été proposés pour relever ces défis, montrant de bons résultats sur des ensembles de données de graphes de connaissances à usage général comme FB15K-237 [35]. Cependant, leur efficacité dans des domaines spécialisés, tels que la médecine, peut ne pas être aussi satisfaisante en raison de difficultés liées à la représentation et au raisonnement autour des entités et relations médicales [11]. Les méthodes existantes ne capturent pas adéquatement les relations complexes, les structures hiérarchiques, et l'hétérogénéité des entités médicales, ni n'abordent les problèmes de données bruyantes, incomplètes et la haute dimensionnalité souvent rencontrés dans les graphes de connaissances médicales.

Dans ce contexte, le plongement d'ontologies se présente comme une approche prometteuse, complétant le KGE. Axée sur la modélisation des relations directes entre entités, le plongement d'ontologies utilise la richesse sémantique et la structure logique des ontologies [19] . Cette méthode permet de capturer non seulement les relations entre entités mais aussi les concepts abstraits, les hiérarchies de classes et les axiomes qui structurent les connaissances dans un domaine spécifique.

L'intégration des techniques de prédiction par apprentissage automatique et d'analyse statistique des ontologies gagne en popularité et des méthodes pour plonger la sémantique des ontologies OWL commencent à émerger dans la littérature. Contrairement aux graphes de connaissances, les ontologies OWL ne se limitent pas à une structure graphique mais incorporent également des constructeurs logiques, et les entités sont souvent enrichies d'informations lexicales détaillées, spécifiées via $rdfs{:}label$, $rdfs{:}comment$ et de nombreuses autres propriétés d'annotation personnalisées ou intégrées. Dans cette approche, le but du plongement d'ontologie OWL est de représenter chaque entité nommée OWL (classe, instance ou propriété) par un vecteur, de manière à conserver dans l'espace vectoriel les relations inter-entités indiquées par les informations mentionnées ci-dessus et à maximiser la performance des tâches en aval où les vecteurs d'entrée peuvent être considérés comme des caractéristiques apprises.

EL Embedding [18] et Quantum Embedding [16] sont deux algorithmes de plongement d'ontologie OWL. Ils élaborent des fonctions de score et des fonctions de perte spécifiques pour les axiomes logiques issus respectivement d'EL++ et d'ALC, en transformant les relations logiques en relations géométriques. Cela encode la sémantique des constructeurs logiques mais néglige la sémantique supplémentaire apportée par les informations lexicales de l'ontologie. De plus, bien que la structure graphique soit explorée en considérant les axiomes de sous-classement et d'appartenance à une classe, l'exploration reste incomplète car elle se limite uniquement aux arêtes $rdfs{:}subClassOf$ et $rdf{:}type$ et ignore les arêtes impliquant d'autres relations.

Onto2Vec [29] et OPA2Vec [30] sont deux algorithmes de plongement d'ontologie utilisant le paradigme du plongement de mots, fondés sur l'architecture skip-gram ou CBOW. Onto2Vec utilise les axiomes d'une ontologie comme corpus pour l'entraînement, tandis qu'OPA2Vec enrichit le corpus d'Onto2Vec avec les informations lexicales fournies par, par exemple, $rdfs{:}comment$. Les deux méthodes adoptent la fermeture déductive d'une ontologie avec un raisonnement par inférence. Les deux méthodes traitent chaque axiome comme une phrase, ce qui signifie qu'elles ne peuvent pas explorer la corrélation entre les axiomes. Cela rend difficile l'exploration complète du graphe et de la relation logique entre les axiomes, et peut également conduire à un problème de pénurie de corpus pour les ontologies de petite à moyenne échelle.

OWL2Vec* [6] propose une solution aux limitations des approches précédentes en enrichissant leur corpus d'axiomes avec des données générées par des parcours sur des graphes RDF issus de la transformation des ontologies OWL. Cette approche prend en compte à la fois le graphe et les constructeurs logiques de l'ontologie. En outre, OWL2Vec* maximise l'exploitation des informations lexicales en créant des plongements non seulement pour les entités de l'ontologie mais également pour les termes lexicaux. Ainsi, OWL2Vec* condense efficacement les informations séman-

tiques et structurelles d'une ontologie dans un espace vectoriel compact, facilitant l'utilisation de ces données par des algorithmes d'apprentissage automatique pour des tâches en aval.

Le cadre d'OWL2Vec* est structuré autour de deux étapes clés comme illustré dans la figure 1 : (i) l'extraction d'un corpus à partir de l'ontologie, et (ii) l'entraînement d'un modèle de plongement de mots avec ce corpus. Ce corpus se compose de trois documents distincts : un document de structure, un document lexical, et un document combiné. Les deux premiers documents sont conçus pour explorer la structure de l'ontologie, ses constructeurs logiques et ses informations lexicales, permettant ainsi l'activation du raisonnement par inférence. Le troisième document vise à maintenir la corrélation entre les entités (IRIs) et leurs étiquettes lexicales (mots), en utilisant le premier document comme base tout en intégrant les informations lexicales disponibles de l'ontologie.

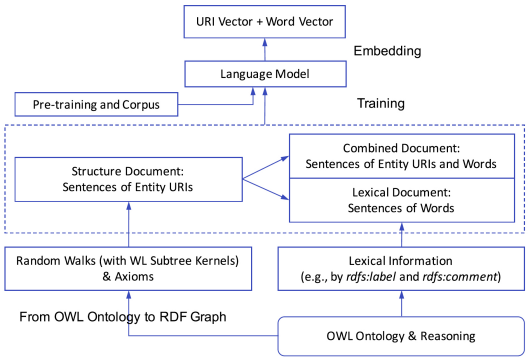


FIGURE 1 – Le contexte général d'OWL2Vec* Source : [6]

3 Système d'annotation

Dans cette section, nous présentons l'architecture du système, SemOntoMap, conçu pour enrichir les CRH de psychiatrie avec des annotations sémantiques. Notre approche s'appuie sur un corpus de textes de psychiatrie non annotés et une ontologie dédiée à ce domaine, visant à structurer ces documents.

Comme le montre la figure 2, la tâche d'annotation sémantique se déroule en trois grandes étapes : le prétraitement des textes, l'identification et la normalisation des entités nommées et l'extraction des relations entre ces entités.

3.1 Jeu de données en psychiatrie

Les documents cliniques exploités dans cette étude sont une compilation de près de 8000 CRH s'étendant sur une période de dix ans, totalisant environ 3,5 millions de mots. Ces CRH ont été collectés au sein du Groupe Hospitalier Universitaire Psychiatrie et Neurosciences de Paris. Ils sont semi-standardisés, en format Word et ont été pseudo-anonymisés au préalable, en remplaçant tous les noms, dates, lieux, etc. Chaque document se conclut par un diagnostic formulé selon la Classification Internationale des Maladies, version 10 (CIM-10 [2]). Rédigés en français, ces comptes rendus fournissent un aperçu détaillé de l'historique et du contexte social des patients, des prescriptions médicamenteuses, des circonstances d'admission à l'hôpital ainsi que les diagnostics psychiatriques actuels et antérieurs. Pour les besoins de l'annotation sémantique, une sélection de 30 comptes rendus a été annotée d'une façon aléatoire, en se basant sur le code CIM-10. Cette méthode de sélection vise à garantir une diversité dans les cas cliniques étudiés, couvrant un large éventail de diagnostics psychiatriques.

3.2 Ontologie de domaine

L'ontologie utilisée dans le processus d'annotation, appelée par la suite OntoPSY est une version fusionnée des modules ontoDOPSY, ontoMEDPSY, ontoDOME et ontoPOF de l'ontologie développée dans le cadre de PsyCARE [3] pour l'intégration des données et leur annotation sémantique. Ces modules contiennent les branches d'intérêt tels que les aspects cliniques psychiatriques (signes, symptômes, troubles psychiatriques), les médicaments identifiés par leur code ATC, des éléments relatifs à l'imagerie ainsi qu'une dimension temporelle pour représenter les connaissances médicales de manière adéquate. À partir de cette base, un schéma d'annotation est construit. Une branche de l'ontologie dédiée à la structure des CRH est ajoutée pour lier les concepts à leur contexte d'apparition dans le document, c'est-à-dire la section dans laquelle ils sont repérés [13] (« Histoire de la maladie », « Traitement de sortie », etc.) . En plus de décrire les aspects cliniques et les entités médicales, l'ontologie détaille les relations entre les différents concepts, enrichissant ainsi notre compréhension des interactions et des liens au sein des données cliniques.

3.3 Prétraitement des données textuelles

Ce processus implique le traitement du format du document et l'extraction de segments textuels pertinents à partir du document source, tout en écartant les balises et les éléments non pertinents. À ce stade, une analyse TALN de base est réalisée, incluant la tokenisation, la normalisation, et le marquage morphosyntaxique (*part-of-speech tagging*, POS). Le produit de cette phase est un texte brut enrichi de certaines annotations. Les algorithmes employés lors de cette étape ont été décrits dans un article antérieur [1].

3.4 Reconnaissance d'entités et normalisation

Dans cette section, nous détaillons les différentes étapes consacrées à l'extraction des candidats pour la reconnaissance des entités nommées (REN) ainsi qu'à leur normalisation en concordance avec les concepts de l'ontologie.

3.4.1 Extraction de syntagmes

L'importance des SN dans l'analyse des textes médicaux et psychiatriques est soulignée par les travaux de chercheurs comme Liu et al. [14] qui mettent en évidence

2. https://icd.who.int/browse10/2019/en
3. Cette ontologie sert à plus de processus que le seul TALN ; elle sert en particulier de modèle d'interopérabilité général pour le projet [13].

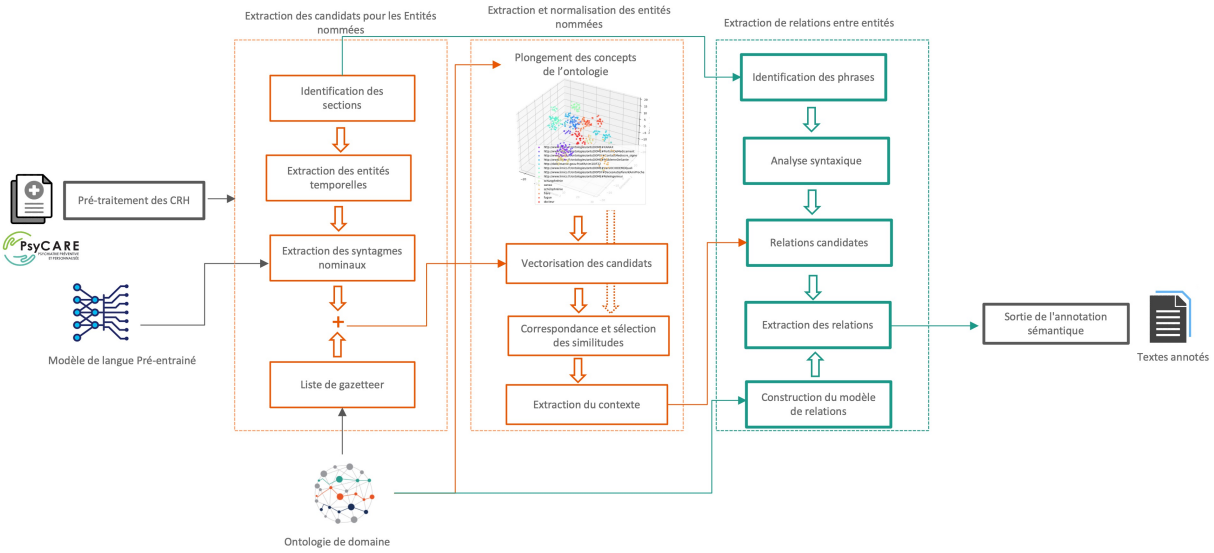


FIGURE 2 – Architecture du système d'annotation sémantique proposé.

la valeur de l'identification précise des termes médicaux pour améliorer l'accès à l'information dans les documents cliniques. Comme mentionné précédement, nous modifions PatternRank (Cf. sec. 2.1) pour l'adapter à la complexité narrative de ces documents. Les étapes sont détaillées dans la figure 3. Cette approche implique la segmentation du texte, le marquage syntaxique (POS), et la sélection des syntagmes qui répondent à des critères spécifiques (e.g., <(NN.|JJ|ADV)+><NN.> pour identifier des séquences commençant par un nom, adjectif, ou adverbe suivis de noms). Par la suite, les similarités cosinus entre la représentation du document et les représentations des syntagmes candidats sont calculées et ces derniers sont classées en ordre décroissant en fonction des scores trouvés. Les candidats sont à la fin transformés en vecteurs et classés par similarité cosinus avec le document pour extraire les termes les plus significatifs.

Dans nos expériences, nous utilisons le modèle de langage pré-entraîné Sentence-CamemBERT-Large développé par La Javaness [4]. Il s'agit d'un modèle SBERT qui a démontré sa capacité à produire de bonnes représentations textuelles pour des tâches de similarité sémantique.

3.4.2 Extraction d'information temporelles, medicaments et dosage

L'extraction des informations temporelles, médicamenteuses et de dosages est essentielle pour le suivi clinique et l'évaluation des traitements des patients. De nombreux travaux se sont concentrés sur ces extractions, notamment en ce qui concerne la temporalité et dans le domaine plus large du TALN médical [4]. Nous avons adopté la solution proposée par Aumiller Dennis [2] pour la reconnaissance et la normalisation des expressions temporelles, en combinant les capacités des librairies HeidelTime et SUTime pour l'identification complète des expressions temporelles dans les textes, car elles couvrent dates, heures, fréquences, et durées, et permettent l'ajustement de la date de référence pour une interprétation contextuelle. HeidelTime est utilisée pour son efficacité dans l'extraction temporelle de narrations non cliniques, adapté ici aux contextes cliniques. SUTime, en complément, offre la flexibilité d'une date de référence, utile pour notre analyse documentaire. Nous intégrons également le Temporal Tagger Service pour une détection précise de ces informations temporelles. Pour l'extraction des mentions de médicaments dans les comptes rendus hospitaliers en psychiatrie, EDS-NLP [36], développé par l'AP-HP, est employé pour sa spécialisation dans le traitement des données de santé en français. Enfin, le *GATE Tagger* [5] est utilisé pour identifier dosages et unités, facilitant l'interprétation des prescriptions.

3.5 Correspondance entre les informations extraites et les concepts de l'ontologie

Plongement de OntoPSY. Afin de produire le plongement sémantique de l'ontologie OntoPSY, nous avons mis en œuvre l'outil OWL2Vec* (voir Section 2.2). Cet outil a été configuré pour se servir d'un modèle Word2Vec préalablement entraîné sur un corpus diversifié, comprenant des articles de Wikipédia en français, des textes biomédicaux, ainsi que des corpus spécialisés [17]. Le modèle a ensuite été finement ajusté pour s'aligner avec les spécificités de l'ontologie, dont le prétraitement a été détaillé dans une publication antérieure [1]. Le réglage fin du modèle avec le corpus de l'ontologie a été réalisé à travers des marches aléatoires d'une profondeur de trois, permettant une exploration approfondie de la granularité de l'ontologie. Réalisée sur une série de 100 itérations, ce réglage a utilisé la même stratégie de marche aléatoire pour garantir une compréhen-

4. https://huggingface.co/dangvantuan/sentence-camembert-large

5. https://github.com/GateNLP/gateplugin-Tagger_Measurements

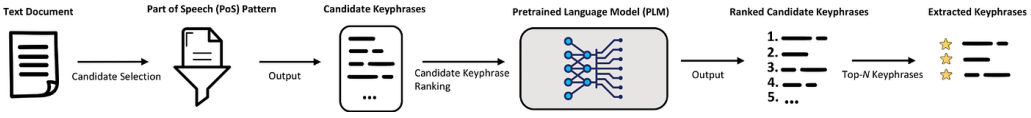


FIGURE 3 – Schéma du processus d'extraction non supervisée des syntagmes en utilisant PatternRank. Source : [26].

sion complète et adéquate de l'ontologie.

Vectorisation des candidats. Chaque syntagme extrait est transformé en un vecteur en utilisant la sortie de OWL2vec*. Cette étape de vectorisation permet leur représentation dans le même espace vectoriel que les axiomes de l'ontologie, facilitant ainsi les comparaisons sémantiques directes entre ces syntagmes et les concepts de l'ontologie.

Correspondance et sélection des similarités. Pour chaque vecteur de syntagmes, nous déterminons les dix concepts ontologiques les plus proches, classés selon leurs scores de similarité sémantique. Pour affiner cette sélection initiale et identifier avec précision le concept adéquat parmi les premiers candidats, nous employons un module de reclassement décrit plus en détail dans cet article [16], lequel se fonde sur une analyse syntaxique poussée. Ce module, exploite l'analyse syntaxique pour distinguer le concept le plus pertinent parmi les options pré-sélectionnées, en se fondant sur les scores de similarité issus de notre modèle de plongement. Le cœur de notre innovation réside dans l'utilisation de l'analyseur syntaxique de SpaCy [6], un outil conçu pour isoler le mot ou le syntagme le plus significatif au sein d'une phrase. En analysant la structure grammaticale du syntagme, le module de reclassement peut identifier avec précision l'entité principale, permettant ainsi une correspondance plus exacte entre le syntagme analysé et le concept de l'ontologie pertinent.

3.6 Extraction non supervisée de relations

Dans le contexte de l'analyse des CRH, l'extraction de relations (ER) constitue une étape cruciale du TALN. Cette tâche vise à identifier et à définir les liens sémantiques existant entre les entités nommées détectées dans le texte. Il existe deux principales techniques d'extraction de relations entre entités : les méthodes fondées sur des règles de modèles (*template rule-based*) et les méthodes fondées sur des vecteurs propres (*eigenvector-based*).

Dans la méthode fondée sur des règles, les caractéristiques linguistiques des relations entre entités sont d'abord organisées par des linguistes. Ensuite, les règles sont compilées [23], enfin, les relations entre entités sont extraites à travers une correspondance fondée sur ces règles.

Les méthodes fondées sur des vecteurs propres peuvent être divisées en deux types : l'apprentissage automatique traditionnel et l'apprentissage profond [38]. Pour répondre à nos besoins, nous utilisons une combinaison des méthodes décrites ci-dessus. Nous combinons l'analyse syntaxique des dépendances et la struture de l'ontologie pour identifier et classer les relations entre les entités.

6. https://spacy.io/models/fr

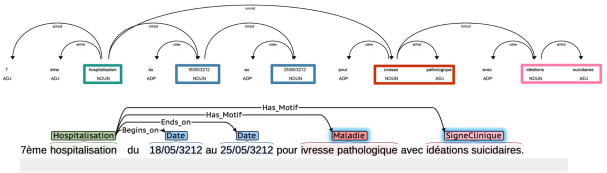


FIGURE 4 – Analyse Syntaxique et Structuration de Texte : Panneau Supérieur - Arbre d'Analyse Syntaxique avec Positions de Mots et Types de Dépendances ; Panneau Inférieur - Regroupement en SN et Identification de hospitalisation/date et hospitalisation/Maladie/SigneClinique.

3.6.1 Définition de la Tâche

À ce stade du processus d'annotation, les informations extraites sont liées aux concepts de l'ontologie OntoPSY. La figure 2 décrit l'architecture du processus d'extraction de relations (représenté par le rectangle vert en pointillés). Notre approche se décompose en quatre phases principales que nous détaillons dans cette section. Nous ciblons spécifiquement l'extraction de 14 relations que nous regroupons en cinq catégories : a) les relations temporelles, qui articulent un ordre ou une séquence d'événements ou d'épisodes de soin ; ensuite, b) les associations « a pour motif » qui relient des événements ou des épisodes de soin à leurs causes sous-jacentes, telles que des maladies ou des symptômes ; puis, c) la relation « participe », qui lie des individus ou des médicaments à l'épisode de soin auquel ils sont associés ; quatrièmement, d) les relations de qualification offrent des précisions sur les entités, en se référant à des attributs comme le niveau scolaire ou la texture pour les individus, ou pour caractériser des soins et événements. Finalement e) les relations de dosage qui concernent la connexion entre des médicaments ou substances chimiques et leurs dosages, modes d'administration, et fréquences d'administration. Cette approche est illustrée dans la figure 4 où nous présentons un exemple de ces relations.

3.6.2 Solution proposée

Dans un cadre non supervisé, le principal défi est l'absence d'échantillons étiquetés indiquant la relation spécifique r pour chaque paire d'entités (e_h, e_t) dans la phrase avec e_h est l'entité tête et e_t et l'entité queue. Par conséquent, l'ensemble des relations $\mathcal{R}$ (ensemble de r_i) est explicitement défini grace aux relations de l'ontologie et la tâche repose sur l'identification de motifs, de dépendances syntaxiques et d'indices sémantiques au sein de la phrase x pour inférer la relation potentielle r.

Phase 1 - Selection des relations candidates. La sélection des relations est initialement guidée par la structure et les

relations présentes dans l'ontologie. Par exemple, si entre les entités *Maladie* et *Signe Clinique* il n'existe pas de relation dans l'ontologie, aucune relation candidate n'est considérée entre ces entités dans notre système. Cette approche nous permet de restreindre l'ensemble des relations possibles à celles qui sont soutenues par des connaissances ontologiques, garantissant une cohérence initiale dans le processus de sélection. Dans une premiere étape, nous identifions les phrases contenant plusieurs entités nommées et établissons toutes les combinaisons possibles de relations entre elles. En tenant compte de l'ordre qui est déterminant dans notre analyse, nous considérons les entités et les relations potentielles illustrées dans notre corpus de données. Considérons la phrase type issue de notre corpus, illustrée dans la figure 4. Nous avons les entités *hospitalisation*, *ivresse pathologique*, *idéations suicidaires* ainsi que les dates spécifiques *18/05/3212* et *25/05/3212*.
Nous examinons alors les combinaisons suivantes :

- $r1(e_h, e_t)$ où e_h est *hospitalisation* et e_t est *ivresse pathologique*, la relation $r1$ pouvant être interprétée comme *a pour motif*;
- $r2(e_h, e_t)$ où e_h est *hospitalisation* et e_t est *idéations suicidaires*, la relation $r2$ étant également *a pour motif*;
- pour les dates qui peuvent être associées respectivement aux entités *hospitalisation*, *ivresse pathologique* et *idéations suicidaires* considérées comme e_h, plusieurs relations candidates sont envisageables en fonction des informations contextuelles et de l'ontologie :
  - $r3(e_h, e_t)$ pour *a pour date de début*,
  - $r4(e_h, e_t)$ pour *a pour date de fin*,
  - $r5(e_h, e_t)$ pour *a pour date*, utilisée si on ne fait pas la distinction entre la date de début et de fin,

  avec, dans les 3 cas, $e_h \in$ {*hospitalisation, ivresse pathologique, idéations suicidaires*}.

Phase 2 - Analyse Syntaxique. Nous utilisons, ensuite, un composant d'analyseur de dépendances fondé sur les transitions de Spacy. Ce dernier est fondé sur le modèle Transformer, notamment camembert-base [20] avec une précision de 0.95 [7]. Les phrases, une fois étiquetées avec des tags de parties du discours (POS), sont analysées par cet outil. Il génère alors un arbre de dépendances pour chaque phrase, illustré dans la figure 4 (panneau supérieur), et assigne une fonction syntaxique à chaque mot.

Phase 3 - Identification des relations. Cette phase exploite l'analyse des chemins syntaxiques au sein de l'arbre de dépendance par mot, traçant le parcours depuis un point de départ e_h, généralement l'effecteur, vers les entités cibles e_t. Cette analyse syntaxique révèle une absence de connexion syntaxique directe entre *ivresse pathologique* et *idéations suicidaires* et les dates, signifiant qu'aucun chemin de dépendance n'indique une relation temporelle explicite avec ces entités. Toutefois, partant de *hospitalisation*, il y a une dépendance syntaxique tant avec les entités temporelles qu'avec *Maladie* et *Signe Clinique*, indice d'une relation de modification ou de causalité. Par conséquent, nous validons les relations $r1$ et $r2$ qui relient *hospitalisation* à *ivresse pathologique* et *idéations suicidaires* via *a pour motif*, marquant un lien causal où l'hospitalisation résulte de ces conditions. Les relations $r3$, $r4$, et $r5$ associant *hospitalisation* aux dates spécifiques restent des candidats et sont sujettes à une analyse plus approfondie dans la phase suivante.

Phase 4 - Application des règles. Cette étape finale du processus d'extraction de relations tire parti de règles spécifiques centrées sur les mots situés avant les entités temporelles pour déterminer la nature précise de la relation temporelle. En s'appuyant sur des indicateurs lexicaux clairs, tels que des mots ou des expressions indiquant un commencement (« début », « commencement », « à partir du ») ou une conclusion (« fin », « jusqu'au », « termine le »), nous pouvons affiner notre compréhension des relations $r3$, $r4$, et $r5$ restantes entre *hospitalisation* et les instances temporelles. La présence de ces indicateurs dans le contexte immédiat avant une entité temporelle nous permet d'attribuer avec précision la relation la plus adéquate. Par exemple, si un indicateur de début ou de fin précède une entité temporelle associée à *hospitalisation*, la relation $r3$ ou $r4$ sont validées. Si le contexte ne spécifie pas clairement un début ou une fin, ou que les indicateurs sont ambigus ou absents, la relation générale $r5$ *a pour date* est considérée comme appropriée. En résultat, illustré dans la figure 4 (panneau Inférieur), les relations finales sélectionnées pour *hospitalisation* sont directement influencées par ces indicateurs lexicaux, renforçant l'exactitude sémantique et contextuelle de notre modèle relationnel.

4 Analyse et résultats

4.1 Analyse des performances du système

Notre approche a été évaluée de manière distincte sur trois composantes clés : l'extraction des entités nommées, la normalisation des entités, et l'extraction des relations. Pour chaque composante, nous avons réalisé une analyse manuelle approfondie en utilisant un schéma d'annotation dédié conçu pour mesurer précisément les performances de notre pipeline. L'outil d'annotation BRAT a été utilisé pour sa facilité d'usage et sa capacité à répondre à nos critères spécifiques. Dans le cadre de l'optimisation de l'évaluation manuelle, nous avons classifié les entités extraites en 17 concepts uniques de haut niveau de l'ontologie OntoPSY. Les annotations dans BRAT incluaient les URI correspondant à chaque concept de l'ontologie, donnant ainsi aux annotateurs la possibilité de corriger les annotations au besoin. Pour les relations, nous avons retenu 14 types de relation. Il faut noter la différence d'approche d'évaluation entre les entités et les relations : les entités correspondent pour une grande majorité à des concepts médicaux et sociétaux : ils peuvent être appréhendés par les évaluateurs qui ont 17 concepts de haut niveau à leur disposition et peuvent préciser les concepts repérés à l'envi en balayant l'ontologie. les relations décrites dans l'ontologie sont très précises en raison du rôle tenu par icelles – modèle de données de

7. https://spacy.io/models/fr#fr_dep_news_trf

la plateforme gérant les données cliniques – dans le projet PsyCARE. Les relations telles qu'elles sont appréhendées par les experts sont plus proches de dépendances syntaxiques visibles dans les phrases : c'est pour cela qu'on en a retenu 14 (synthétisées en 5 types, Cf.3.6.1) et que nous sollicitons les experts dessus sans leur demander d'approfondir les URI.

Notre corpus d'évaluation est composé de 60 CRH extraits aléatoirement de l'ensemble de données (5120 phrases, 10013 concepts ontologiques non uniques annotés). Deux personnes ont évalué l'annotation sémantique et leur contexte, notamment le repérage de la négation, l'hypothétique, la temporalité et la personne impliquée (p. ex. le patient vs un membre de sa famille). Dans les résultats, pour les tâches REN et ER, nous indiquons les scores de précision, de rappel et de F1.

Dans le processus de normalisation des entités, où il est possible d'attribuer à chaque entité détectée plusieurs URIs candidats issus de l'ontologie, l'utilisation de métriques fondées sur le classement s'avère essentielle pour évaluer avec précision les correspondances établies. Ainsi, nous mettons en œuvre des mesures largement utilisées dans ce domaine : Hits@1, Hits@5, ainsi que le rang réciproque moyen MRR. Hits@1 et Hits@5 évaluent le rappel en mesurant la présence des correspondances correctes parmi les 1 et 5 premiers résultats proposés par notre système de normalisation. Le MRR, quant à lui, offre une perspective quant à la qualité du classement en calculant la moyenne des inverses des positions attribuées aux correspondances correctes. Hits@1, en particulier, permet de déterminer dans quelle mesure l'URI le mieux classé par notre système coïncide avec une correspondance vérifiée. Le MRR complète cette analyse en appréciant de manière globale l'exactitude du classement des URIs candidats, grâce à l'agrégation des positions relatives des correspondances avérées.

4.2 Résultat

Nous avons évalué l'accord inter-annotateurs à travers les contributions des deux annotateurs sur l'ensemble des tâches. Cet accord s'est avéré être de 0,79, indiquant une cohérence significative dans les annotations fournies. En conséquence, nous avons procédé à la consolidation de tous les comptes rendus annotés. La REN et l'extraction du contexte des concepts ont démontré une précision globale de 0.9610, un rappel de 0.9248 et un score F1 de 0.9425. Les résultats détaillés sont présentés dans le Tableau 1.

Dans le cadre de l'évaluation de la ER, nous avons distingué 14 types de relations différents. Ces derniers ont été synthétisés dans le Tableau 2, représentant un ensemble de 3473 annotations. Les performances globales atteintes pour cette tâche sont résumées par une précision de 0.92, un rappel de 0.81, et un score F1 de 0.86.

Dans le cadre de notre analyse des performances de normalisation des entités, les résultats obtenus pour les métriques clés sont particulièrement révélateurs. Pour Hits@1, nous avons atteint un taux de 84.8%. Cette performance souligne l'efficacité du système à déterminer l'URI le plus pertinent pour chaque entité dès la première proposition.

TABLE 1 – Résultats quantitatifs de l'évaluation de la reconnaissance d'entités nommées.

Entité nommée		Total	Precision	Recall	F1
Age		164	0.977	0.904	0.939
Substance	Name	1034	0.8200	0.9805	0.8822
	Dosage	608	0.99	0.94	0.97
	DrugForm	14	0.857	0.857	0.857
Temporal Inf.	Date	1208	0.9942	0.9709	0.9824
	Duration	221	0.7481	0.9619	0.8417
	Frequency	511	0.9954	0.7688	0.9819
	Time	88	0.8750	0.9459	0.9091
EpisodeDeSoin		400	0.975	0.9485	0.8273
EvenementVecu		343	0.9589	0.9333	0.9459
ExamenClinique		308	0.9799	0.8811	0.8875
Hospitalisation		520	0.9954	0.9688	0.9819
Individu		540	0.991	0.7774	0.8747
Maladie		1166	0.9894	0.9852	0.9843
PartieDuCorps		22	0.98	0.6667	0.8000
Qualifier		600	0.9882	0.8802	0.9342
SigneClinique		1250	0.9870	0.9775	0.9823
AnnotationsToAdd		721	0.9256	0.8077	0.8626

TABLE 2 – Résultats quantitatifs des évaluations de l'extraction de relations.

Relation	Total	Precision	Rappel	F1
Relations Temporelles	860	0.9130	0.8077	0.8571
A pour motif	1050	0.9750	0.9070	0.9398
Participe	286	0.9831	0.5800	0.7296
Qualifie	756	0.9211	0.7368	0.8188
Relations Medicaments dosage	521	0.9046	0.8333	0.8678

En élargissant notre évaluation aux cinq premières propositions avec Hits@5, le taux s'améliore pour atteindre 90.4%, démontrant ainsi la capacité du système à inclure la correspondance exacte parmi les choix les plus privilégiés. Cette métrique confirme que même si la correspondance idéale n'est pas toujours première, elle figure presque toujours parmi les premières propositions.

Quant au MRR, qui offre une vue d'ensemble de la performance du système en prenant en compte le rang de la bonne réponse, le score obtenu est de 85%.

5 Discussion

Les résultats témoignent des performances obtenues au sein de notre étude. Notre démarche méthodique, adaptée tant à l'analyse des entités qu'à celle des relations, a été fructueuse et a mis en lumière les défis spécifiques liés au traitement de textes complexes, en particulier ceux du domaine de la psychiatrie.

Ces résultats encourageants s'expliquent principalement par deux facteurs. Premièrement, la richesse du vocabulaire de l'ontologie, notamment dans les domaines des signes, symptômes psychiatriques, des maladies, et des événements vécus, ce qui contribue directement à la qualité du plongements ontologique ainsi qu'à celle de la normalisation. En outre, nous avons réalisé des expérimentations sur la qualité du plongement générés par OWL2Vec*, qui ont

révélé notre aptitude à distinguer efficacement les classes de premier niveau de l'ontologie OntoPSY, cette analyse est disponible sur un notebook Jupyter GitHub [8].

Le second facteur déterminant est l'intégration de la structuration spécifique des CRH dans le processus d'extraction des relations, notamment à travers l'ajout de règles de filtrage. Cette adaptation améliore considérablement la précision de notre système, bien que cela puisse représenter un défi pour la généralisation de cette approche à d'autres types de documents.

De plus, il est essentiel de souligner que la performance globale de notre système est étroitement liée à l'efficacité du modèle d'extraction des syntagmes. Les imperfections inhérentes à ce processus ne se limitent pas à leur occurrence initiale ; elles se propagent à travers le système, impactant chaque étape subséquente de l'analyse. Cette interdépendance souligne la nécessité d'une extraction précise des syntagmes dès les premiers stades, étant donné que toute erreur générée peut être amplifiée et influencer l'ensemble des résultats obtenus. Cette limitation nécessite une étude d'ablation pour comprendre l'impact de chaque étape sur les résultats finaux du système d'annotation. En outre, l'analyse de dépendance influence également la précision et le rappel des tâches d'extraction de relations. Cette interdépendance met en exergue l'importance vitale d'une extraction précise des groupes nominaux dès les premiers instants, puisque les erreurs initiales peuvent être exacerbées, affectant de manière significative la qualité totale des résultats. Face à cette contrainte, il s'avère indispensable de recourir à une méthode d'ablation pour identifier avec exactitude l'impact de chaque élément sur la performance globale du système d'annotation.

L'utilisation de méthodes d'apprentissage non supervisé, intégrant une ontologie spécifique au domaine pour affiner la précision de l'apprentissage, pourrait diminuer le besoin d'annotations manuelles. La performance globale de l'annotation représente la somme des performances des différents composants. Suite à plusieurs améliorations apportées à chaque élément, comme l'intégration de règles spécifiques ou l'emploi de plongements pré-entraînés sur un corpus médical, nous avons atteint un niveau de performance jugé satisfaisant. Néanmoins, des opportunités d'amélioration de la performance de chaque composant du système proposé subsistent et feront l'objet de recherches approfondies dans nos travaux futurs.

6 Conclusion

L'objectif principal de ce travail est de reconstruire les données structurées des patients à partir de leurs CRH afin d'enrichir les données du projet PsyCARE. À cette fin, nous avons combiné l'utilisation de méthodes d'apprentissage non supervisées avec OntoPSY, une ontologie spécifique à la psychiatrie, pour récupérer et normaliser les entités biomédicales et identifier les relations entre ces paires d'entités dans le texte.

Initialement, pour l'extraction d'information, nous avons adapté l'algorithme PatternRank d'extraction de syntagmes clés. Nous avons ensuite exploité le plongement de l'ontologie dans un espace vectoriel avec OWL2Vec* pour associer ces informations aux concepts correspondants de l'ontologie. Enfin, en nous appuyant sur la structure de l'ontologie et l'analyse des dépendances syntaxiques, nous avons pu extraire les relations entre les entités.

Ce travail se distingue par l'exploitation des technologies du web sémantique combinées à l'apprentissage profond pour créer automatiquement des documents annotés dans le domaine de psychiatrie. Les performances de notre système sont prometteuses et ouvrent la voie à de nombreuses améliorations en termes de performances. Cette initiative a mis en lumière l'apport des plongements d'ontologie dans le contexte d'ontologies biomédicales variées et interconnectées, renforçant l'efficacité de l'annotation sémantique. Bien que cet article se concentre sur le domaine de la psychiatrie, des tests préliminaires dans le champ de la néphrologie avec une ontologie dédiée ont également révélé des perspectives encourageantes, bien que ces dernières ne soient pas l'objet principal de cette publication.

Les prochaines étapes de notre recherche incluent une analyse comparative entre notre méthode utilisant des plongements de mots non contextuels (word2vec) et les plongements sémantiques contextuels [7] pour l'annotation sémantique. Nous visons à améliorer le taux de rappel dans l'extraction de relations, en envisageant l'utilisation de notre base de données annotées et l'application d'apprentissage faiblement supervisé. Nous prévoyons également de tester l'efficacité de notre approche avec des données annotées en français disponibles publiquement.

8. https://github.com/AouinaOns/Semantic-Annotation

Remerciements

Ce travail a bénéficié d'une aide de l'État gérée par l'Agence Nationale de la Recherche au titre du Programme d'Investissements d'Avenir portant la référence PsyCARE ANR-18-RHUS- 0014.

Références

[1] Ons Aouina, Jacques Hilbey, and Jean Charlet. Ontology-Based Semantic Annotation of French Psychiatric Clinical Documents. *Studies in health technology and informatics*, 302 :793–797, May 2023.

[2] Dennis Aumiller et al. Online dateing : A web interface for temporal annotations. 07 2022.

[3] Antoine Bordes et al. Translating Embeddings for Modeling Multi-relational Data. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013.

[4] Borui Cai et al. Temporal knowledge graph completion : A survey. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, IJCAI-2023. International Joint Conferences on Artificial Intelligence Organization, August 2023.

[5] Ricardo Campos et al. YAKE ! Keyword extraction from single documents using multiple local features. *Information Sciences*, 509, 2020.

[6] Jiaoyan Chen et al. OWL2Vec* : Embedding of OWL Ontologies, January 2021. arXiv :2009.14654 [cs].

[7] Jiaoyan Chen et al. Contextual Semantic Embeddings for Ontology Subsumption Prediction, March 2023.

[8] Shivani Choudhary, Tarun Luthra, Ashima Mittal, and Rajat Singh. A Survey of Knowledge Graph Embedding and Their Applications, July 2021.

[9] Pratik Devkota et al. Using ontology embeddings with deep learning architectures to improve prediction of ontology concepts from literature. 2023.

[10] Jacob Devlin et al. BERT : Pre-training of Deep Bidirectional Transformers for Language Understanding, May 2019. arXiv :1810.04805 [cs].

[11] Aryo Pradipta Gema et al. Knowledge graph embeddings in the biomedical domain : Are they useful ? a look at link prediction, rule learning, and downstream polypharmacy tasks, 2023.

[12] Maarten Grootendorst. Keybert : Minimal keyword extraction with bert., 2020.

[13] Jacques Hilbey, Xavier Aimé, and Jean Charlet. *Temporal Medical Knowledge Representation Using Ontologies*. May 2022.

[14] Ali Hur et al. A Survey on State-of-the-art Techniques for Knowledge Graphs Construction and Challenges ahead, December 2021.

[15] Lars Juhl Jensen et al. Literature mining for the biologist : from information retrieval to biological discovery. *Nature Reviews Genetics*, 7(2), 2006.

[16] İlknur Karadeniz et al. Linking entities through an ontology using word embeddings and syntactic reranking | BMC Bioinformatics 2019 | Full Text.

[17] Dongkwan Kim et al. Supervised Graph Attention Network for Semi-Supervised Node Classification. 2019.

[18] Maxat Kulmanov et al. El embeddings : Geometric construction of models for the description logic el ++.

[19] Xuexiang Li et al. Efficient Medical Knowledge Graph Embedding : Leveraging Adaptive Hierarchical Transformers and Model Compression. 12, 2023.

[20] Louis Martin et al. Camembert : a tasty french language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, volume abs/1911.03894, 2019.

[21] Rada Mihalcea et al. TextRank : Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*.

[22] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space, September 2013.

[23] Scott Miller, Heidi Fox, Lance Ramshaw, and Ralph Weischedel. A Novel Use of Statistical Parsing to Extract Information from Text. In *1st Meeting of the North American Chapter of the Association for Computational Linguistics*, 2000.

[24] Jose Moreno et al. Combining word and entity embeddings for entity linking. 2017.

[25] Dietrich Rebholz-Schuhmann et al. Text processing through Web services. *Bioinformatics*, 2008.

[26] Tim Schopf et al. PatternRank : Leveraging Pretrained Language Models and Part of Speech for Unsupervised Keyphrase Extraction. In *Proceedings of the 14th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management*, pages 243–248, 2022.

[27] N. Shadbolt, T. Berners-Lee, and W. Hall. The Semantic Web Revisited. *IEEE Intelligent Systems*, 21(3) :96–101, January 2006.

[28] Wei Shen et al. Entity linking with a knowledge base : Issues, techniques, and solutions. *IEEE Transactions on Knowledge and Data Engineering*, 27 :443–460, 2015.

[29] Fatima Zohra Smaili et al. Onto2Vec : joint vector-based representation of biological entities and their ontology-based annotations. *Bioinformatics*, 2018.

[30] Fatima Zohra Smaili et al. OPA2Vec : combining formal and informal content of biomedical ontologies to improve similarity-based prediction. 35, 11 2018.

[31] Chengyu Sun et al. A Review of Unsupervised Keyphrase Extraction Methods Using Within-Collection Resources. *Symmetry*, 12(11) :1864, November 2020.

[32] Si Sun, Zhenghao Liu, Chenyan Xiong, et al. Capturing Global Informativeness in Open Domain Keyphrase Extraction, September 2021.

[33] Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. RotatE : Knowledge Graph Embedding by Relational Rotation in Complex Space, February 2019.

[34] The OBI Consortium, Barry Smith, et al. The OBO Foundry : coordinated evolution of ontologies to support biomedical data integration. *Nature Biotechnology*, 25(11) :1251–1255, November 2007.

[35] Kristina Toutanova et al. Observed versus latent features for knowledge base and text inference. In *Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality*, April 2015.

[36] Perceval Wajsburt et al. Eds-nlp : efficient information extraction from french clinical notes.

[37] Xiaojun Wan et al. Single document keyphrase extraction using neighborhood knowledge. In *Proceedings of the 23rd National Conference on Artificial Intelligence - Volume 2*, AAAI'08. AAAI Press, 2008.

[38] Rui Xing et al. BioRel : towards large-scale biomedical relation extraction. *BMC Bioinformatics 2020*.

[39] Bishan Yang et al. Embedding entities and relations for learning and inference in knowledge bases, 2015.

KEOPS-CTS : Knowledge ExtractOr Pipeline System pour l'analyse de Champs Thématiques Stratégiques

S. Valentin[1,2], T. Helmer[3], X. Rouvière[3], M. Roche[1,2]

[1] CIRAD, UMR TETIS, 34398 Montpellier, France
[2] TETIS, Univ Montpellier, AgroParisTech, CIRAD, CNRS, INRAE, Montpellier, France
[3] CIRAD, DSI, 34398 Montpellier, France

sarah.valentin@cirad.fr

Résumé

Les outils de gestion des connaissances visent à faciliter les processus de collecte et d'organisation des connaissances afin de rendre ces connaissances disponibles dans une base partagée. Nous présentons la plateforme KEOPS-CTS dédiée à l'extraction et la gestion de connaissances à partir de données textuelles produites par un organisme de recherche qui traite les problématiques en agriculture appliquées aux pays du Sud. La démarche de KEOPS est guidée par les informations sémantiques (CTS - Champ Thématique Stratégique) contenues dans les textes hétérogènes intégrés à la plateforme. Nous proposons une évaluation de l'expansion lexicale afin d'améliorer l'analyse des documents relatifs à l'agroécologie.

Mots-clés

Système de gestion des connaissances, Fouille de texte, Expansion du vocabulaire, Plongements lexicaux

Abstract

Knowledge management tools aim to facilitate the process of collecting and organising knowledge to make it available in a shared database. We present the KEOPS-CTS platform, dedicated to the extraction and management of knowledge from textual data produced by a research organization addressing agricultural issues in the Global South. The KEOPS approach is guided by the semantic information (CTS - Strategic Thematic Field) contained in the heterogeneous texts integrated into the platform. We propose an evaluation of lexical expansion to improve the analysis of documents related to agroecology.

Keywords

Knowledge management system, Text mining, Vocabulary expansion, Word embeddings

1 Introduction

Le processus de gestion de connaissances à partir de données (*Knowledge Management*) implique généralement la succession de plusieurs étapes, parmi lesquelles le nettoyage, l'indexation, la sélection et la transformation des données avec l'utilisation éventuelle de modèles. Á ces étapes peut s'ajouter la mise en forme des résultats via des choix de visualisation adaptés.

Quel que soit le domaine d'application, une part importante de l'information disponible est stockée sous forme de texte, dans des documents provenant de sources hétérogènes telles que des documents officiels, institutionnels, des contenus de site web, des communications, etc. Les données textuelles sont dites "non-structurées" et nécessitent des techniques de recherche d'information et d'analyse adaptées fondées sur la fouille de texte et le traitement automatique du langage naturel (TALN). Par exemple, [1] utilise une approche de classification non supervisées afin de regrouper des documents d'ingénierie sur la base de leur proximité sémantique. [14] compare des termes extraits par une mesure de pondération classique avec des termes identifiés suite à un clustering automatique dans un processus de classification automatique de la polarité de documents d'évaluation de projet. D'autres travaux se sont intéressés à la classification de connaissances textuelles à partir documents normatifs [6], dans une approche supervisée reposant sur des modèles de langue pré-entrainés de type *Transformers*.

Parallèlement aux travaux de recherche, les approches d'analyse de données textuelles sont de plus en plus intégrées aux outils de gestion des connaissances (Knowledge Management Systems, ou KMS). Ces outils ont pour but de soutenir "l'un des trois processus fondamentaux de gestion des connaissances : la génération, la codification et le transfert de connaissances." [3]. Par exemple, TyDI (Terminology Design Interface) est une plateforme collaborative pour la validation manuelle et la structuration de termes à partir de terminologies existantes ou de termes extraits automatiquement à l'aide d'outils dédiés [11]. D'autres outils comme NooJ [13] utilisent des approches linguistiques pour construire et gérer des dictionnaires et des grammaires. NooJ intègre plusieurs méthodes de traitement du langage naturel, comme les approches de reconnaissance des entités nommées. D'autres plateformes intègrent des composants d'exploration de texte comme CorTexT [2]. CorTexT permet l'extraction d'entités nommées et des approches avancées d'exploration de texte (par exemple, la modélisation de sujets, l'intégration de termes, etc.) sont intégrées dans

cette plateforme.

La création d'outils de gestion des connaissances a un tropisme historique dans le domaine des pratiques organisationnelles en entreprises [3, 4]. Cependant, la création et l'utilisation efficace de l'information et du savoir sont aussi des besoins clés dans le domaine de la recherche et de la gestion de projets scientifiques. La plateforme AReS (*Agricultural Research e-Seeker*) est une plateforme qui permet d'explorer et d'extraire du contenu à partir de dépôts d'informations et de données textuelles liés au Groupe consultatif pour la recherche agricole internationale (CGIAR) et à ses partenaires [1]. Cet outil est conçu pour aider à rendre les connaissances du CGIAR trouvables, accessibles, interopérables et réutilisables et propose une indexation avec Agrovoc, un thésaurus dédié au domaine agricole [2].

KEOPS (Knowledge ExtractOr Pipeline System) est une plateforme qui applique diverses méthodes d'indexation et de classification à des données textuelles provenant de bases de données, de documents ou de pages web [9]. Une caractéristique de KEOPS est de guider l'indexation, l'analyse et la visualisation des informations et connaissances produites selon un angle sémantique. Ce dernier s'appuie sur un vocabulaire contrôlé. Par exemple, dans le cadre du projet LEAP4FNSSA [3], les documents ont été analysés selon un lexique lié à la sécurité alimentaire [12].

En sortie, KEOPS combine les résultats de classification et d'indexation pour générer des connaissances sur chaque texte et groupe de textes.

Dans cet article, nous décrivons l'adaptation de l'outil KEOPS à la gestion de documents sous le prisme des Champs Thématiques Stratégique (CTS) d'un organisme de recherche, le Cirad (section 2). Nous présentons ensuite notre méthodologie sur l'expansion d'un vocabulaire relatif à l'agroécologie (section 3), les résultats du cas d'étude proposé (section 4) et une discussion.

2 KEOPS-CTS

2.1 Données

L'objectif de l'outil KEOPS-CTS est de permettre la collecte, l'indexation et l'analyse de données textuelles issues de différentes sources et caractérisant différentes étapes du cycle de vie de l'activité de recherche du Centre de coopération internationale en recherche agronomique pour le développement (Cirad) : l'orientation scientifique, les activités en cours, et les productions scientifiques associées (Figure 1). Les données et leurs sources sont détaillées ci-après :

2.1.1 Orientation scientifique

Cet axe est représenté par (1) les profils de poste, qui renseignent sur la manière dont les compétences techniques et scientifiques sont utilisées dans les domaines couverts par les Champs Thématiques Stratégique (CTS), (2) les lettres pluriannuelles d'objectif (LPO) qui explicitent la politique scientifique et partenariale des unités de recherche, en cohérence avec la stratégie du Cirad et (3) les notes CTS, qui font un état des lieux des recherches conduites au Cirad dans chacun des CTS, les enjeux spécifiques et les fronts de science sur lesquels le Cirad souhaite être visible avec ses partenaires.

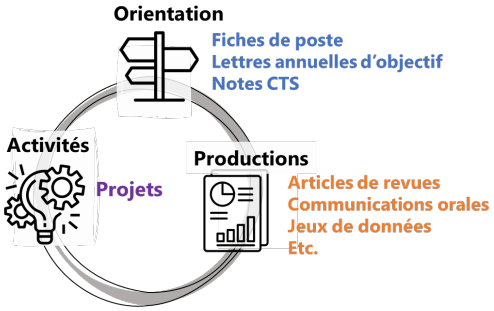


FIGURE 1 – Données textuelles intégrées dans l'outil KEOPS-CTS.

2.1.2 Activités scientifiques

Le contenu des activités est analysé sous le prisme des résumés des projets dans lesquels le Cirad est impliqué et recensés sur la plateforme CORDIS [4]), principale source des projets financés par les programmes cadres de l'Union Européenne.

2.1.3 Productions scientifiques

Enfin, nous avons extrait de l'archive ouverte du Cirad, (Agritrop [5]), l'ensemble des résumés correspondants aux productions suivantes : articles de revues scientifiques avec ou sans comité de lecture, communications avec actes, ouvrages, thèses et jeux de données.

Au moment de l'extraction, la base de données textuelles finale contient 18721 documents (Tableau 1).

Type de source	Nombre de documents	Nombre moyen de termes
Orientation scientifique		
Fiches de postes	1776	290
Lettres d'objectifs	1047	350
Notes CTS	14	6096
Activités		
Projets	121	439
Productions		
Articles de revue	9386	246
Ouvrages	2932	233
Communications	2383	244
Thèses	543	247
Jeux de données	519	177

TABLE 1 – Nombre de documents par type de source au 1/03/2024

1. https://cgspace.cgiar.org/explorer/
2. https://agrovoc.fao.org/browse/agrovoc/en/
3. Long-term Europe-Africa Research and Innovation Partnership for Food and Nutrition Security and Sustainable Agriculture
4. https ://cordis.europa.eu/
5. https ://agritrop.cirad.fr/

2.2 Classification

Un module de classification est intégré dans l'outil KEOPS-CTS afin de déterminer automatiquement le type de source de chaque document. Cette classification repose sur une approche supervisée : plusieurs familles de classifieurs (e.g. Random Forest, Multilayer Perceptron) sont entraînées sur un jeu de données annotées afin de prédire le type de source de chaque nouveau document [9].

2.3 Indexation

L'indexation dans KEOPS CTS repose sur trois types d'approches :

- L'indexation thématique, réalisée à partir de vocabulaires thématiques construits par des experts (termes sources et leurs synonymes)
- L'indexation thématique réalisée à partir de thésaurus spécialisés tel que Agrovoc [6].
- L'indexation par une terminologie acquise automatiquement à l'aide de BioTex [8] ;
- L'indexation automatique par un ensemble d'entités nommées (e.g. lieux, organisations, etc.) extraites par un modèle pré-entrainé issu de la librairie SpaCy [7].

Chaque document est indexé avec l'ensemble de ces approches (Figure 2), ce qui permet de réaliser des requêtes combinant informations thématiques et informations transversales (e.g. localisations).

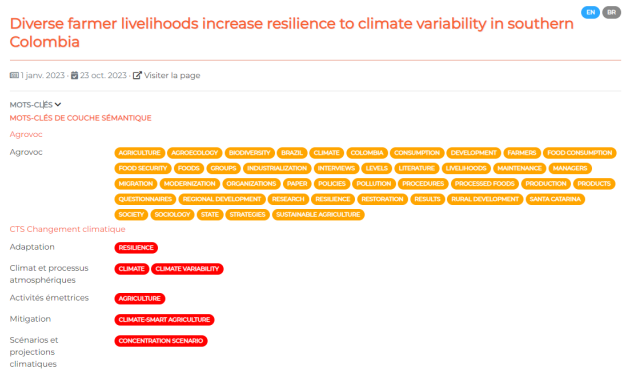


FIGURE 2 – Interface de KEOPS-CTS montrant la liste des vocabulaires d'indexation et termes associés.

3 Constitution de vocabulaires thématiques

Les lexiques (ou vocabulaires) thématiques sont des entrées indispensables au processus d'extraction de connaissances. Ils constituent l'angle de vue expert à partir duquel les données textuelles vont être indexées et analysées. Les termes, qui constituent un vocabulaire donné, visent à refléter de façon la plus exhaustive possible les concepts associés à une thématique. Les lexiques correspondant aux différents CTS sont initialement construits grâce à une méthode itérative combinant avis d'expert et méthodes d'extraction automatique [12, 5].

Dans les sections suivantes, nous décrivons les approches proposées pour étendre automatiquement les termes source d'un vocabulaire donné, en les évaluant à travers un lexique dédié à l'agroécologie.

3.1 Expansion du lexique

La découverte de synonymes à partir d'un corpus massif est une tâche indispensable pour la découverte automatique de connaissances : elle permet d'améliorer les tâches d'indexation et de recherche d'information. Pour un terme donné, ses synonymes font référence aux termes qui peuvent être utilisés de manière interchangeable dans certains contextes. Nous avons comparé deux méthodes pour l'expansion de vocabulaires source, i.e. le plongement lexical (*word embedding*) et une approche fondée sur un modèle de langue génératif (Figure 3).

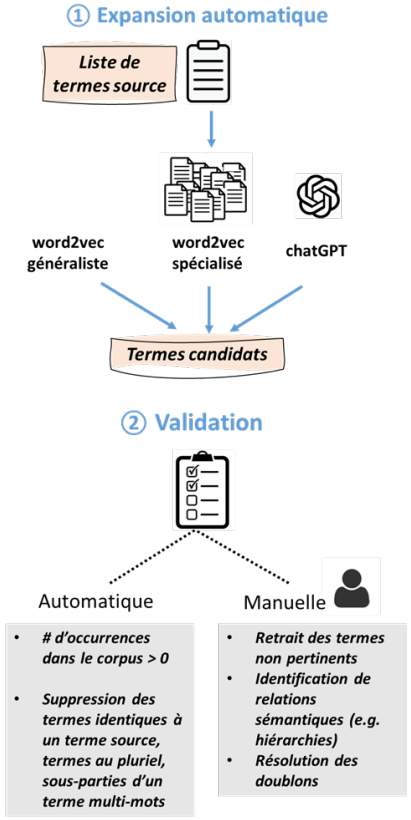


FIGURE 3 – Approche globale d'expansion du vocabulaire source combinant approches automatiques et validation d'expert.

Ci-après, l'expression "terme source" désigne un terme issu du vocabulaire constitué par les experts. Il s'agit des termes pour lesquels nous souhaitons obtenir des synonymes. Ils incluent des termes formés d'un seul mot (e.g. *agroforestry*) et des termes formés de plusieurs mots, ou "termes multi-mots" (e.g. *family farming*). Les approches d'expansion ont

6. https://agrovoc.fao.org/browse/agrovoc/en/
7. https://spacy.io/

été évaluées manuellement sur la base de trois termes par terme source.

3.1.1 Modèles de plongement lexical

Les modèles de word embedding ont été introduits par Mikolov et al. en 2013 [10]. Cette approche repose sur l'entraînement de modèles à partir de large corpus afin de traduire l'ensemble des termes (vocabulaire) en une représentation numérique de haute dimension. Cette représentation numérique des termes peut être utilisée pour comparer différents termes et trouver des termes similaires sur le plan sémantique. Nous avons comparé deux stratégies d'obtention d'*embeddings* :

- l'utilisation d'un modèle de word embedding pré-entrainé à partir d'un large corpus (modèle w2vG) et disponible sur HuggingFace [8]
- l'entraînement de modèles spécialisés à partir de notre corpus d'application (modèles w2vS). Les paramètres d'entrainement correspondent aux paramètres par défaut du modèle Word2vec implantés dans la librairie python Gensim, en faisant varier le type d'architecture (skip-gram et CBOW).

Les modèles w2vS skip-gram et CBOW ont été entrainés à partir de l'ensemble des documents de la base de KEOPS-CTS. Le modèle ne produit de représentation vectorielle que pour les termes initialement présents dans le corpus d'apprentissage. Afin de générer des représentations des termes multi-mots du vocabulaire source lors de l'entrainement des modèles w2vS, ces termes ont donc préalablement été détectées et concaténées avec le séparateur '_' dans le corpus d'entrainement. Le corpus a été converti en minuscule et les caractères non alphanumériques ont été retirés.

3.1.2 Modèle de langue génératif

Notre seconde approche a consisté à utiliser la capacité générative du modèle ChatGPT-3 afin de générer des synonymes. Contrairement à l'entrainement d'un modèle de plongement de termes, l'utilisation de ChatGPT ne nécessite pas de corpus d'apprentissage et le vocabulaire n'est pas restreint. Le prompt utilisé a été le suivant : "*Pour chacun de ces termes ou termes constitués de plusieurs mots, proposez au maximum trois synonymes. Les synonymes doivent être sémantiquement et grammaticalement corrects.*"

3.2 Validation

L'étape de validation manuelle consistait initialement à valider les synonymes pertinents pour chaque terme source. Or, lors de l'évaluation préliminaire, nous avons identifié un second niveau de pertinence, à savoir des termes ne correspondant pas à des synonymes, mais pouvant être d'intérêt dans le cadre de l'extension d'un vocabulaire. Trois cas ont été distingués : les termes correspondant à un concept hiérarchique supérieur, ou hyperonymes (e.g. *fertilizer* pour *green manure*), à un concept hiérarchique inférieur, ou hyponymes (e.g. *bean* pour *legume*) ou à un autre concept pertinent sans relation hiérarchique clairement identifiable (*pastoralism* pour *family farming*).

8. https ://huggingface.co/fse/word2vec-google-news-300

Pour les termes non pertinents, nous avons distingué les termes non pertinents pour le domaine d'application ou (e.g. *darling* proposé comme synonyme de *honey*) et termes ou expressions incorrects (orthographe incorrecte, sous-partie d'un terme multi-mot).

3.3 Evaluation

Au-delà de la pertinences des synonymes, nous souhaitons en évaluer la capacité d'expansion. Pour cela, nous avons défini le coefficient d'expansion (CE), à l'échelle de l'occurrence et à l'échelle du document :

- Le coefficient d'expansion à l'échelle de l'occurrence (*CEocc*) correspond au coefficient multiplicateur entre le nombre d'occurrences d'un terme source dans le corpus, et la somme du nombre d'occurrences du même terme et de son synonyme.
- Le coefficient d'expansion à l'échelle du document (*CEdoc*) correspond au coefficient multiplicateur entre (i) le nombre de documents dans lesquels apparait un terme, (ii) le nombre de documents dans lesquels apparaissent un terme ou son synonyme.

Pour une méthode d'expansion donnée, son index d'expansion est calculé en faisant la moyenne de tous les synonymes générés par cette méthode. Les termes au singulier et leur version au pluriel sont considérés. Pour pouvoir calculer ce coefficient dans le cas où le terme source n'apparait pas dans le corpus mais où le synonyme proposé est détecté, l'occurrence du terme a été arbitrairement définie à 1.

4 Résultats

Dans cette section, nous présentons l'évaluation des différentes approches d'expansion à partir d'un vocabulaire dédié à l'agroécologie. Ce vocabulaire source est dérivé d'un lexique construit par avis d'experts [5]. Il contient 213 termes source, parmi lesquels 26 termes simples et 187 termes multi-mots.

4.1 Vocabulaires des modèles

Les modèles de plongements lexicaux se basent sur des vocabulaires fixes définis par leur corpus d'entraînement et les étapes de prétraitement appliquées, telles que la suppression des nombres et des caractères spéciaux, ou la concaténation des termes multi-mots. Par conséquent, leur taux de couverture d'un vocabulaire spécifique (ensemble des termes d'un corpus) varie. Concernant le vocabulaire lié à l'agroécologie, le modèle w2vG obtient les résultats les moins satisfaisants, particulièrement pour les termes multi-mots. Ayant été entraîné sur un corpus spécialisé, le modèle w2vS couvre la quasi-totalité des termes simples, mais seulement 45% des termes multi-mots (Tableau 2).

4.2 Validation des termes issus des méthodes d'expansion

Le nombre total de synonymes pertinents après validation manuelle était de 25 pour w2vG, 12 pour w2vS (cbow), 13 pour w2vS (skip) et 433 pour ChatGPT. ChatGPT et w2vG obtiennent les meilleures proportions de synonymes pertinents (71% et 39.1%, respectivement) (Tableau 3).

Méthode d'expansion	Termes simples	Termes multi-mots
w2vG	69.3%	3.7%
w2vS	92.3%	45%
ChatGPT	100%	100%

TABLE 2 – Taux de couverture des différentes méthodes d'expansion

La proportion de synonymes parmi les termes proposés par les modèles d'embedding spécialisés (w2vS) sont très faibles (autour de 4%). Cependant, ces modèles proposent des concepts pertinents de type hyperonyme, hyponyme et autres concepts associés que ne génère pas ChatGPT.

	w2vG	w2vS (cbow)	w2vS (skip)	ChatGPT
Pertinent - synonyme	39.1%	3.8%	4.1%	71%
Pertinent - autre				
Hyperonyme	0%	2.6%	1.9%	1.1%
Hyponyme	7.8%	3.2%	1.6%	1.3%
Autre concept	17.2%	12.8%	15.9%	0.7%
Non pertinent :				
Terme non pertinent	14.1%	75.0%	73.2%	25.9%
Pluriel	12.5%	1.9%	2.2%	0%
Mauvaise orthographe	6.2%	0%	0%	0%
Sous-partie	3.1%	0.6%	1.0%	0%

TABLE 3 – Évaluation de la pertinence des termes issus des différentes méthodes d'expansion. Pour chaque méthode, les proportions représentent le nombre de termes de chaque catégorie, par rapport au nombre total de termes obtenus par cette méthode.

La répartition des termes source en fonction du nombre de synonymes pertinents générés par les différentes approches est résumée dans le Tableau 4. Les modèles de plongement de mots étudiés sont très peu performants du point de vue de la synonymie. La prise en compte de l'occurrence des synonymes dans le corpus impacte très négativement la performance de ChatGPT : 52% des termes source, les synonymes proposés sont non pertinents ou pertinents mais absents du corpus. En effet, bien que généralement correctes d'un point de vue grammatical, les propositions de ChatGPT sont parfois des constructions terminologiques peu susceptibles d'être utilisées dans un corpus spécialisé (par exemple, *eco-friendly fertilizer* pour *biofertilizer*). Les modèles w2vS ayant été entrainés sur le corpus, les termes proposés y apparaissent nécessairement.

4.3 Evaluation de l'expansion

Les coefficients d'expansion de ChatGPT sont significativement supérieurs à ceux des modèles d'embeddings : le nombre de détections d'occurrences est multiplié en moyenne par 12.6. Notamment, pour 24 termes source n'apparaissant pas dans le corpus, ChatGPT a généré des synonymes permettant de détecter le terme initial (e.g. *indigenous species* permettant de détecter le terme *native breed*). Ce comportement offre un gain conséquent en termes d'indexation.

	w2vG	w2vS (cbow)	w2vS (skip)	ChatGPT
Synonymes pertinents				
0	93.3%	93.9%	92.8%	1.6%
1	1.1%	5.5%	7.2%	16.6%
2	3.9%	0.6%	0%	22.7%
3	1.7%	0%	0%	59.1%
Synonymes pertinents et présents				
0	96.0%	93.9%	92.8%	52.0%
1	1.7%	5.5%	7.2%	26.5%
2	1.7%	0.6%	0%	16.0%
3	0.6%	0%	0%	5.5%

TABLE 4 – Proportion de termes source en fonction du nombre de synonymes pertinents obtenus par les différents modèles et du nombre de synonymes pertinents présents dans le corpus.

	w2vG	w2vS0	w2vS1	ChatGPT
CEocc	1.13	2.30	1.80	12.6
CEdoc	1.15	2.44	1.86	7.5
Nb termes source	12	11	13	178

TABLE 5 – Comparaison du coefficient d'expansion entre les différents modèles, à l'échelle du nombre d'occurrences (*CEocc*) et du nombre de documents (*CEdoc*).

5 Discussion

Dans ces travaux préliminaires, nous avons comparé deux familles d'approches pour l'expansion de vocabulaire sur la thématique de l'agroécologie, i.e. le plongement lexical et un modèle de langue génératif. Une différence fondamentale entre ces deux approches repose sur la définition de la tâche : la proximité vectorielle dans l'espace d'un modèle de word embedding ne correspond pas nécessaire à une relation de synonymie. Au contraire, la nature d'une tâche peut être explicitement définie lors du prompt associé à modèle génératif, ce qui assure une homogénéité des résultats produits. ChatGPT s'est montré particulièrement performant pour l'expansion de termes multi-mots, tâche pour laquelle les modèles de plongements lexicaux nécessitent des étapes de pre-processing adaptées et offrent des performances moindres. Les modèles d'embeddings permettent d'identifier des termes issus de relations hiérar-

chiques diverses et susceptibles d'enrichir le vocabulaire source. Lorsque les modèles sont appris à partir d'un corpus spécialisé, ils peuvent participer à l'identification de concepts pertinents non identifiés par les experts et participer à l'enrichissement d'une hiérarchie. L'évaluation de prompts dédiés à la recherche de termes issus de différents types de hiérarchies est cependant nécessaire afin de comparer les deux approches vis-à-vis de cette tâche. De plus, l'adaptation de modèles de langues sur notre corpus thématique (*fine-tuning*) pourrait significativement améliorer les performances des tâches d'extraction de synonymes et autres types de liens sémantiques [7].

6 Conclusion

Nous présentons KEOPS-CTS un système de gestion des connaissances dédié aux données textuelles produites par des activités de recherche. L'intégration des récentes avancées en traitement automatique de la langue et Intelligence Artificielle ne sont encore que peu incorporées dans les outils open-source. Nous proposons une première contribution sur l'expansion lexicale en comparant des modèles de plongements lexicaux et un modèle de langue génératif, ChatGPT, et en montrons leur complémentarité.

Les futurs travaux consisteront à intégrer ces approches et à évaluer le résultat de ces expansions à travers les différents axes d'analyse, en particulier selon les types de documents (orientation, activités et productions scientifiques).

Remerciements

Nous remercions Julien Rabatel pour le développement de KEOPS, les personnes de la Délégation à l'information scientifique et à la science ouverte et le service des ressources humaines du Cirad pour l'extraction des données. Ces travaux menés dans le cadre du projet CEA-First ont reçu le financement de l'Union Européenne - Programme HORIZON - Grant Agreement No. 101136771. Les données sources en agroécologie ont été acquises dans le cadre du projet ASSET (AFD, Union Européenne, FFEM).

Références

[1] Ivar Örn Arnarsson, Otto Frost, Emil Gustavsson, Mats Jirstrand, and Johan Malmqvist. Natural language processing methods for knowledge management—Applying document clustering for fast search and grouping of engineering documents. *Concurrent Engineering*, 29(2) :142–152, June 2021.

[2] Philippe Breucker, Jean-Philippe Cointet, Alexandre Hannud Abdo, Guillaume Orsal, Constance de Quatrebarbes, Tam-Kien Duong, Cristian Martinez, Juan Pablo Ospina Delgado, Luis Daniel Medina Zuluaga, Diego Fernando Gómez Peña, Tatiana Andrea Sánchez Castaño, Joenio Marques da Costa, Hajar Laglil, Lionel Villard, and Marc Barbier. Cortext manager. `https://docs.cortext.net`, October 2016.

[3] Thomas Davenport and Laurence Prusak. Working knowledge : How organizations manage what they know. *Ubiquity*, 1, January 1998.

[4] Rodrigo Valio Domínguez Gonzalez and Manoel Fernando Martins. Knowledge Management : an Analysis From the Organizational Development. *Journal of technology management & innovation*, 9(1) :131–147, April 2014.

[5] Thierry Helmer, Mathieu Roche, Pierre Martin, François Enten, Lucie Reynaud, Marie-Christine Lebret, Estelle Bienabe, Melanie Blanchard, Albrecht Ehrensperger, Ricardo Hernandez, and Germain Priour. ASSET Theoretical Lexicon : An agroecology lexicon. `https://dataverse.cirad.fr/dataset.xhtml?persistentId=doi:10.18167/DVN1/TVN3AC`, January 2023.

[6] Gu Jianan, Ren Kehao, and Gao Binwei. Deep learning-based text knowledge classification for whole-process engineering consulting standards. *Journal of Engineering Research*, July 2023.

[7] Ehsan Latif and Xiaoming Zhai. Fine-tuning chatgpt for automatic scoring. *Computers and Education : Artificial Intelligence*, 6 :100210, 2024.

[8] Juan Antonio Lossio-Ventura, Clement Jonquet, Mathieu Roche, and Maguelonne Teisseire. BIOTEX : A system for Biomedical Terminology Extraction, Ranking, and Validation. In *International Semantic Web Conference*, 2014.

[9] Pierre Martin, Thierry Helmer, Julien Rabatel, and Mathieu Roche. KEOPS : Knowledge ExtractOr Pipeline System. In *Research Challenges in Information Science*, volume 415, pages 561–567. 2021.

[10] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. *arXiv :1301.3781 [cs]*, January 2013.

[11] Claire Nédellec, Wiktoria Golik, Sophie Aubin, and Robert Bossy. Building large lexicalized ontologies from text : A use case in automatic indexing of biotechnology patents. In *Knowledge Engineering and Management by the Masses*, pages 514–523, 2010.

[12] Mathieu Roche, Agneta Lindsten, Tomas Lundén, and Thierry Helmer. LEAP4FNSSA lexicon : Towards a new dataset of keywords dealing with food security. *Data in Brief*, 45 :108680, December 2022.

[13] Max Silberztein and Agnès Tutin. NooJ, un outil TAL pour l'enseignement des langues. Application pour l'étude de la morphologie lexicale en FLE. *Apprentissage des langues et systèmes d'information et de communication (Alsic)*, (Vol. 8, n° 1) :123–134, December 2005.

[14] Nadeem Ur-Rahman and Jenny Harding. Textual data mining for industrial knowledge management and text classification : A business oriented approach. *Expert Systems with Applications*, 39(5) :4729–4739, April 2012.

Découverte de connaissances

Graphaméléon : apprentissage des relations et détection d'anomalies sur les traces de navigation Web capturées sous forme de graphes de connaissances

L. Tailhardat[1,3], B. Stach[2], Y. Chabot[1], R. Troncy[3]

[1] Orange, France
[2] UTBM, Belfort, France
[3] EURECOM, Sophia-Antipolis, France

lionel.tailhardat@orange.com ; benjaminstach.pro@gmail.com ; yoan.chabot@orange.com ; raphael.troncy@eurecom.fr

Résumé

Les modèles comportementaux sont essentiels pour la détection d'anomalies ou d'actes malveillants sur des systèmes de télécommunication à travers le Web. Cependant, les données nécessaires ne sont pas toujours disponibles et une connaissance complète de la topologie des systèmes est nécessaire pour exploiter pleinement les inférences faites par ces modèles. Pour résoudre ce problème, nous proposons l'extension Web Graphaméléon et une représentation des traces de navigation sous forme de graphe de connaissances RDF en utilisant les ontologies UCO et NORIA-O.

Mots-clés

Traces de navigation Web, Analyse du comportement des utilisateurs et des entités (UEBA), Analyse des processus, Graphe de connaissances.

Abstract

Behavioral models are essential for detecting anomalies or malicious activities on telecommunications systems occuring through the Web. However, the necessary data is not always available, and a complete understanding of the system's topology is required to fully exploit the inferences made by these models. To address this issue, we propose the Graphameleon Web extension and a representation of navigation traces in the form of an RDF knowledge graph using the UCO and NORIA-O ontologies.

Keywords

Web Browsing Traces, User and Entity Behavior Analytics (UEBA), Process Mining, Conformance Checking, Knowledge Graph.

1 Introduction

En même temps que les technologies de l'information et de la communication évoluent et posent de nouveaux défis, la cybercriminalité n'a cessée d'augmenter durant la dernière décennie. Détecter et diagnostiquer rapidement des anomalies sur les réseaux et systèmes d'information sont de fait devenus une préoccupation majeure pour de nombreuses entreprises, notamment pour les gestionnaires de réseaux critiques et de grande envergure (téléphonie fixe et mobile, fourniture d'accès Internet, réseaux nationaux et internationaux d'échange de données). En cybersécurité, l'analyse du comportement des utilisateurs et des entités [1] correspond à un ensemble de techniques pour identifier et atténuer les menaces au niveau des éléments structurants des réseaux (p.ex. routeurs, serveurs, applications) à partir de données d'usage. Cela consiste typiquement à découvrir des motifs comportementaux nominaux (ou standards), tant au niveau des interactions entre les utilisateurs et les systèmes techniques qu'entre les éléments structurants eux-mêmes, et de s'en servir comme références pour alerter sur une utilisation potentiellement malveillante.

Une part importante des interactions utilisateurs-applications se fait désormais via une interface Web. Prenons l'exemple d'un scénario simple d'exploitation d'une vulnérabilité d'une application accessible via l'Internet [2] : après une phase de reconnaissance du système ciblé, l'attaquant accède directement à la page d'accueil de la plateforme de services, utilise une technique d'injection SQL [3] pour tromper le système d'authentification, exporte des données privées, puis quitte le service en naviguant directement vers une autre page Web. Analyser les interactions de l'utilisateur avec la plateforme, et ainsi détecter ce scénario, suppose l'analyse conjointe des journaux de l'application et du trafic réseau. Or les journaux peuvent être inaccessibles ou inutilisables en raison de problèmes de confidentialité ou de format. De même, le trafic réseau peut être chiffré ou inaccessible à la collecte. Ces deux aspects entraînent une perte des informations nécessaires pour qualifier le scénario d'attaque [13].

De nombreux outils de détection existent aujourd'hui dans le domaine de la cybersécurité, chacun se concentrant sur un type spécifique de source de données. Dans cet article,

1. "User and Entity Behavior Analytics" (UEBA) en Anglais.
2. https://attack.mitre.org/techniques/T1190/
3. https://fr.wikipedia.org/wiki/Injection_SQL

nous affirmons que la mise en œuvre simultanée de ces outils n'est pas suffisante pour une compréhension efficace des situations anormales, et qu'il est nécessaire d'utiliser un vocabulaire commun pour analyser les anomalies en associant les observables (p.ex journaux applicatifs, traces réseau, alertes des outils de détection) à la topologie du réseau. Dans cette optique, nous étendons le projet Dynagraph [23] (une approche combinant des outils de capture de traces avec une application Web pour un rendu graphique des données de navigation) afin d'apprendre des modèles d'activité interprétables sous forme de données liées : l'extension Web Graphaméléon collecte les traces d'activité de l'utilisateur (trafic réseau, interactions avec le navigateur Web) lors d'une session de navigation Web et sérialise ces données dans la syntaxe RDF selon le vocabulaire UCO [40]. Les données résultantes sont ensuite injectées dans un graphe de connaissances [1] pour interpréter les traces d'activité à un niveau sémantique et dériver des motifs, notamment sous forme de réseaux de Petri. Ces modèles d'activité peuvent ensuite être utilisés, du côté utilisateur ou du côté réseaux, pour identifier des situations analogues en les projetant sur le graphe de connaissances et, sur la base de cette projection, obtenir des informations contextuelles en parcourant le graphe.

Le reste de ce document est organisé comme suit. En Section 2, nous présentons les travaux connexes du point de vue de la cartographie du Web, de la modélisation de l'activité et de la détection des anomalies. En Section 3, nous présentons notre approche pour capturer les connaissances à partir des traces de navigation Web. Cela implique une modélisation de l'activité en trois couches (HTTP, micro-activités, macro-activités) basée sur le vocabulaire UCO. Nous décrivons également le composant de collecte Graphaméléon et l'utilisation des réseaux de Petri pour la détection des anomalies dans les traces de navigation. Nos expériences et résultats sont présentés en Section 4. Enfin, nous concluons et abordons les travaux futurs en Section 5. Le code source de Graphaméléon est disponible sur `https://github.com/Orange-OpenSource/graphameleon`, ainsi que le jeu de données expérimentales sur `https://github.com/Orange-OpenSource/graphameleon-ds`.

2 Travaux connexes

Collecte et représentation des connaissances. La cartographie du Web [12] est une thématique de recherche visant la compréhension de la structure du Web et de ses utilisateurs. Les études du domaine portent sur des sujets variés tels que les méthodes de prétraitement des données [26, 37], les techniques d'identification des utilisateurs [21], les algorithmes de reconnaissance de session [8, 31], et les méthodes de découverte de motifs [9]. Du point de vue de l'analyse de l'activité, le concept de raisonnement basé sur les traces [3] guide la conception d'outils d'interprétation sémantique des artefacts de services numériques en suggérant l'utilisation de vocabulaires contrôlés et de modèles de données liées. Concernant la représentation des événements et des activités au sein de graphes de connaissances, divers modèles de données – tantôt génériques, tantôt spécifiques à un domaine d'application – sont disponibles : modélisation de processus (BBO [4], réseaux de Petri [11, 18], HTTPinRDF [16, 28]) ; analyse causale (FARO [39]) ; cybersécurité (UCO [40], MITRE D3FEND [33]) ; opérations réseau (NORIA-O [25]) ; villes intelligentes (iCity ActivityOntology [29]).

Détection d'anomalies et analyse des processus. Pour la détection d'anomalies, diverses approches ont été proposées autour d'un principe commun d'identification des écarts par rapport aux comportements normaux, notamment par des modèles statistiques [38, 34, 32], des techniques d'apprentissage automatique [41, 30] et des méthodes basées sur les graphes [7, 10]. Le domaine de l'analyse des processus se concentre sur l'extraction de modèles de processus à partir de journaux d'événements et sur l'analyse du flux réel des activités. Ces modèles fournissent des informations sur le comportement typique et la structure sous-jacente des processus de navigation Web. Des techniques de vérification de conformité [17] ont été développées dans l'exploration de processus pour comparer le comportement observé aux modèles de processus attendus et identifier les écarts.

Positionnement. Nous étendons le concept de raisonnement basé sur les traces au domaine de la cartographie du Web en considérant l'utilisation des graphes de connaissances comme moyen de représenter les données de topologie du Web et les données d'utilisation de façon conjointe et cohérente. Nous abordons ainsi une nouvelle opportunité induite par l'émergence de modèles de données applicables dans les domaines des infrastructures réseaux (pour la description de systèmes hétérogènes) et de la cybersécurité (pour la description et la gestion des attaques et des risques). Nous supposons que, dans cette émergence, la communauté est désormais en mesure de répondre au besoin de corréler les informations d'usage du Web avec la description de la structure du Web lui-même, afin d'améliorer la compréhension et la conception de systèmes complexes tout en tenant compte du couple utilisateur-système. Pour exemple, il est évident que (en particulier dans UCO), l'analyse des attaques et des vulnérabilités repose principalement sur des indicateurs de compromission par l'énumération d'artefacts issus de situations passées. Ces indicateurs ne sont cependant jamais corrélés avec la topologie des réseaux et des services, ni même avec l'organisation temporelle des artefacts, ce qui correspond à une description statique des situations anormales et met de côté la structure propre des activités (i.e. la stratégie employée en rapport à la dynamique des événements). À cet égard, nous montrons notamment avec notre proposition comment incorporer le concept de traces de navigation dans l'ontologie UCO, ce qui permet de bénéficier simultanément des connaissances en cybersécurité et du contexte réseau enregistré par ailleurs (i.e. par les analystes en cybersécurité et les opérateurs réseau, respectivement) via l'ontologie NORIA-O, tout en garantissant une représentation nor-

malisée et homogène des données. De plus, nous étendons l'utilisation de l'analyse des processus et de la vérification de conformité aux graphes de connaissances, en capitalisant sur l'alignement des ces techniques avec les principes du raisonnement basé sur les traces.

3 Approche

L'approche proposée comporte trois parties, le but étant de réaliser une collecte de données dont le résultat permettra à la fois d'analyser les traces de navigation Web dans leur contexte réseau et d'apprendre des motifs d'activité. La première partie consiste à développer une modélisation sémantique des activités des utilisateurs sous forme de graphe de connaissances en réutilisant l'ontologie UCO (§3.1). La seconde partie concerne la conception de l'outil Graphaméléon, une extension de navigateur Web permettant de capturer les données de navigation et de les sérialiser en RDF (§3.2). La troisième partie porte sur l'intégration de Graphaméléon dans une chaîne de traitement de données (Figure 1) dont le principe est d'extraire des motifs d'activité en utilisant les outils d'analyse des processus et une représentation sous la forme de réseaux de Petri (§3.3).

3.1 Modélisation sémantique

Modélisation de l'activité des utilisateurs. Le concept d'activité manque de définition précise pour analyser la navigation sur le Web, car son interprétation repose fortement sur les données et l'échelle d'observation choisies. Il est en effet nécessaire de distinguer si l'identification d'une activité repose sur les interactions d'un utilisateur avec un site Web, ou si elle repose sur les échanges de paquets TCP entre le navigateur et le serveur portant ledit site Web. Pour commencer, supposons qu'une connexion HTTP est établie entre le navigateur Web de l'utilisateur et le serveur Web à partir d'une demande initiée par l'utilisateur. Le document demandé (p.ex. une page Web) nécessite, en règle générale, le chargement de ressources complémentaires telles que des images, des scripts ou autres. Ces dépendances impliquent un ensemble de sous-requêtes. Du point de vue de l'utilisateur, l'action consiste à naviguer vers un site Web par un clic sur un hyperlien (ou à accéder directement à la page via une URL), alors que du point de vue du navigateur Web, il s'agît d'une séquence de requêtes. De cette distinction, nous définissons deux niveaux de granularité pour discuter des traces de navigation. Celui nommé "micro-activité" correspond aux requêtes. Dans le niveau supérieur, nommé "macro-activité", nous considérons une trace comme étant un ensemble de requêtes et d'interactions. Par interaction, nous entendons toute action à l'initiative de l'utilisateur qui a une conséquence sur une page Web (p.ex. clic sur un hyperlien, renseigner un champ de formulaire, clic sur un bouton du navigateur Web).

Projection sémantique. Les graphes de connaissances permettent de gérer de façon unifiée des données hétérogènes et issues de sources variées. Le fonctionnement type des navigateurs Web repose d'ores-et-déjà sur des normes et des protocoles établis. Par rapport à cette normalisation, nous considérons que l'apport stratégique des graphes de connaissances est de faciliter l'intégration de données provenant de sources extérieures au contexte du navigateur Web. Nous remarquons que l'ontologie UCO [40] semble bien adaptée à notre objectif car elle permet la représentation des activités de navigation Web à différentes échelles, en incluant des informations concernant les cycles d'actions, les actions individuelles, les connexions réseaux, les protocoles de communication, les ressources techniques utilisées, les noms de domaines Internet et les adresses IP.
Ainsi, notre stratégie pour construire le graphe de connaissances consiste à maximiser la réutilisation des concepts/propriétés définis dans UCO, et de faire correspondre les champs et les valeurs capturés au niveau du navigateur Web avec ces concepts/propriétés chaque fois que leur sémantique s'aligne. La Figure 2 illustre cette mise en œuvre en présentant le modèle de données. Les règles de construction de graphe correspondantes en syntaxe RML [5] sont disponibles dans le dépôt de code `https://github.com/Orange-OpenSource/graphameleon`.
Dans les détails, une requête HTTP est représentée par une entité de la classe `ucobs:HTTPConnectionFacet`, et ses en-têtes sont représentées par des propriétés spécifiques telles que `ucobs:startTime` et `ucobs:endTime` pour les horodatages, et `core:tag` pour les en-têtes de type Fetch Metadata [36]. Une adresse IP ou une URL pouvant être communes à plusieurs requêtes (p.ex. un utilisateur répétant le même appel à un site Web, un site Web avec divers services hébergés sur le même serveur), ces éléments sont matérialisés par l'intermédiaire des classes `ucobs:IPAddressFacet` et `ucobs:URLFacet`, respectivement. Les références croisées entre les entités résultantes sont établies grâce à des propriétés telles que `ucobs:hasFacet` et `ucobs:host`. Pour les macro-activités, nous considérons les interactions de l'utilisateur comme des instances de la classe `ucoact:ObservableAction`, avec des relations vers les entités `ucobs:HTTPConnectionFacet` et `ucobs:URLFacet` mentionnées ci-dessus pour décrire le contexte dans lequel elles se produisent. De plus, nous utilisons les propriétés `types:threadNextItem` et `types:threadPreviousItem` de UCO pour représenter la chronologie des traces d'activité.

3.2 Collecte de données avec Graphaméléon

Le navigateur Web étant l'interface principale entre un utilisateur et le Web, nous considérons pour la suite que la collecte de données doit porter à la fois sur les requêtes HTTP et des interactions utilisateur/navigateur pour comprendre et analyser pleinement le système utilisateur-réseau-application car ces deux ensembles reflètent l'intention directe et indirecte de l'utilisateur.

Collecte des requêtes. À son activation au sein du navigateur, l'outil Graphaméléon associe des fonctions de rappel aux processus d'envoi et de réception du navigateur. Cela permet d'intercepter toutes les requêtes gérées par le navigateur pour récupérer des informations à partir

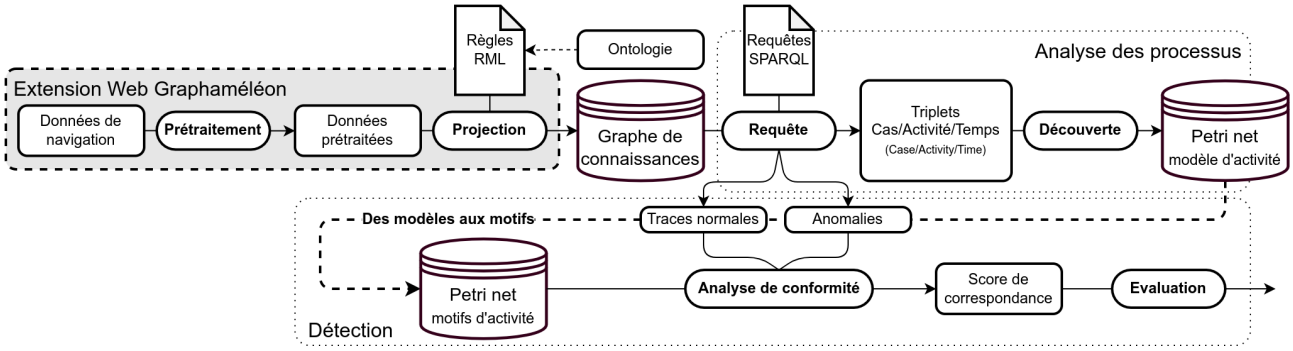


FIGURE 1 – Aperçu de la chaîne de traitement des données.
L'extension Web Graphaméléon capture et annote l'activité de l'utilisateur au niveau du navigateur Web. Un composant d'extraction des processus dérive des modèles d'activité places/transitions (Petri net) à partir du graphe de connaissances RDF résultant. Ces modèles peuvent être utilisés pour construire une bibliothèque de motifs d'activité, qui sont ensuite utilisés par un composant de vérification de conformité, côté client ou côté réseaux/serveurs, pour classer de nouvelles traces d'activité comme normales ou anormales.

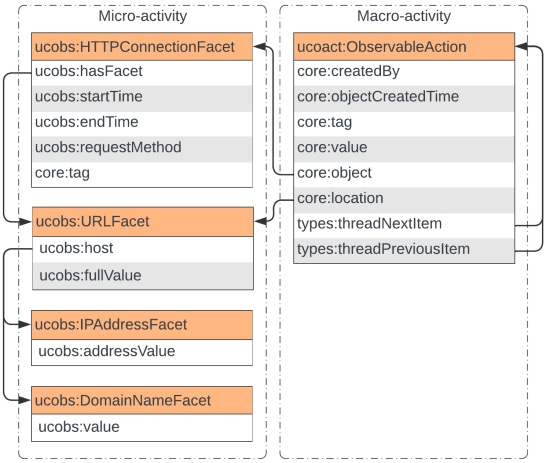


FIGURE 2 – Modèle de données.
Ce diagramme de classe définit les concepts et les propriétés utilisés pour la représentation sémantique des micro-activités (à gauche) et des macro-activités (à droite) tels que décrits dans la section 3.1. Pour les micro-activités, les classes et les propriétés présentées décrivent de manière précise une séquence de requêtes capturées au niveau du navigateur Web. Les macro-activités améliorent encore la modélisation en permettant la description des interactions. Les noms des concepts et des propriétés utilisés ici sont définis dans le vocabulaire UCO, les espaces de noms suivants s'appliquent : *core* = `https://ontology.unifiedcyberontology.org/uco/core#`, *ucobs* = `https://ontology.unifiedcyberontology.org/uco/observable#` et *types* = `https://ontology.unifiedcyberontology.org/uco/types#`.

des en-têtes de celles-ci. La Table 1 résume le type de données collectées par Graphaméléon. Ces informations incluent les URLs, les addresses IP et les noms de domaines associés, l'horodatage de la requête et les Fetch Metadata [4]. Les Fetch Metadata nous permettent de déduire des connaissances indirectes à partir des traces de navigation. Par exemple, le champ `Sec-Fetch-Site` indique la relation entre l'initiateur de la requête et sa cible, fournissant ainsi des informations sur la topologie du réseau. De même, le champ `Sec-Fetch-Mode` aide à différencier les requêtes initiées par l'utilisateur de celles correspondant à des sous-requêtes pour charger des images et autres ressources. Enfin, nous tokenisons les URLs utilisées dans les requêtes en remplaçant tous les arguments présents par les noms de leurs paramètres respectifs. Cela permet d'abstraire d'éventuelles informations de contexte définies par les sites Web, et éviter ainsi une diversité excessive dans l'interprétation des activités pour des cas similaires. Pour exemple, les URLs `https://www.shop.com/?client_id=2313` et `https://www.shop.com/?client_id=346`, indépendamment de l'utilisateur initiant ces requêtes, reflètent le même comportement. Après tokenisation, ces URLs sont représentées par `https://www.shop.com/?client_id=[client_id]`.

Portée	Paramètre ou nom de l'en-tête HTTP	**Micro**	**Macro**
Requête	Method	✓	✓
	URL	✓	✓
	IP	✓	✓
	Domain	✓	✓
	Sec-Fetch-Dest	✓	✓
	Sec-Fetch-Site	✓	✓
	Sec-Fetch-User	✓	✓
	Sec-Fetch-Mode	✓	✓
Interaction	EventType	-	✓
	Element	-	✓
	Base URL	-	✓
Les deux	User-Agent	✓	✓
	Start time	✓	✓
	End time	✓	✓

TABLE 1 – Données collectées par Graphaméléon.
Types de données collectées par l'extension Web Graphaméléon en fonction du mode de capture (micro-activité *vs* macro-activité), et regroupées selon leur portée (requête *vs* interactions *vs* les deux).

Collecte des interactions. Afin de collecter les interactions entre l'utilisateur et le navigateur, l'outil Graphaméléon lie un "script de contenu" à chaque onglet actif du navigateur. Ces scripts associent des fonctions de rappel à tous les éléments interactifs des pages Web, tels que les hyperliens, les boutons, les formulaires, etc. Cette approche minimise l'impact de la collecte sur les performances du navigateur et évite de capturer des interactions indésirables, telles que des clics erronés sur des éléments non interactifs.
Afin d'identifier les interactions, nous prenons en compte le type d'événement enregistré, l'élément avec lequel l'utilisateur a interagi, et l'URL de la ressource correspondante. Lorsqu'un élément a un attribut id, définir une référence

4. `https://www.w3.org/TR/fetch-metadata/`

vers celui-ci est évident. Ce n'est cependant pas le cas général et il est donc nécessaire de construire une référence grâce à la position absolue de l'élément dans la hiérarchie du DOM [5] ; pour exemple : `body > maindiv[2] > div > div > a`. Bien que cette méthode permette de faire référence de manière déterministe aux éléments de la page, il est important de noter que les références sous forme de chemin hiérarchique sont difficiles à interpréter sans une capture de la page Web et des interactions car ces références portent peu d'information sur la finalité de l'élément. Une alternative pour générer ces références consisterait à injecter des attributs id dans les éléments de la page Web à l'aide de fonctions de rappel, mais cela ne résout pas le problème de la stabilité des références entre chaque session de navigation pour les pages ayant un contenu dynamique.

3.3 Détection d'anomalies et réseaux de Petri

Trois familles de techniques de détection d'anomalies sont présentées dans [24] pour analyser des données de réseau représentées à l'aide d'un graphe de connaissances : *Model-Based Design*, où le graphe de connaissances contient les données nécessaires et suffisantes pour déduire les situations indésirables à l'aide de requêtes ; *Process Mining*, pour les situations liées à un modèle de décision et limitées dans le temps et l'espace, en utilisant des outils de vérification de conformité et une représentation des cas de détection sous forme de réseaux de Petri (réseaux P/T) ; *Statistical Learning* (apprentissage statistique) à l'aide de techniques de plongement de graphes [14] où les modèles d'anomalies (i.e. une généralisation du contexte pour un ensemble de situations anormales) sont dérivés de la structure du graphe de connaissances.

Dans ce travail, nous nous concentrons sur l'approche du *Process Mining* (analyse des processus), en considérant que la collecte de données à l'aide de Graphaméléon correspond à des sessions de navigation Web relativement bien définies en termes de durée et d'activités : un seul utilisateur génère une trace d'activité capturée au niveau du navigateur Web, trace qui peut être directement annotée par l'utilisateur en termes de but de l'activité à la fin de la session de navigation Web. Nous postulons que les traces d'activité sont similaires à des modèles de décision, car la séquence d'actions de l'utilisateur lors d'une session de navigation (p.ex. cliquer sur un hyperlien, utiliser le bouton de retour du navigateur Web, remplir une zone de saisie) conditionne l'atteinte d'un objectif spécifique (i.e. le but de l'activité) en fonction des informations présentées sur les pages Web. Nous supposons de même que les réseaux P/T sont une représentation adaptée pour analyser et catégoriser les traces de navigation car : *1)* ils possèdent une explicabilité intrinsèque par leur nature graphique ; *2)* les modèles de décision associés aux réseaux P/T peuvent se généraliser à différentes situations indépendamment de la représentation des connaissances sous-jacente ; *3)* les modèles de décision peuvent être facilement dérivés à partir de documents de spécifications produits par des experts métiers (p.ex. ingénieurs et techniciens réseaux), et implémentés sous forme de réseaux P/T à l'aide d'outils tels que TINA [6]. L'utilisation des réseaux P/T permet de tirer parti des techniques de détection d'anomalies couramment appelées "vérification de conformité", c'est-à-dire évaluer la pertinence d'une trace par rapport à un modèle donné (score de correspondance) ou rejouer une trace à travers un modèle pour analyser les étapes incohérentes au sein de l'activité.

Dans ce qui suit, nous définissons deux concepts pour clarifier la notion d'anomalie par rapport à ce qui est observé et à ce qui est attendu du point de vue des activités. Tout d'abord, nous définissons un "modèle d'activité" comme la traduction de toute trace d'activité (obtenue lors de la phase de collecte de données) en une représentation de type réseau P/T à l'aide d'un algorithme de découverte de processus (process mining). Selon cette définition, les collectes de données réalisées avec l'extension Web Graphaméléon (§3.2) permettent aux utilisateurs d'établir un catalogue de modèles d'activité. Ensuite, nous définissons un "motif d'activité" comme un modèle universel d'activité représenté avec des réseaux P/T. Un motif est établi en se basant soit sur une spécification du comportement attendu du couple utilisateur-système pour une situation spécifique, soit sur un comportement idéal dérivé de l'agrégation et du raffinement de plusieurs traces d'activité provenant du catalogue de modèles d'activité. Nous considérons que la gestion et la conversion des modèles d'activité en modèles relèvent de la responsabilité de l'utilisateur (analyse, sélection, raffinement), et dépasse le cadre de cet article.

La détection d'anomalies est donc définie par la comparaison d'un modèle d'activité à un motif d'activité. Ainsi, en supposant un "motif d'activité normale" (p.ex. l'authentification à une messagerie Web suivie d'une phase de consultation des e-mails), une mesure de correspondance inférieure à un seuil d'acceptation équivaut à détecter une situation anormale : $anormal \equiv correspondance_{\{alignement|rejeu\}}(\textit{modèle}, \textit{motif}) < \eta$, avec η un paramètre de seuil. Dans ce cas, nous pouvons déclencher une alerte, sans être pour autant en mesure de fournir plus de détails sur la nature de l'anomalie. En pratique, nous considérons qu'il est nécessaire de tester par rapport à un ensemble de "motifs d'activité anormaux" complémentaires dans une deuxième phase appelée phase de "qualification" afin de catégoriser l'anomalie.

4 Experimentations et résultats

Dans cette section, nous détaillons les expériences menées sur la base des approches décrites précédemment (§3), et présentons les résultats associés. Tout d'abord, nous analysons la corrélation entre le volume de triplets RDF générés par Graphaméléon et l'objet de sites Web visités (§4.1). Ensuite, nous modélisons et identifions trois scénarios de navigation Web en utilisant Graphaméléon et des réseaux P/T au sein d'un environnement contrôlé (§4.2). Les expériences sont menées à l'aide de Graphaméléon v2.1.0. Les données associées à ces expériences sont disponible sur `https://github.com/Orange-`

5. `https://developer.mozilla.org/fr/docs/Web/API/Document_Object_Model`

OpenSource/graphameleon-ds.

4.1 Trafic réseau et complexité des sites Web

Dans cette première expérience, nous cherchons à comprendre dans quelle mesure le comportement d'un site Web varie lors d'une première connexion, et de fait génère des indicateurs significatifs pour créer une signature du site utilisable ultérieurement pour la détection d'anomalies. Pour cela, nous étudions la relation entre la complexité *a priori* d'un ensemble de sites Web et les ressources téléchargées, en termes de nombre et de taille. Nous étudions cette complexité en mesurant le nombre de triplets RDF générés par Graphaméléon lors de la connexion initiale. La Table 2 présente les mesures enregistrées.

À notre connaissance, il n'existe actuellement aucune étude décrivant des groupes (clusters) de complexité de sites Web bien connus, sauf d'un point de vue marketing [35] (p.ex. secteur d'activité *vs* nombre moyen de connexions à la page d'accueil du site, poids moyen de la page en octets, indice de vitesse de chargement). De plus, avec plus d'un milliard de sites Web référencés à ce jour [19], les outils d'analyse de sites Web proposent principalement des analyses de positionnement par rapport à la concurrence [15]. Cela souligne le défi de sélectionner des exemples représentatifs pour chaque groupe. Pour cette expérience, nous proposons d'établir un corpus de sites Web organisé selon trois groupes de complexité arbitraires. L'idée sous-jacente est que la complexité est liée au volume du contenu éditorial à afficher. Pour chaque catégorie, nous sélectionnons un sous-ensemble de trois sites Web de référence sur la base d'opinions d'experts tiers :

One-Page où "Swappa Bottle" [6], "Garden Studio" [7] et "Mark My Images" [8] (MMI) sont identifiés dans [27] comme les trois meilleurs exemples de sites Web d'une seule page dont s'inspirer dans le cadre de projets de conception de sites ;

Encyclopedia où "Encyclopedia Britannica Online" [9] (EBO), "Scholarpedia" [10] et "Encyclopedia.com" [11] sont présentés dans [20] comme les trois principales alternatives à Wikipédia du point de vue de la fiabilité de l'information ;

Content-Heavy où "RTI International" [12], "PrintMag" [13] et la "International Women's Media Foundation" [14] (IWMF) sont identifiés dans [22] comme les trois principaux sites Web présentant une grande quantité de contenu tout en offrant une expérience intuitive.

Ensuite, nous réalisons la collecte et l'analyse des données de traces de navigation pour chaque page d'accueil des sites Web de la manière suivante : *1)* dans une instance de Firefox sur ordinateur (anti-pistage $\in \{stricte, standard\}$), charger Graphaméléon et activer la capture de données (mode de collecte $\in \{micro, macro\}$, type de sortie $= semantize$) ; *2)* ouvrir un onglet de navigation et la console Network Monitor [15] (mise en cache = *désactivée*) ; *3)* accéder au site Web cible en saisissant son URL dans la barre de navigation ; *4)* arrêter la capture par Graphaméléon 10 secondes après la détection de l'événement de chargement complet de la page dans la console Network Monitor pour garantir l'exécution cohérente des scripts intégrés à la page Web (i.e. l'événement `DOMContentLoaded` [16]) ; *5)* enregistrer les données dans un fichier (sérialisation $= Turtle$) ; *6)* recueillir les statistiques de collecte de données (nombre de requêtes, nombre de réponses, nombre d'interactions, nombre de sommets, nombre d'arêtes) à partir de l'interface utilisateur de Graphaméléon, ainsi que celles du graphe de connaissances résultant grâce à un ensemble de requêtes SPARQL (nombre de triplets, nombre de sujets, nombre d'instances de classe).

Site Web	CM-Trk.	TC	SC	UDN	UHC	UIP	UURL
One-Page							
Swappa Bottle	μ-Str.	n.a.	-	-	-	-	-
	μ-Std.	n.a.	-	-	-	-	-
	M-Str.	n.a.	-	-	-	-	-
Garden Studio	μ-Str.	886	163	5	84	5	69
	μ-Std.	985	189	11	89	11	78
	M-Str.	21	5	1	1	1	1
MMI	μ-Str.	427	81	3	38	3	37
	μ-Std.	423	80	3	38	3	36
	M-Str.	21	5	1	1	1	1
Encyclopedia							
EBO	μ-Str.	599	122	13	54	13	42
	μ-Std.	2195	472	71	194	70	137
	M-Str.	21	5	1	1	1	1
Scholarpedia	μ-Str.	452	111	4	55	4	48
	μ-Std.	579	143	11	64	11	57
	M-Str.	n.a.	-	-	-	-	-
Encyclopedia	μ-Str.	350	66	2	31	2	31
	μ-Std.	1483	320	44	125	144	
	M-Str.	21	5	1	1	1	1
Content-Heavy							
RTI	μ-Str.	381	76	6	33	6	31
	μ-Std.	562	118	14	48	14	42
	M-Str.	21	5	1	1	1	1
PrintMag	μ-Str.	552	111	9	47	8	47
	μ-Std.	1143	234	25	101	24	84
	M-Str.	21	5	1	1	1	1
IWMF	μ-Str.	362	72	5	31	5	31
	μ-Std.	388	78	6	33	6	6
	M-Str.	21	5	1	1	1	1

TABLE 2 – Statistiques pour l'expérimentation "Trafic réseau et complexité des sites Web".
Statistiques basées sur les modes de collecte "micro" (CM = μ) et "macro" (CM = M), et en fonction de la politique de blocage des traqueurs du navigateur Web. Abréviations : CM = mode de collecte, Trk. = politique de blocage des traqueurs (strict *vs* standard), TC = nombre de triplets, SC = nombre de sujets, UOA = nombre d'entités *ucobs :DomainNameFacet*, UDN = nombre d'entités `ucobs:DomainNameFacet`, UHC = nombre d'entités `ucobs:HTTPConnectionFacet`, UIP = nombre d'entités `ucobs:IPAddressFacet`, UURL = nombre d'entités `ucobs:URLFacet`, n.a. = non applicable.

Résultats & discussion. En utilisant cette procédure, 27 échantillons de données ont été produits (trois groupes × trois sites × trois configurations du mode de collecte), dont 23 ont permis une analyse et quatre sont inexploitables (une

6. https://swappabottle.com/
7. https://gardenestudio.com.br/
8. https://www.markmyimages.com/
9. https://www.britannica.com/
10. http://www.scholarpedia.org/
11. https://www.encyclopedia.com/
12. https://www.rti.org/
13. https://www.printmag.com/
14. https://www.iwmf.org/

15. https://firefox-source-docs.mozilla.org/devtools-user/network_monitor/
16. https://developer.mozilla.org/en-US/docs/Web/API/Document/DOMContentLoaded_event

	Stricte		Standard		Std. / Str.	
	UHC	UIP	UHC	UIP	UHC	UIP
One-Page	61.0	4.0	63.5	7.0	1.04	1.8
Encyclopedia	46.7	6.3	127.7	41.7	**2.73**	**6.6**
Content-Heavy	37.0	6.3	60.7	14.7	1.64	2.3

TABLE 3 – Moyenne du nombre d'entités en mode micro.
Comparaison de la moyenne du nombre d'entités UHC et UIP à partir de la Table 2 en fonction du niveau de complexité et de la politique anti-pistage. Seules les valeurs "Garden Studio" et "MMI" sont prises en compte pour la catégorie "One-Page". Abréviations : UHC = nombre d'entités `ucobs:HTTPConnectionFacet`, UIP = nombre d'entités `ucobs:IPAddressFacet`.

erreur d'accès `SSL_ERROR_NO_CYPHER_OVERLAP` côté serveur pour "Swappa Bottle" en mode micro et macro, et une erreur de traitement indéterminée de l'extension Web pour "Scholarpedia" en mode macro). La Table 2 présente les statistiques relatives aux triplets RDF. Pour les échantillons issus du mode macro (CM = M), nous observons que les statistiques sur les triplets RDF restent cohérentes quel que soit le site visité. Une analyse des fichiers Turtle résultants révèle également que la structure de données RDF est conforme au modèle de données de la Figure 2. En ce qui concerne le mode micro (CM = μ), les mesures présentent une variabilité significative entre chaque catégorie de complexité pour une politique d'anti-pistage donnée. La Table 3 permet de préciser ce point en présentant le nombre moyen d'entités pour les classes d'objets `ucobs:HTTPConnectionFacet` (UHC) et `ucobs:IPAddressFacet` (UIP), ce pour chaque scénario. La comparaison des valeurs moyennes du nombre d'entités en fonction de la politique d'anti-pistage (colonne "Std. / Str." dans la Table 3) révèle une augmentation du nombre moyen de connexions et de serveurs distants vers lesquels une connexion a été faite lorsque les politiques sont assouplies, et ce quel que soit le niveau de complexité. De ces mesures, nous concluons à la fois sur le bon fonctionnement de Graphaméléon et sur sa pertinence pour l'étude des primo-connexions. Bien que les groupes de complexité proposés puissent être discutés en raison de la taille limitée de l'échantillon et de la variabilité du contenu des sites Web, l'augmentation des échanges réseau en fonction des politiques d'anti-pistage fournit une base pour de futurs travaux de catégorisation par les stratégies de suivi mises en œuvre par les sites Web (tracking & analytics) et de la topologie de réseau associée.

4.2 Catégorisation de traces de navigation

Dans cette deuxième expérience, notre objectif est de catégoriser les traces de navigation Web comme comportements normaux ou anormaux. Nous analysons les trois scénarios suivants en utilisant la modélisation de macro-activité (§3.1) et les réseaux de Petri (§3.3), puis rendons compte de la capacité à identifier une anomalie.

Scénario de base (normal) : un utilisateur accède au site Web, se connecte à son compte en utilisant son nom d'utilisateur et son mot de passe, navigue vers la page "Vendre un livre", renseigne une formulaire, puis retourne à la page d'accueil où il retrouve son livre dans la liste des ventes.

Scénario alternatif (normal alternatif) : un utilisateur accède au site Web, se connecte à son compte en utilisant un système à authentification unique (SSO), navigue vers la page "Vendre un livre", renseigne un formulaire, puis retourne à la page d'accueil où il retrouve son livre.

Scénario d'attaque XSS (anormal) : un attaquant accède au site Web, se connecte à son compte en utilisant son nom d'utilisateur et son mot de passe, navigue vers la page "Vendre un livre" et effectue une injection de code dans le champ "Auteur". Enfin, il retourne à la page d'accueil où le script injecté est exécuté.

Nous utilisons une simulation de site Web de librairie en ligne afin d'être en situation d'expérience contrôlée. Cela permet une exposition intentionnelle du site à diverses vulnérabilités de sécurité (une vulnérabilité XSS dans le cas présent, une forme courante d'attaque). Cela permet de même, avant l'étude, d'étiqueter chaque élément des pages Web du site, ce qui améliore l'interprétabilité des données collectées.

Nous réalisons la collecte et l'analyse des données de traces de navigation pour chaque scénario de la manière suivante : *1)* dans une instance de Firefox sur ordinateur, charger Graphaméléon et activer la capture de données (mode de collecte $= macro$, type de sortie $= semantize$) ; *2)* ouvrir un onglet de navigation et parcourir le site Web simulé selon le scénario de navigation ; *3)* arrêter la capture par Graphaméléon et enregistrer les données dans un fichier (sérialisation $= Turtle$) ; *4)* recueillir les statistiques de collecte de données (nombre de requêtes, nombre de réponses, nombre d'interactions, nombre de sommets, nombre d'arêtes) à partir de l'interface utilisateur de Graphaméléon ; *5)* calculer le modèle d'activité à partir de la trace enregistrée en utilisant la bibliothèque PM4PY Process Mining [2] (méthode $\in \{Inductive, Alpha, LogSkeleton, Heuristic, AlphaPlus\}$) ; *6)* calculer la correspondance du modèle d'activité au motif de référence en utilisant la bibliothèque PM4PY (méthode $\in \{TokenBasedReplay, Alignment\}$). Le scénario de base, qui correspond au comportement "normal", est utilisé comme motif d'activité (i.e. le modèle d'activité du scénario de base en tant que référence).

	Base	Alternatif	Attaque XSS
Requêtes	10	13	11
Interactions	18	14	18
Nœuds	263	283	277
Arcs	404	431	426

TABLE 4 – Statistiques pour l'expérimentation "catégorisation de traces de navigation".
Statistiques en termes du nombre de requêtes réseau, des interactions de l'utilisateur avec le navigateur Web, des nœuds et des arcs du graphe de navigation résultant, tel que rapporté par l'interface utilisateur de Graphaméléon pour les scénarios de navigation définis au §4.2.

Résultats & discussion. En utilisant cette procédure, trois échantillons de données ont été produits. La Table 4 présente les statistiques relatives aux graphes de navigation résultants, et la Table 5 compare les résultats de diffé-

		Alternatif	Attaque XSS
Token-Based	Alpha	0.886	0.968
	Alpha+	0.890	0.969
	Inductive	0.923	1.000
	Heuristic	0.923	1.000
Alignement	Alpha	-	-
	Alpha+	-	-
	Inductive	0.718	0.976
	Heuristic	0.718	0.976
Log Skeleton		0.684	0.999

TABLE 5 – Scores de correspondance au motif de référence. Comparaison des scores de correspondance au motif de référence (modèle d'activité du scénario de base) pour les modèles d'activité des scénarios "alternatif" et "attaque XSS". Différentes techniques et algorithmes de vérification de conformité sont utilisés pour calculer les scores de correspondance. Les techniques "token-based" et "alignement" nécessitent une découverte préalable du modèle d'activité ; les algorithmes "Alpha/Alpha+", "Inductive" et "Heuristic Miner" sont utilisés pour cela. La technique "Log Skeleton" fournit directement les scores de correspondance en utilisant les traces d'activité.

rentes techniques d'évaluation de la correspondance au motif d'activité. Du point de vue des statistiques des graphes de navigation, nous observons que le scénario alternatif implique moins d'interactions mais plus de transactions réseau que pour le scénario de base. Cela correspond au fait que l'utilisateur n'a qu'un seul bouton à cliquer pour l'authentification, et que l'authentification est déléguée à diverses entités externes fournissant le service d'authentification. Pour le scénario "attaque XSS", le nombre d'interactions reste le même, mais le nombre de requêtes augmente d'une unité. Cela correspond à la séquence d'authentification identique à celle du scénario de base, mais avec une requête supplémentaire causée par l'injection SQL. Toujours pour ce même scénario, nous remarquons une légère variation dans les scores de correspondance (une correspondance moyenne de 98% au motif d'activité), ce qui correspond également à la requête supplémentaire causée par l'injection SQL. Nous observons en outre que cette requête supplémentaire est facilement identifiable par l'utilisation des techniques d'alignement de séquences sur les traces sémantisées, le modèle de données proposé en §3.1 et appliqué au niveau de l'extension Graphaméléon permettant en effet de standardiser l'interprétation des traces.

Par conséquent, bien que notre approche fournisse une représentation formelle des traces de navigation, nous observons que son utilisation directe n'est pas adaptée à la détection d'anomalies lorsque des micro-changements se produisent par rapport à un motif d'activité (i.e. lorsque les éléments de différenciation pour qualifier les écarts sont relativement rares dans la séquence). De même, nous remarquons que, bien que les algorithmes de découverte utilisent généralement plusieurs échantillons de traces pour générer un modèle généralisé de l'activité, nous avons dans notre cas considéré un motif parfait déduit d'une seule réalisation de trace. Or un modèle de comportement normal en situation réelle est potentiellement plus complexe. Pour exemple, lors de la saisie d'un formulaire, l'ordre de lecture conventionnel est généralement suivi. Cependant, en raison de biais cognitifs, un utilisateur pourrait le remplir dans un ordre différent tout en restant dans les limites d'un comportement normal réel.

Enfin, en prenant du recul sur la collecte de données et le traitement sémantique, nous remarquons une faible compression lexicale des données de trace de navigation en raison d'un formatage cohérent (p.ex l'URL de la requête est toujours située dans l'en-tête "url"). Cependant, cette compression concerne d'avantage la sémantique des interactions. En effet, l'un des défis de l'alignement des modèles d'activité réside dans le manque d'une méthode fiable pour identifier les éléments HTML (surtout en l'absence d'un ID explicite) à travers les navigateurs, les sessions et les utilisateurs. Ce défi devient apparent lorsque le DOM du contenu de la page change à chaque visite du site, en particulier lorsque des insertions publicitaires dynamiques se produisent.

5 Conclusions et travaux futurs

Dans ce travail, nous avons cherché des moyens d'analyser les traces d'activités de navigation sur le Web dans le but de caractériser les activités des utilisateurs et le comportement des infrastructures réseaux. Les domaines d'application types envisagés dans cette recherche sont la gestion d'incident concernant les systèmes de télécommunication, la cybersécurité, et l'ingénierie des infrastructure réseaux. Sur les bases du projet DynaGraph [23], nous avons émis l'hypothèse que les graphes de connaissances peuvent structurer de façon adéquate les données collectées sur un navigateur Web au cours de sessions de navigation, et ce dans l'idée d'une analyse avancée des traces de navigation au travers d'une modélisation sous forme de réseaux de Petri et l'utilisation des outils associés aux techniques d'analyse des processus.

Pour tester notre approche, nous avons développé les concepts de micro-activité et de macro-activité en rapport au vocabulaire UCO [40] pour la représentation sémantique des activités. Nous avons également mis au point l'outil Graphaméléon, une extension Web en open source disponible à l'adresse `https://github.com/Orange-OpenSource/graphameleon` permettant la collecte en direct de données au niveau du navigateur Web et la sémantisation des traces de navigation. Enfin, nous avons analysé des traces d'activité collectées via Graphaméléon selon un plan expérimental en deux parties. Nous avons montré, dans l'expérience d'analyse de trafic par famille de complexité des sites Web, que l'augmentation des volumes d'échanges est fonction des politiques d'anti-pistage et fournit une base de travail intéressante pour la catégorisation des sites selon les stratégies d'analyse d'audience employées et la topologie de réseau associée. Ensuite, avec l'expérience de catégorisation des traces de navigation, nous avons montré les limites de la technique de vérification de conformité pour la détection d'anomalies lorsque des micro-changements se produisent par rapport à un motif de référence. Nous avons également remarqué le défi que représente l'harmonisation des modèles d'activité en raison de l'absence d'une mé-

thode fiable pour identifier les éléments HTML au sein des navigateurs Web, notamment par comparaison entre sessions de navigation ou entre utilisateurs.

Sur la base de ces développements et résultats, nous envisageons des travaux futurs approfondissant les aspects de la cartographie du Web, de l'analyse du comportement du couple utilisateur-système, et de la détection d'anomalies. En ce qui concerne l'outil Graphaméléon, des aspects techniques spécifiques nécessitent des développements complémentaires, tels que la génération de graphe en flux, l'annotation des activités via l'interface utilisateur et la gestion simultanée de plusieurs sessions de navigation Web. En ce qui concerne l'analyse de conformité, trois options se présentent pour réduire la sensibilité de notre approche. La première consiste à partitionner le motif de référence en sous-motifs, ce qui devrait réduire l'amplitude de variation du score de correspondance en cas de non-conformité. La seconde consiste à utiliser des motifs spécifiques pour qualifier un groupe d'actions et localiser le groupe par alignement de séquence (p.ex. un motif décrivant une injection SQL plutôt qu'une description générale du comportement normal). La troisième approche consiste à pondérer l'importance des actions dans le calcul du score de correspondance activité/motif en utilisant le graphe de connaissances pour fournir du contexte (p.ex. une adresse IP source peu fréquente lors d'une attaque par injection SQL, un saut réseau impossible, un même utilisateur connecté depuis deux pays) ; l'idée étant d'utiliser une pondération qui masquerait les variations mineures dues au "bruit" par rapport aux variations causées par des erreurs réelles. Enfin, nous envisageons d'intégrer les modèles d'activité – via des vocabulaires appropriés pour les réseaux de Petri [11, 18] – dans un graphe RDF structuré par l'ontologie NORIA-O [25], ce afin de calculer des contextes d'anomalies enrichis par un processus de décision en utilisant la technique de plongement de graphes [24]. Nous analyserons notamment en quoi les modèles d'activité renforcent l'aide à la décision (p.ex. performance de la détection, interprétabilité) dans une situation de gestion d'incident avec connaissance partielle de l'activité des utilisateurs, comme cela peut être le cas lorsque l'analyse est menée côté réseaux/serveurs.

Références

[1] Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard de Melo, Claudio Gutierrez, José Emilio Labra Gayo, Sabrina Kirrane, Sebastian Neumaier, Axel Polleres, Roberto Navigli, Axel-Cyrille Ngonga Ngomo, Sabbir M. Rashid, Anisa Rula, Lukas Schmelzeisen, Juan Sequeda, Steffen Staab, and Antoine Zimmermann. Knowledge Graphs, 2020.

[2] Alessandro Berti, Sebastiaan van Zelst, and Wil van der Aalst. Process Mining for Python (pm4py) : Bridging the Gap between Process-and Data Science. In *Proceedings of the ICPM Demo Track 2019, co-located with 1st International Conference on Process Mining (ICPM 2019)*, 2019.

[3] Amélie Cordier, Marie Lefevre, Pierre-Antoine Champin, Olivier Georgeon, and Alain Mille. Trace-Based Reasoning - Modeling Interaction Traces for Reasoning on Experiences. In *The 26th International FLAIRS Conference*, 2013.

[4] Amina Annane, Nathalie Aussenac-Gilles, and Mouna Kamel. BBO : BPMN 2.0 Based Ontology for Business Process Representation. In *20th European Conference on Knowledge Management (ECKM)*, Lisbon, Portugal, 2019.

[5] Anastasia Dimou, Miel Vander Sande, Pieter Colpaert, Ruben Verborgh, Erik Mannens, and Rik Van de Walle. RML : A Generic Language for Integrated RDF Mappings of Heterogeneous Data. In *Proceedings of the Workshop on Linked Data on the Web, LDOW 2014, co-located with the 23rd International World Wide Web Conference (WWW 2014)*. CEUR-WS.org, 2014.

[6] Bernard Berthomieu, Pierre-Olivier Ribet, and François Vernadat. The tool tina – construction of abstract state spaces for petri nets and time petri nets. *International Journal of Production Research*, 2004.

[7] Caleb C. Noble and Diane J. Cook. Graph-based anomaly detection. 2003.

[8] Maria Carla Calzarossa and Luisa Massari. Analysis of header usage patterns of http request messages. In *IEEE International Conference on High Performance Computing and Communication*, 2014.

[9] Giovanna Castellano, Anna M. Fanelli, and Maria A. Torsello. *Web Usage Mining : Discovering Usage Patterns for Web Applications*. Springer Berlin Heidelberg, 2013.

[10] Pierre Dagnely, Tom Ruette, Tom Tourwé, and Elena Tsiporkova. Ontology-driven multilevel sequential pattern mining : mining for gold in event logs of photovoltaic plants. In *2018 International Conference on Intelligent Systems (IS)*, 2018.

[11] Dragan Gašević and Vladan Devedžić. Petri net ontology. *Knowledge-Based Systems*, 2006.

[12] Franck Ghitalla, Dominique Boullier, and Mathieu Jacomy. *Qu'est-Ce Que La Cartographie Du Web ?: Expéditions Scientifiques Dans l'univers Des Données Numériques et Des Réseaux*. 2021.

[13] Iman Akbari, Mohammad A. Salahuddin, Leni Ven, Noura Limam, Raouf Boutaba, Bertrand Mathieu, Stephanie Moteau, and Stéphane Tuffin. Traffic classification in an increasingly encrypted web. *Communications of the ACM*, 2022.

[14] Ines Chami, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. Machine Learning on Graphs : A Model and Comprehensive Taxonomy, 2020.

[15] James Parsons. Alexa.com is dead – here are 20 of the best alternatives. `https://www.contentpowered.com/blog/alexa-com-`

dead-alternatives/, 2023. Accessed : 2023-08-10.

[16] Johannes Koch, Carlos A. Velasco, and Philip Ackermann. Http vocabulary in rdf 1.0. W3c working group note, W3C, 2017.

[17] Jorge Munoz-Gama. *Conformance Checking and Diagnosis in Process Mining : Comparing Observed and Modeled Processes*. PhD thesis, Universitat Politècnica de Catalunya – BarcelonaTech, Barcelona, 2014.

[18] Juan C. Vidal, Manuel Lama, and Alberto Bugarın. A High-level Petri Net Ontology Compatible with PNML. 2006.

[19] Kathy Haan. Top website statistics for 2023. https://www.forbes.com/advisor/business/software/website-statistics/, 2023. Accessed : 2023-08-10.

[20] Kent Campbell. Seven free wikipedia alternatives. https://blog.reputationx.com/wikipedia-alternatives, 2023. Accessed : 2023-08-10.

[21] Pierre Laperdrix, Nataliia Bielova, Benoit Baudry, and Gildas Avoine. Browser fingerprinting : A survey, 2019.

[22] Laura Held. Examples of content heavy editorial website designs. https://www.newmediacampaigns.com/blog/best-examples-of-content-heavy-editorial-website-designs, 2021. Accessed : 2023-08-10.

[23] Lionel Tailhardat, Raphaël Troncy, and Yoan Chabot. Walks in cyberspace : Towards better web browsing and network activity analysis with 3d live graph rendering. Association for Computing Machinery, 2022.

[24] Lionel Tailhardat, Raphael Troncy, and Yoan Chabot. Leveraging knowledge graphs for classifying incident situations in ict systems. In *18th International Conference on Availability, Reliability and Security (ARES)*, 2023.

[25] Lionel Tailhardat, Yoan Chabot, and Raphaël Troncy. NORIA-O : an Ontology for Anomaly Detection and Incident Management in ICT Systems. In *Semantic Web – 21st International Conference, ESWC 2024, Hersonissos, Crete, Greece, May 26 - 30, 2024, Proceedings*, 2024.

[26] Vítor Santos Lopes and João Mendes-Moreira. A comparative analysis of data preprocessing techniques in web usage mining. In *IEEE/WIC/ACM International Conference on Web Intelligence (WI)*. IEEE, 2019.

[27] Madhu Murali. 11 examples of one-page websites to inspire you. https://blog.hubspot.com/website/11-examples-of-one-page-websites-for-inspiration, 2023. Accessed : 2023-08-10.

[28] Mathieu Lirzin and Béatrice Markhoff. Vers Une Ontologie Des Interactions HTTP. In *31emes Journées Francophones d'Ingénierie Des Connaissances*, Angers, France, 2020.

[29] Megan Katsumi and Mark Fox. iCity Transportation Planning Suite of Ontologies. Technical report, University of Toronto, 2020.

[30] Guansong Pang, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel. Deep learning for anomaly detection. *ACM Computing Surveys*, 2021.

[31] Heeryon Park and Doo-Kwon Baik. Web log session identification based on cluster-based classification. In *7th International Conference on Advanced Information Networking and Applications (AINA)*. IEEE, 2011.

[32] Stephen Pauwels and Toon Calders. Extending dynamic bayesian networks for anomaly detection in complex logs, 2018.

[33] Peter E. Kaloroumakis and Michael J. Smith. Toward a Knowledge Graph of Cybersecurity Countermeasures. Technical report, The MITRE Corporation, 2021.

[34] Sasan Saqaeeyan, Hamid Haj Seyyed Javadi, and Hossein Amirkhani. Anomaly detection in smart homes using bayesian networks. *KSII Transactions on Internet and Information Systems*, 2020.

[35] thinkwithgoogle.com. Find out how you stack up to new industry benchmarks for mobile page speed. https://think.storage.googleapis.com/docs/mobile-page-speed-new-industry-benchmarks.pdf, 2017. Accessed : 2023-08-10.

[36] W3C. Fetch metadata request headers. Working draft, W3C, July 2021.

[37] Xindong Wu, Xingquan Zhu, Gongqing Wu, and Wei Ding. Data mining with big data. *IEEE Transactions on Knowledge and Data Engineering*, 2004.

[38] Nong Ye. A markov chain model of temporal behavior for anomaly detection. 2000.

[39] Youssra Rebboud, Pasquale Lisena, and Raphael Troncy. Beyond Causality : Representing Event Relations in Knowledge Graphs. In *Knowledge Engineering and Knowledge Management*. Springer International, 2022.

[40] Zareen Syed, Ankur Padia, M. Lisa Mathews, Tim Finin, and Anupam Joshi. UCO : A Unified Cybersecurity Ontology. In *AAAI Workshop on Artificial Intelligence for Cyber Security*. AAAI Press, 2016.

[41] Yan Zhao, Liwei Deng, Xuanhao Chen, Chenjuan Guo, Bin Yang, Tung Kieu, Feiteng Huang, Torben Bach Pedersen, Kai Zheng, and Christian S. Jensen. A comparative study on unsupervised anomaly detection for time series : Experiments and analysis, 2022.

Validation temporelle explicable de faits par la découverte de contraintes temporelles complexes dans les graphes de connaissances

Thibaut Soulard, Joe Raad, Fatiha Saïs
Université Paris-Saclay, CNRS, Laboratoire Interdisciplinaire des Sciences du Numérique
91190 Gif-sur-Yvette, France

prénom.nom@lisn.fr

Résumé

La question de la détection de fausses nouvelles a pris de l'importance avec la diffusion croissante de flux d'informations non vérifiées sur le Web. Pour relever ce défi, les graphes de connaissances (GC) sont un moyen efficace pour vérifier le contenu des informations diffusées grâce aux faits structurés qu'ils contiennent. Toutefois, la validité de certains faits est dépendante d'un certain contexte temporel. Cette validation temporelle est une question qui n'a pas encore reçu beaucoup d'attention dans la littérature. Notre travail introduit une nouvelle approche interprétable et explicable qui exploite la puissance des graphes de connaissances pour classer les faits en évaluant leur validité ou leur réfutation dans un intervalle temporel. Nous avons dévelopé une approche symbolique fondée sur les relations d'Allen entre intervalles temporels et qui étend ces relations aux séquences temporelles. Nous avons évalué notre approche sur l'un des plus grands graphes de connaissances disponibles publiquement et analysé les résultats en fonction de multiples hyper-paramètres. Nous avons procédé de même pour une variante neuro-symbolique que nous avons aussi proposée.

Mots-clés

Graphe de connaissances, Véracité, Séquence temporelle, IA Hybride

Abstract

The issue of fake news has gained significance in light of the escalating flow of unverified information among users. To face this challenge, Knowledge Graphs (KGs) are a means for verifying news content from statements in the KG. However, an aspect that has yet to receive attention is the temporal validation of these statements (i.e. verifying whether a statement is true in a given time interval). Our work introduces a new interpretable and explainable approach that leverages the power of KGs to classify user-inputted facts by assessing their temporal validity or refutation. We developed a symbolic approach that is based on Allen's relations between temporal intervals and extends these relations to time sequences. We test our symbolic framework on one of the largest publicly available KGs and compare its performance across multiple hyper-parameters with the neuro-symbolic extension that we also developed.

Keywords

Knowledge Graph, Veracity, Temporal Aequence, Hybrid AI

1 Introduction

Les graphes de connaissances sont devenus des sources de données cruciales, en particulier pour le développement de nombreuses applications liées à l'intelligence artificielle. En s'appuyant sur un langage simple et standardisé tel que RDF[1], les graphes de connaissances peuvent représenter des faits et des connaissances complexes du monde réél, sous la forme de triplets : `<sujet, propriété, objet>`.

Cependant, parmi le grand nombre de graphes de connaissances publiés ces dernières années, seuls quelques-uns associent explicitement une composante temporelle à leurs faits. Cette composante temporelle, incluse par exemple dans Wikidata[2] -l'une des plus grandes bases de données publiques, est essentielle pour la qualité des bases de données, car de nombreux faits ne sont valables que pendant un intervalle de temps spécifique ou à un moment précis. Par exemple, `<Obama, headOfState, USA>` n'est valable que dans l'intervalle de temps commençant à `2009/01/20` et se terminant à `2017/01/20`.

Ces graphes de connaissances étant de plus en plus utilisés pour des applications de vérification de faits, il devient nécessaire de s'assurer de leur validité temporelle. Dans ce travail, nous proposons une approche explicable pour vérifier si un fait est valide dans un intervalle de temps donné. Cette approche peut être appliquée à tout graphe de connaissances associant un intervalle temporel à ses faits.

Nous commençons par présenter les travaux connexes (section 2) et par introduire les notations nécessaires (section 3). Nous présentons ensuite notre approche pour découvrir les contraintes d'ordre temporel dans les données (section 4), telles que la contrainte selon laquelle un député doit

1. Resource Description Framework : https://www.w3.org/RDF/
2. https://www.wikidata.org/

être élu avant d'occuper sa fonction. Ensuite, nous étendons et combinons les contraintes découvertes pour prendre une décision sur la validité temporelle d'un fait donné (section 5). Nous explorons différentes méthodes de combinaison, telles que le traitement démocratique de toutes les contraintes en utilisant un système de vote pour évaluer la validité d'un fait, ou l'entraînement d'un modèle d'apprentissage automatique pour apprendre quelles contraintes temporelles sont les plus appropriées pour valider un fait. Enfin, nous montrons les premiers résultats d'évaluation de notre approche sur différentes classes, contenant des millions de faits temporels, dans le graphe de connaissance Wikidata (section 6).

2 État de l'art

Bien que les questions liées à la spécification du temps de validité des faits ou de leur portée temporelle, soient bien connues [9] dans la communauté des bases de données relationnelles, ce n'est que ces dernières années que ce sujet a pris une importance particulière dans le développement des graphes de connaissances [3, 13]. Lorsque ces graphes prennent en compte la dynamicité des prédicats des faits dont la vérité est une fonction du temps, un fait est alors considéré comme vrai dans un laps de temps donné. Ce type d'informations sur la dynamicité des prédicats, lorsqu'elles sont disponibles, peut améliorer les résultats des approches de complétion de graphes de connaissances [11] qui peuvent tirer parti de la portée temporelle pour désambiguïser les différents candidats possibles. Cependant, ces approches dépendent à la fois de la disponibilité et de la validité de ces informations temporelles dans le graphe de connaissances et de leur validité. C'est pourquoi certaines approches ont été conçues pour générer de nouvelles informations temporelles, comme décrit dans [11]. Ces approches peuvent s'intéresser soit à la tâche d'interpolation, où l'on peut compléter des faits qui se sont déroulés dans le passé, soit à une tâche d'extrapolation où l'on peut chercher à prédire l'apparition de nouveaux faits.

Comme présenté dans l'étude [11], les approches de complétion GC existantes qui s'intéressent aux informations temporelles utilisent diverses techniques telles que celles fondées sur l'apprentissage profond ou les approches neuro-symboliques. Les approches fondées sur l'apprentissage profond peuvent utiliser *la traduction dans un espace vectoriel*, *la décomposition du tenseur*, *GCN*, *LSTM*, *GRU* pour apprendre un modèle de prédiction. Les approches neuro-symboliques telles que [14] qui sont à la fois basées sur la connaissance du domaine et l'apprentissage automatique utilisent des contraintes temporelles pour injecter la connaissance du domaine dans le modèle de prédiction. Cette approche améliore TTransE [7] et TA-TransE [6] grâce à l'introduction d'un *ordre temporel* (par exemple, `wasBornIn` $\leq$ `diedIn`) pour les prédicats de la même entité et *disjonction temporelle*. Par exemple, dans les pays monogames, une personne ne peut pas être mariée à deux individus différents au cours du même intervalle temporel. Dans [4], les auteurs exploitent l'interaction entre les intervalles temporels des prédicats pour la même entité mais s'appuient sur *Markov Logic Networks* et *Probalistic Soft Logics* pour résoudre la tâche finale.

En ce qui concerne la découverte de contraintes temporelles, il existe dans les bases de données relationnelles, et plus particulièrement dans le domaine du profilage des données [1], certaines approches qui découvrent des relations d'ordre entre les attributs, telles que [5, 12]. Mais les contraintes découvertes n'impliquent que des opérateurs arithmétiques (c'est-à-dire $\leq, <$) et n'utilisent pas d'opérateurs dédiés aux intervalles. A notre connaissance, il n'existe aucune approche permettant de découvrir des contraintes temporelles avec l'expressivité présentée dans ce travail, et il n'existe aucune approche traitant du problème de la validation de l'information temporelle en combinant et en exploitant des contraintes temporelles aussi expressives que celles utilisées le travail que nous présentons dans cet article.

3 Préliminaires

Notre approche de validation de faits temporels est conçue pour être appliquée aux graphes de connaissances temporels et repose sur l'algèbre d'Allen [2] pour la comparaison des intervalles de temps. Dans cette section, nous présentons les notions préliminaires et introduisons les notations utilisées dans la suite de l'article.

3.1 Algèbre d'Allen

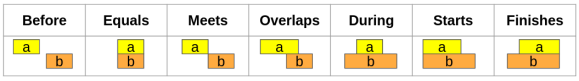


FIGURE 1 – Les 7 relations atomiques d'Allen

L'algèbre d'intervalles d'Allen est une référence pour la représentation des relations entre les intervalles de temps. Elle est composée de treize relations élémentaires qui sont distinctes, c'est-à-dire, qu'au plus une relation peut être énoncée pour une paire d'intervalles. Ces relations sont exhaustives car chaque paire d'intervalles peut être décrite par l'une des treize relations et qualitatives, dans le sens où qu'aucune durée numérique n'est prise en compte. Dans cet article nous utilisons seulement sept relations atomiques présentées dans la figure 1.

Definition 1 (L'ensemble des axiomes d'Allen)
L'ensemble des axiomes d'Allen peut être divisé en deux groupes différents : Axiomes disjoints (DA) représentant l'ensemble des axiomes où les intervalles sont disjoints, et Axiomes avec intersection (IA) représentant les axiomes où les intervalles impliquent une intersection temporelle :
– $DA = \{Before, Meets\}$,
– $IA = \{Equals, Overlaps, During, Starts, Finishes\}$.

Pour comparer deux intervalles de temps $I1$ et $I2$, nous pouvons soit spécifier la relation atomique entre eux, soit la généraliser en spécifiant s'ils sont disjoints ou s'ils ont une intersection. Par exemple, *before(I1, I2)* peut être généralisée en *DA(I1, I2)*.

3.2 Graphe de connaissances temporels

Dans ce travail, nous nous concentrons sur les graphes de connaissances (GC) temporels, c'est-à-dire les GC qui associent certains de leurs faits à une information temporelle pour exprimer l'intervalle de temps pendant lequel un fait est valide. Nous définissons tout d'abord les graphes de connaissances RDF, puis les graphes de connaissances temporels.

Definition 2 (Graphe de connaissance RDF) *Nous considérons un graphe de connaissances défini par une paire $(\mathcal{O}, \mathcal{G})$, où :*

- *$\mathcal{O} = (\mathcal{C}, \mathcal{P})$ est une ontologie représentée en OWL*[3] *et composée d'un ensemble de classes $\mathcal{C}$ et de propriétés $\mathcal{P}$.*
- *$\mathcal{G}$: est un graphe de données RDF, composé d'un ensemble de faits représentés par des triplets de la forme {(s, p, o) | $s \in \mathcal{I}$, $p \in \mathcal{P}$, $o \in \mathcal{I} \cup \mathcal{L}$}, où $\mathcal{I}$ est l'ensemble des entités (IRIs), $\mathcal{P}$ est l'ensemble des propriétés, et $\mathcal{L}$ est l'ensemble de littéraux (tels que des nombres ou des chaînes de caractères).*

Dans un graphe de connaissances composé d'un ensemble de triplets <s, p, o>, on peut distinguer trois types de faits :

- **Faits temporels concrets :** dont l'objet est de type `xsd:date` et dont la validité est illimitée dans le temps, comme : `<Mozart, dateNaissance, "1756/01/27">`.
- **Faits tautologiques :** vrais pendant toute la durée de vie d'une entité et dont le prédicat n'est pas sensible au temps, comme : `<Mozart, lieuNaissance, Salzbourg>`.
- **Faits dépendant du temps :** dont la validité est limitée à un intervalle de temps et dont le prédicat est sensible au temps, comme : `<Obama, présidentDe, USA>`.

Dans ce travail, nous nous concentrons sur *faits dépendant du temps* qui sont associés à une composante temporelle, en plus de *faits temporels concrets* qui aident à générer des contraintes temporelles.

Definition 3 (Graphe de connaissances temporels) *Nous définissons un graphe de connaissances temporel $\mathcal{TKG}$ comme un ensemble de quadruplets sous la forme de (s, p, o, t), qui étend les triples du graphe de données RDF en ajoutant la composante temporelle t exprimant la validité temporelle du fait qui peut être un un instant dans le temps ou un intervalle de temps.*

Nous considérons que information temporelle t peut être représenté par un intervalle de temps $[t; t + \epsilon]$, ϵ étant une durée insignifiante qui peut être déterminée en fonction de la granularité temporelle considérée (par exemple, des siècles, des années, des jours, des minutes). Par conséquent, dans le reste de l'article, nous nous référons à l'information temporelle en tant qu'intervalle de temps. Nous désignons par $\mathcal{T}$ l'ensemble de tous les intervalles utilisés dans TKG, chaque intervalle $I \in \mathcal{T}$ ayant une date de début et une date de fin, notées respectivement $I.s$ et $I.e$.

3. https://www.w3.org/OWL/

4 Découverte de contraintes temporelles

Pour valider un fait dans un graphe de connaissances temporel TKG, notre approche consiste à vérifier si l'information temporelle encodée pour ce fait est cohérente par rapport à une liste de contraintes temporelles. Ces contraintes peuvent exprimer soit une disjonction, soit une intersection entre les intervalles temporels des faits (voir la définition 1). Par exemple, une contrainte temporelle exprimant la disjonction peut indiquer qu'un président américain doit être élu *avant* d'occuper sa fonction, sous la forme `Before(elected, headOfState)`. Cependant, de telles contraintes temporelles sont rarement incluses dans le TKG et sont difficiles à collecter manuellement. C'est pourquoi, dans une première étape de notre approche, nous introduisons une nouvelle méthode pour découvrir ce type de contraintes temporelles à partir du TKG. Notre approche consiste d'abord à découvrir toutes les contraintes temporelles pour une seule entité, soit des contraintes simples qui peuvent être exprimées en utilisant les axiomes d'Allen (section 4.2), soit des contraintes complexes (section 4.3). Ensuite, les contraintes découvertes sont évaluées et généralisées à toutes les entités du TKG du même type (section 4.4).

4.1 Définition et comparaison des séquences temporelles

Pour chaque entité du TKG, nous pouvons construire une séquence temporelle pour chaque propriété dépendant du temps et décrivant l'entité dans le graphe (voir la définition 4). Par abus de langage, nous utiliserons parfois la formulation *un quadruplet se produit dans un intervalle de temps* pour faire référence au fait que le fait est valide dans l'intervalle de temps.

Definition 4 (Séquence temporelle) *La séquence temporelle d'une entité x pour une propriété p est l'ensemble ordonné des intervalles S des quadruplets $\{q_1, \ldots, q_n\}$, sous la forme de $< x, p, y_k, I_k >$ avec I_1 ayant la date de début la plus ancienne et I_n ayant la date de début la plus tardive dans la séquence temporelle.*

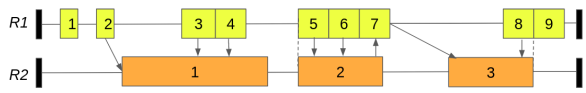


FIGURE 2 – Exemple d'une paire de deux séquences temporelles contenant respectivement 9 et 3 intervalles de temps. Les flèches renvoient à des comparaisons entre intervalles dans les séquences.

La figure 2 présente les séquences temporelles de deux propriétés $R1$ et $R2$ pour une seule entité, chaque élément de la séquence représentant un intervalle de temps. Afin de pouvoir comparer les intervalles de temps au sein d'une même

séquence temporelle et entre différentes séquences temporelles, nous nous limitons dans ce travail aux propriétés qui sont temporellement fonctionnelles (voir la définition 5).

Definition 5 (Propriété temporellement fonctionnelle) *Une propriété p est temporellement fonctionnelle si, pour chaque entité, il n'existe pas une paire d'intervalles se chevauchant dans la séquence temporelle correspondante. Nous notons $\mathcal{FP}$ l'ensemble des propriétés temporellement fonctionnelles. Une propriété p est dans $\mathcal{FP}$ ssi :*

$$\forall x \in \mathcal{I},\ \forall y_1, y_2 \in \mathcal{I} \cup \mathcal{L},\ \forall I_1, I_2 \in \mathcal{T},$$

$$< x, p, y_1, I_1 > \wedge < x, p, y_2, I_2 > \wedge DA(I_1, I_2)$$

Par exemple, `headOfState` est temporellement fonctionnelle, puisqu'un président ne peut pas être `headOfState` de deux pays différents dans des intervalles de temps qui se chevauchent.

Dans notre approche, pour une entité donnée, nous générons les séquences temporelles pour toutes ses propriétés temporellement fonctionnelles dans TKG. L'objectif est de générer des contraintes temporelles en comparant les intervalles au sein de la séquence temporelle (intra-séquence) et entre ses séquences temporelles (inter-séquence). Pour éviter la génération de contraintes bruitées et inutiles, nous limitons les comparaisons aux intervalles pertinents qui sont proches dans les séquences, sur la base des définitions suivantes :

Definition 6 (Comparaisons intra-séquence pertinentes) *Pour une séquence temporelle donnée S, les comparaisons intra-séquence pertinentes sont l'ensemble des paires d'intervalles consécutifs.*

Par exemple, dans la figure 2, il y a quatre comparaisons intra-séquence pertinentes : `Meets(3,4)`, `Meets(5,6)`, `Meets(6,7)`, et `Meets(8,9)`. Ces informations sont ensuite stocké dans la matrice $M_{\lhd}$.

Definition 7 (Comparaisons pertinentes entre séquences) *Pour une paire donnée de séquences temporelles S et S' des propriétés temporellement fonctionnelles P et P' respectivement, deux intervalles I de S et I' de S' sont considérés comme pertinents pour la comparaison si :*
$(I \cap_t I' \neq \emptyset)$
$\vee\ (I.s < I'.e)$
$\wedge(\nexists I'' \in S \setminus \{I\}, (I''.s \geq I.e \wedge I''.s \leq I'.s)$
$\wedge(\nexists I'' \in S' \setminus \{I'\}, (I''.e \leq I'.e \wedge I''.e \geq I.s)$
$\vee\ (I.s > I'.e)$
$\wedge(\nexists I'' \in S \setminus \{I\}, (I''.e \geq I'.s \wedge I''.e \leq I.s)$
$\wedge(\nexists I'' \in S' \setminus \{I'\}, (I''.s \leq I.s \wedge I''.s \geq I'.e),$
Nous désignons l'ensemble des inter-comparaisons pertinentes entre S et S' par $\Omega(S, S')$.

Dans l'exemple de la figure 2, les comparaisons inter-séquences pertinentes sont illustrées par des flèches. Elles peuvent également être représentées dans une matrice $M_{\rhd}$

Axiom	o($R_1.I$, $R_2.I$)	o($R_2.I$, $R_1.I$)
Before	2	0
Equals	0	0
Meets	0	0
Overlaps	0	1
During	3	0
Starts	1	0
Finishes	1	0

TABLE 1 – Matrice $M_{\rhd}$ pour les séquences S_1 et S_2

qui peut être utilisée pour indiquer le nombre de comparaisons inter-séquences pertinentes qui remplissent chaque axiome, comme présenté dans le tableau 1. Par exemple, les comparaisons `Starts(5,2)` et `Finishes(8,3)` représentent respectivement les seules comparaisons de début et de fin dans $M_{\rhd}$.

4.2 Découverte de contraintes temporelles simples

Sur la base du nombre d'intervalles dans chaque séquence temporelle, l'algorithme fournit les axiomes d'Allen o tels que l'expression : $\forall I$ d'une séquence S_1, $\exists\ I'$ de la séquence S_2, tel que o($S_1.I$, $S_2.I'$) est satisfaite et que $(I, I') \in \Omega(S_1, S_2)$. Nous notons que dans notre méthode, l'axiome $equal$ est assoupli en une contrainte non symétrique que nous notons $subsumes$. Dans l'exemple du tableau 1, il n'y a pas d'axiome d'Allen généralisable pour les deux séquences S_1 et S_2.

4.3 Découverte de contraintes temporelles complexes

Afin d'obtenir des contraintes temporelles plus expressives, notre algorithme procède en combinant plusieurs axiomes d'Allen. L'algorithme obtient un ensemble de contraintes temporelles complexes de différents types. D'abord les contraintes complexes qui sont fondées sur les axiomes d'Allen exprimant la disjonction temporelle (DA) et celles qui sont fondées sur les axiomes d'Allen exprimant une certaine intersection (IA). En outre, les contraintes obtenues peuvent être représentées dans un arbre d'ordonnancement, dans lequel les contraintes sont organisées grâce à la relation `isMorePrecise`. Comme dans les axiomes d'Allen, certaines de nos contraintes complexes sont symétriques, comme `NAND`, alors que `Sequence Meets` ne l'est pas. La figure 3 présente l'arbre d'ordonnancement de toutes les contraintes complexes que nous considérons.
Dans ce qui suit, nous définissons formellement les contraintes qui peuvent être découvertes par notre algorithme.

4.3.1 Contrainte NAND.

Pour deux séquences temporelles S et S', et leur ensemble correspondant d'inter-comparaisons pertinentes $\Omega(S, S')$, une contrainte `NAND` exprime que pour chaque inter-comparaison pertinente (i, i') il n'y a pas d'axiome d'intersection qui soit satisfait.

Definition 8 (Contrainte NAND) *Considérons IA l'ensemble des axiomes d'intersection (Définition 1), les deux*

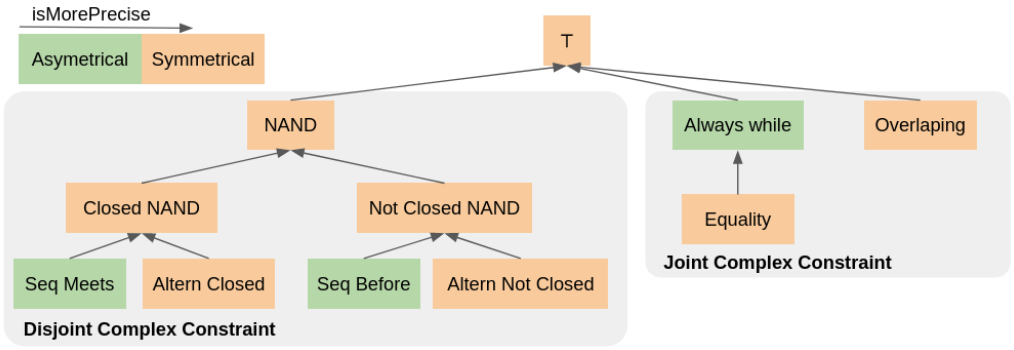


FIGURE 3 – Schema des différentes relation complexes.

séquences temporelles S et S' des propriétés P et P' respectivement, et la matrice d'inter-comparaisons $M_{\triangleleft}$ de S et S' remplit la contrainte NAND si :

$$(\sum_{a \in IA} M_{\triangleright}[a][o(P,P')]) = 0$$

4.3.2 Contrainte NAND fermée.

Pour deux séquences temporelles S et S', et leur ensemble correspondant de comparaisons pertinentes $\Omega(S,S')$, une contrainte NAND fermée exprime qu'aucune interruption n'apparaît entre le premier et le dernier quadruplet, quelle que soit la séquence temporelle.

Definition 9 *(Contrainte NAND fermée.) Considérons les deux séquences temporelles S et S' des propriétés P et P' respectivement, la matrice des inter-comparaisons $M_{\triangleright}$ de S et S', et la matrice des intra-comparaisons $M_{\triangleleft}$ de S et S' remplissent la contrainte NAND fermée si :*

$$M_{\triangleright}[Meets][o(P,P')] + M_{\triangleright}[Meets][o(P',P)] +$$
$$M_{\triangleleft}[Meets][P] + M_{\triangleleft}[Meets][P'] = |S| + |S'| - 1$$

4.3.3 Contrainte d'alternance fermée.

Pour deux séquences temporelles S et S', et leur ensemble correspondant de comparaisons pertinentes $\Omega(S,S')$, une contrainte d'*alternance fermée* exprime qu'après l'apparition d'un quadruplet d'une séquence temporelle, un quadruplet de l'autre séquence temporelle se produira.

Definition 10 *(Contrainte d'alternance fermée.) Considérons les deux séquences temporelles S et S' des propriétés P et P' respectivement, et la matrice d'inter-comparaisons $M_{\triangleright}$ de S et S' satisfait la contrainte d'*alternance fermée *si :*

$$M_{\triangleright}[Meets][o(P,P')] + M_{\triangleright}[Meets][o(P',P)]$$
$$= |S| + |S'| - 1$$

4.3.4 Contrainte de séquence coïncidente.

Pour deux séquences temporelles S et S', et leur ensemble correspondant de comparaisons pertinentes $\Omega(S,S')$, une contrainte de séquence coïncidente appelée *Sequence Meets* exprime que le dernier quadruplet de S rencontre le premier quadruplet de S'.

Definition 11 *(Contrainte de séquence coïncidente.) Considérons AA l'ensemble des axiomes conjoints (Définition 1), les deux séquences temporelles S et S' des propriétés P et P' respectivement, et la matrice d'inter-comparaisons $M_{\triangleright}$ de S et S' remplit la contrainte de séquence coïncidente si :*

$$M_{\triangleright}[meets][o(P,P')] = 1 \wedge$$
$$(\sum_{a \in AA} M_{\triangleright}[a][o(P,P')] + M_{\triangleright}[a][o(P',P)]) = 1$$

4.3.5 Contrainte NAND non fermée.

Pour deux séquences temporelles S et S', et leur ensemble correspondant de comparaisons pertinentes $\Omega(S,S')$, une contrainte *NAND non fermée* exprime qu'un espace apparaît toujours entre les intervalles (inter ou intra séquence temporelle).

Definition 12 *(Contrainte NAND non fermée). Considérons IA l'ensemble des axiomes intersectés (Définition 1), les deux séquences temporelles S et S' des propriétés P et P' respectivement, la matrice des inter-comparaisons $M_{\triangleright}$ de S et S', et la matrice des intra-comparaisons $M_{\triangleleft}$ de S et S' remplit la contrainte NAND non fermée si :*

$$M_{\triangleleft}[meets][P] + M_{\triangleleft}[meets][P'] = 0 \wedge$$
$$(\sum_{a \in AA/\{before\}} M_{\triangleright}[a][o(P,P')] + M_{\triangleright}[a][o(P',P)]) = 0$$

4.3.6 Contrainte d'alternance non fermée.

Pour deux séquences temporelles S et S', et leur ensemble correspondant de comparaisons pertinentes $\Omega(S,S')$, une contrainte *Alternance non fermée* exprime qu'après l'apparition d'un quadruplet d'une séquence temporelle, un quadruplet de l'autre séquence temporelle se produira après un intervalle.

Definition 13 *(Contrainte d'alternance non fermée.) Considérons les deux séquences temporelles S et S' des propriétés P et P' respectivement, et la matrice d'inter-comparaisons $M_{\triangleright}$ de S et S' satisfait la contrainte d'alternance non fermée si :*

$$M_{\triangleright}[before][o(P,P')] +$$
$$M_{\triangleright}[before][o(P',P)] = |S| + |S'| - 1$$

4.3.7 Contrainte de séquence-amont.

Pour deux séquences temporelles S et S', et leur ensemble correspondant de comparaisons pertinentes $\Omega(S, S')$, une contrainte séquence-amont appelée*sequence-before* exprime que le dernier quadruplet de S se produit avant tous les autres quadruplets de S'.

Definition 14 *(Contrainte de sequence-before.) Considérons IA l'ensemble des axiomes d'intersection (Définition 1), les deux séquences temporelles S et S' des propriétés P et P' respectivement, et la matrice d'inter-comparaisons $M_{\triangleright}$ de S et S' remplit la contrainte* sequence-before *ssi :*

$$M_{\triangleright}[before][o(P, P')] = 1 \wedge$$

$$(\sum_{a \in AA} M_{\triangleright}[a][o(P, P')] + M_{\triangleright}[a][o(P', P)]) = 1$$

4.3.8 Contrainte d'apparition simultanée.

Pour deux séquences temporelles S et S', et leur ensemble correspondant de comparaisons pertinentes $\Omega(S, S')$, une contrainte *d'apparition simultanée* exprime que tous les quadruplets d'une séquence temporelle partagent une intersection avec un autre quadruplet de l'autre séquence temporelle qui est égale à son intervalle temporel (c'est à dire que $q.I \cap_T q'.I = q.I$).

Definition 15 *(Contrainte d'apparition simultanée.) Étant donné la paire de séquences temporelles S et S' des propriétés P et P' respectivement, et la matrice d'inter-comparaisons $M_{\triangleright}$ de S et S' remplit la contrainte d'apparition simultanée si :*

$$M_{\triangleright}[equals][o(P, P')] + M_{\triangleright}[during][o(P, P')] +$$

$$M_{\triangleright}[starts][o(P, P')] + M_{\triangleright}[finishes][o(P, P')] = |S|$$

4.3.9 Contrainte d'égalité.

Pour deux séquences temporelles S et S', et leur ensemble correspondant de comparaisons pertinentes $\Omega(S, S')$, une contrainte d'*égalité* exprime que chaque quadruplet d'une séquence temporelle a un quadruplet dans l'autre séquence temporelle qui a le même intervalle.

Definition 16 *(Contrainte d'égalité.) Considérons les deux séquences temporelles S et S' des propriétés P et P' respectivement, et la matrice d'inter-comparaisons $M_{\triangleright}$ de S et S' satisfait la contrainte d'égalité si :*

$$M_{\triangleright}[equality][o(P, P')] +$$

$$M_{\triangleright}[equality][o(P', P)] = |S| + |S'|$$

4.3.10 Contrainte de chevauchement.

Pour deux séquences temporelles S et S', et leur ensemble correspondant de comparaisons pertinentes $\Omega(S, S')$, une contrainte *chevauchement* exprime que chaque quadruplet chevauche un quadruplet de l'autre séquence temporelle (à l'exception du quadruplet qui commence le plus tard).

Definition 17 *(Contrainte de chevauchement.) Considérons IA l'ensemble des axiomes d'inersection (Définition 1), les deux séquences temporelles S et S' des propriétés P et P' respectivement, et la matrice d'inter-comparaisons $M_{\triangleright}$ de S et S' remplit la contrainte de chevauchement si :*

$$M_{\triangleright}[overlapping][o(P, P')] +$$

$$M_{\triangleright}[overlapping][o(P', P)] = |S| + |S'| - 1$$

4.4 Généralisation des contraintes temporelles

Dans les sections précédentes, nous avons décrit comment des contraintes temporelles simples ou complexes peuvent être découvertes à partir de deux séquences temporelles pour une seule entité $e \in C$. Dans cette section, nous présentons notre approche pour généraliser les contraintes découvertes parmi l'ensemble d'entités de la classe C.
Pour évaluer si une contrainte peut être généralisée pour une classe C, nous introduisons deux mesures : le *taux d'erreur* et le *taux de généralisation*. La première permet de prendre en compte l'imperfection des données dans un graphe de connaissances temporel lorsqu'une entité est décrite avec des informations temporelles erronées, ou de surmonter la présence d'entités aberrantes qui ne suivent pas un comportement temporel similaire à celui d'autres entités dans C (voir la définition 18). Cette dernière méthode permet de filtrer les contraintes qui ne sont pas partagées par un pourcentage minimum d'entités dans C (voir la définition 19). Ensuite, étant donné un seuil d'erreur err et un seuil de généralisation gen, nous sélectionnons toutes les contraintes qui peuvent être généralisées, c'est-à-dire toutes les contraintes ayant un taux d'erreur supérieur à err et un taux de généralisation inférieur à gen (voir la définition 20). Enfin, parmi les contraintes généralisées restantes, nous ne conservons que celles dont la relation complexe la plus précise (comme décrit précédemment dans la section 4.3) afin de filtrer les contraintes temporelles redondantes.

Definition 18 (Taux d'erreur) *Étant donné la contrainte temporelle TC entre les propriétés P et P', l'ensemble des entités $E_{P,P'}$ de la classe C qui sont décrites par les deux propriétés, et le sous-ensemble d'entités $X_{P,P'} \subseteq E_{P,P'}$ où TC a été réfuté. Nous définissons le taux d'erreur comme suit :*

$$ErrorRate(TC) = \frac{|X_{P,P'}|}{|E_{P,P'}|}$$

Definition 19 (Taux de généralisation) *Étant donné la contrainte temporelle TC entre les propriétés P et P', l'ensemble des entités E de la classe C, et l'ensemble des entités $E_{P,P'} \subseteq E$ qui sont décrites par les deux propriétés. Nous définissons le taux de généralisation comme suit :*

$$GeneRate(TC) = \frac{|E_{P,P'}|}{|E|}$$

Definition 20 (Contraintes temporelles généralisées) *Étant donné un seuil d'erreur $err \in [0, 1]$ et*

un seuil de généralisation $gen \in [0, 1]$*, une contrainte de temps* TC *peut être généralisée ssi :* $ErrorRate(TC) \leq err \wedge GeneRate(TC) \geq gen$.

4.5 Extension aux propriétés non fonctionnelles temporellement

Restreindre l'approche aux propriétés du graphe de connaissances temporel qui sont temporellement fonctionnelles peut conduire à ne pas prendre en compte des contraintes pertinentes, en particulier pour certaines propriétés ayant un large éventail de types de valeurs. Par exemple, une personne peut être liée par la propriété `memberOf` à différentes valeurs (par exemple, un groupe musical ou une certain congrès), chacune désignant un type d'assertion différent dans lequel les intervalles de temps sont susceptibles de se croiser dans la séquence temporelle. Ainsi, la probabilité de découvrir des contraintes temporelles pertinentes pouvant être généralisées à toutes les entités de la classe est plus faible. Par exemple, la propriété `memberOf` peut être spécialisée par valeur en `memberOf-USACongress` afin de découvrir des contraintes plus pertinentes entre la séquence temporelle spécialisée par valeur pour cette propriété et d'autres séquences temporelles (voir la définition 21).
Par conséquent, l'extension proposée dans cette section vise à améliorer la portée et la précision de notre approche, en spécialisant toutes les propriétés qui ont une entité comme valeur. Ces propriétés spécialisées sont ensuite utilisées pour générer les contraintes de la même manière que les propriétés normales. Pour les grands graphes de connaissances temporels, le processus de spécialisation des valeurs peut être limité aux propriétés qui ont comme valeurs des entités qui sont communément partagées entre plusieurs entités.

Definition 21 (Valeur de séquence temporelle spécialisée) *La séquence temporelle spécialisée d'une entité* x *d'une propriété* p *et pour une valeur* v *dans*$\mathcal{I}$ *est l'ensemble ordonné* S *d'un ensemble de quadruplets* $\{q_1, \ldots, q_n\}$*, sous la forme de* $< x, p, v, I_k >$ *tel que* $I_1.start$ *est le plus tôt* $I_n.start$ *est le plus tard.*

5 Validation des faits temporels basée sur des contraintes

Après avoir décrit notre approche de découverte et de généralisation des contraintes temporelles, nous présentons dans cette section comment une contrainte peut être appliquée pour valider ou réfuter un fait (section 5.1). Nous décrivons ensuite comment l'ensemble des contraintes peuvent être combinées et utilisées pour valider ou réfuter un fait (section 5.2).

5.1 Application d'une seule contrainte

Pour vérifier la validité d'un fait (s, p, o, t) dans un graphe de connaissances temporel, nous recherchons toutes les contraintes temporelles qui sont pertinentes pour ce fait. Pour qu'une contrainte $tc = o(P_1, P_2)$ soit pertinente pour

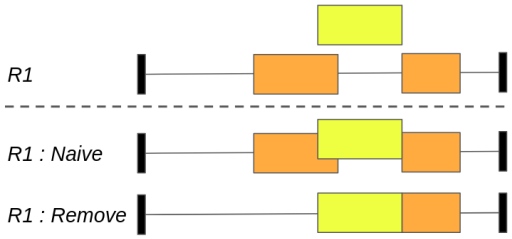


FIGURE 4 – Stratégies d'insertion dans la séquence temporelle $R1$ pour l'intervalle t du fait que nous cherchons à valider (représenté en jaune).

valider ou réfuter un fait, l'une des propriétés incluses dans une contrainte doit être liée à p : soit $P_k = p$, soit P_k représente la spécification de la valeur de p, avec $P_k \in P_1, P_2$.
Lorsqu'une contrainte temporelle est pertinente pour un fait, nous récupérons les séquences temporelles de l'entité s pour les relations P_1 et P_2. Ensuite, nous insérons le fait que nous voulons valider dans les séquences temporelles de s pour vérifier son comportement. Nous proposons deux stratégies d'insertion, illustrées dans la figure 4 : une **insertion naïve** et une **suppression de l'intersection**. Dans la première stratégie, nous ajoutons l'intervalle t du fait à la séquence temporelle de s, sans tenir compte du fait que cette insertion rompt la fonctionnalité temporelle de la propriété. Dans la stratégie de suppression de l'intersection, nous supprimons tous les intervalles de la séquence qui partagent une partie de leur ligne temporelle avec l'intervalle t.
Enfin, après l'insertion de l'intervalle du fait en utilisant l'une des deux stratégies, nous vérifions si les deux séquences temporelles sont toujours temporellement fonctionnelles et non vides. Si les deux conditions sont remplies, nous vérifions si les séquences temporelles respectent toujours la contrainte temporelle tc (ce qui indique que le fait est temporellement valide), ou si tc est maintenant violé (ce qui indique que le fait est réfuté).

5.2 Application globale des contraintes

Dans la section précédente, nous avons décrit comment la validité temporelle d'un fait peut être vérifiée par rapport à une seule contrainte temporelle. Cette section décrit comment un ensemble de contraintes temporelles peut être combiné et utilisé pour valider ou réfuter un fait. Nous proposons deux stratégies de combinaison : une *approche symbolique* basée sur un système de vote, et une *approche neuro-symbolique* qui tire parti des techniques d'apprentissage automatique pour apprendre quelles contraintes temporelles sont les plus appropriées pour valider temporellement un fait.

5.2.1 Approche symbolique par système de vote

Dans cette stratégie de combinaison, nous vérifions la validité temporelle d'un fait par rapport à toutes ses contraintes temporelles pertinentes. En utilisant un système de vote simple, nous comptons le nombre de contraintes que le fait respecte et le nombre de contraintes que le fait viole. Si le pourcentage de contraintes respectées est supérieur à

un seuil minimum, alors notre approche considère ce fait comme temporellement valide. Cette stratégie a l'avantage de rendre la décision résultante complètement explicable, c'est-à-dire que l'on peut retrouver toutes les contraintes temporelles qui ont été utilisées pour justifier la validation ou la réfutation d'un fait.

5.2.2 Approche Neuro-Symbolique

Dans cette stratégie de combinaison, nous vérifions la validité temporelle d'un fait par rapport à toutes les contraintes temporelles disponibles. Ensuite, pour chaque contrainte temporelle tc, nous associons un nombre pour représenter tous les comportements possibles : 0 si la tc est respectée par le fait ; 1 si la tc est violée ; et différentes valeurs pour indiquer si la tc n'est pas pertinente pour le fait, si la fonctionnalité temporelle de la séquence de la propriété est rompue, ou si l'une des séquences temporelles est vide. Il en résulte une matrice $n * m$, où n est la taille de l'ensemble des contraintes temporelles et m le nombre de faits à vérifier, ainsi qu'un vecteur de vérité terrain de dimension n qui peut être utilisé pour former et tester le modèle d'apprentissage automatique.

6 Évaluation expérimentale

Dans cette section, nous présentons l'évaluation expérimentale de notre approche sur une série de jeux de données. Toutes les expériences sont réalisées sur un processeur "Intel® Xeon® E5-2630 v4" avec 10 cœurs et 128 Go de RAM. Le code source et les jeux de données sont disponibles sur ce dépot github. [4]

6.1 Jeux de données

Nous évaluons notre approche sur trois jeux de données extraits extraits de de Wikidata [10], représentant tous les faits liés aux entités de type Pays (Q6256), Groupe musical (Q215380) et Homme politique (Q82955). Le tableau 2 indique le nombre d'entités pour chaque classe et le nombre total de faits temporels (quadruplets) les décrivant.

Les jeux de données ont été divisés selon un ratio de 80%, 10%, 10% pour l'ensemble d'apprentissage, de validation et de test. L'échantillonnage négatif a été effectué en changeant de manière aléatoire la partie temporelle de chaque fait autour de la durée de vie de l'entité. Ainsi, pour une entité ayant existé entre 1900 et 2000, la valeur aléatoire ne prend sa valeur qu'entre 1850 et 2050 afin de créer des faits raisonnablement faux. L'ensemble d'entraînement échantillonné non négatif est utilisé pour découvrir la contrainte temporelle, tandis que l'ensemble augmenté sert à l'entraînement de l'algorithme d'apprentissage automatique.

6.2 Réglage étape par étape

Pour évaluer les différents hyper-paramètres de notre approche, nous avons procédé étape par étape en réglant d'abord la stratégie pour la décision finale. Ensuite, le type de contrainte temporelle autorisé (le type par défaut est uniquement avec des relations), suivi de la stratégie d'insertion (la stratégie par défaut est la stratégie naïve). Puis la stratégie d'insertion (la stratégie par défaut est la stratégie naïve), et enfin le réglage du seuil d'erreur en même temps que celui de la généralisation (la valeur par défaut est $ET = 5\%$ et $GT = 5\%$). Chaque expérience est ensuite évaluée selon deux métriques : **Précision** (Acc.) et la **Couverture** (cov.) réalisée. Nous notons également le temps d'exécution (RT), de la découverte des contraintes à l'application, de chaque hyperparamètre.

4. https ://github.com/SoulardThibaut/TemporalConstraints

Classe	# Entités	# Quadruplets
Pays (Q6256)	205	183 249
Groupe de musique (Q215380)	55 507	131 476
Politicien (Q82955)	658 445	2 085 232

TABLE 2 – Description des trois ensembles de données

Classe	Deci. Type	Acc.	Cov.	RT
Country	Symbolic	79.5	9.4	**2m 30s**
	Neuro-Symb.	**80.4**	9.4	52m
Groupe de musique	Symb.	64.0	37.5	**2m 50s**
	Neuro-Symb.	**64.3**	37.5	5m 50s
Politicien	Symb.	61.6	44.0	**44m**
	Neuro-Symb.	**62.3**	44.0	1h 30m

TABLE 3 – Comparaison des stratégies de combinaison de contraintes

6.2.1 Stratégies de combinaison des contraintes

La première expérience consiste à évaluer quelle stratégie, utilisée pour combiner toutes les contraintes temporelles de validation ou de réfutation d'un fait, permet d'obtenir de meilleures performances. Le tableau 3 présente les résultats de cette expérience sur les trois classes. Il montre que la stratégie neuro-symbolique peut valider des faits avec une précision légèrement supérieure à celle de la stratégie symbolique pour toutes les classes testées. Par conséquent, nous choisirons la stratégie de combinaison neuro-symbolique pour le reste des expériences, malgré l'augmentation significative du temps d'exécution.

6.2.2 Contraintes avec spécialisation des valeurs

La deuxième expérience consiste à évaluer si l'ajout de séquences temporelles spécialisées (RxV) permet d'obtenir de meilleures performances par rapport à l'utilisation de séquences temporelles normales (R). Le tableau 4 montre les avantages de l'extension de notre approche pour la classe *Politicien* (Q6256), où le pourcentage de couverture qui peut être fait est augmenté de près de 50% (de 9,4 à 14,1) et le nombre d'évaluations exactes est augmenté de 13% (de 80,4 à 90,8). Par conséquent, nous appliquons cette extension pour les expériences restantes, malgré l'absence d'amélioration pour les deux classes restantes, car elle ne présente pas d'inconvénients.

6.2.3 Stratégie d'insertion

Cette expérience consiste à comparer les deux stratégies d'insertion présentées à la section 5.1. Le tableau 5 montre que la stratégie de suppression d'insertion réduit légère-

Classe	Const. Type	Acc.	Cov.	RT
Pays	R	80.4	9.4	**52m**
	R & RxV	**90.8**	**14.1**	2h 35m
Groupe de musique	R	64.2	37.5	5m 50s
	R & RxV	64.2	37.5	5m 50s
Politicien	R	61.6	44.0	1h 30m
	R & RxV	61.6	44.0	1h 30m

TABLE 4 – Comparaison des spécialisations

Classe	Insert Type	Acc.	Cov.	RT
Country	Naive	**90.8**	14.1	**2h 35m**
	Remove	88.5	**14.4**	2h 54m
Groupe de musique	Naive	64.2	37.5	5m 50s
	Remove	64.2	37.5	5m 50s
Politicien	Naive	**62.3**	44.0	**1h 30m**
	Remove	62.1	**46.9**	1h 32m

TABLE 5 – Comparaison des stratégies d'insertions

ment la précision de notre approche, mais qu'en contrepartie, elle augmente légèrement le pourcentage de couverture possible. Pour les expériences suivantes, nous avons utilisé la stratégie de suppression d'insertion car elle permet d'évaluer le plus grand nombre de faits.

6.2.4 Seuils d'erreur et de généralisation

Le tableau 6 présente les résultats pour différents seuils d'erreur err et de généralisation gen. Nous pouvons constater qu'une valeur de err plus élevée et une valeur de gen plus faible permettent à notre approche d'évaluer un plus grand nombre de faits, sans impact significatif sur la précision. Cependant, cela est fait au détriment d'un temps d'exécution plus élevé car l'approche dispose d'un plus grand nombre de contraintes temporelles à utiliser pour valider ou réfuter un fait. Le nombre élevé de contraintes peut entraîner des problèmes d'évolutivité en termes de mémoire, par exemple, la première ligne où la présence d'environ 24.000 contraintes temporelles crée un problème pour le classifier basé sur les arbres des décison.

Classe	gen	err	Acc.	Cov.	RT
Country	2	10	-	-	-
	2	5	**88.6**	16	8h 50m
	5	10	87.9	**17.2**	4h 30m
	5	5	88.5	14.4	**2h 54m**
Groupe de musique	2	10	**64.6**	**38.2**	6m 6s
	2	5	**64.6**	**38.2**	5m 54s
	5	10	64.2	37.5	**5m 51s**
	5	5	64.2	37.5	**5m 51s**
Politicien	2	10	63.4	**51.9**	1h 35m
	2	5	62.1	49.9	**1h 32m**
	5	10	**63.5**	48.9	1h 34m
	5	5	62.1	46.9	**1h 32m**

TABLE 6 – Comparaison des seuils d'erreur et de généralisation

7 Conclusion

Dans cet article, nous avons présenté une nouvelle approche pour évaluer la validité temporelle des faits dans un graphe de connaissances. L'approche utilise et étend l'algèbre d'intervalles d'Allen pour découvrir les contraintes temporelles du graphe de connaissances. Pour l'évaluation, nous avons procédé à une mise au point étape par étape afin d'évaluer et d'expliquer l'impact de chaque stratégie proposée dans ce travail. Grâce à ces expériences, nous avons montré que notre approche peut valider ou réfuter un fait temporel avec une grande précision (jusqu'à 90.8%), malgré une couverture relativement faible (couverture maximale de 51.9%). À l'avenir, nous nous efforcerons de résoudre les différents problèmes soulevés par les expériences. Nous prévoyons tout d'abord de réduire le nombre de caractéristiques qui ont un impact important sur les performances de l'approche neuro-symbolique, en éliminant les contraintes temporelles qui sont moins importantes. Ensuite, comme nous n'avons considéré que des ensembles de données extraits d'une seule source (Wikidata), nous n'avons pas pu évaluer si l'ensemble des contraintes temporelles découvertes dans un graphe peut être transféré et utilisé pour valider ou réfuter des faits temporels dans un autre graphe avec une précision et une couverture élevées. C'est pourquoi nous souhaitons tester la transférabilité de ces contraintes temporelles sur plusieurs autres graphes de connaissances temporels, tels que YAGO [8].

Références

[1] Ziawasch Abedjan, Lukasz Golab, Felix Naumann, and Thorsten Papenbrock. *Data Profiling*. Synthesis Lectures on Data Management. Morgan & Claypool Publishers, 2018.

[2] James F Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11) :832–843, 1983.

[3] Borui Cai, Yong Xiang, Longxiang Gao, Heng Zhang, Yunfeng Li, and Jianxin Li. Temporal knowledge graph completion : A survey. In *International Joint Conference on Artificial Intelligence*, 2022.

[4] Melisachew Chekol, Giuseppe Pirrò, Joerg Schoenfisch, and Heiner Stuckenschmidt. Marrying uncertainty and time in knowledge graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

[5] Cristian Consonni, Paolo Sottovia, Alberto Montresor, and Yannis Velegrakis. Discovering order dependencies through order compatibility. In Melanie Herschel, Helena Galhardas, Berthold Reinwald, Irini Fundulaki, Carsten Binnig, and Zoi Kaoudi, editors, *Advances in Database Technology - 22nd International Conference on Extending Database Technology, EDBT 2019, Lisbon, Portugal, March 26-29, 2019*, pages 409–420. OpenProceedings.org, 2019.

[6] Alberto García-Durán, Sebastijan Dumančić, and Mathias Niepert. Learning sequence encoders for tem-

poral knowledge graph completion. *arXiv preprint arXiv :1809.03202*, 2018.

[7] Julien Leblay and Melisachew Wudage Chekol. Deriving validity time in knowledge graph. In *Companion Proceedings of the The Web Conference 2018*, pages 1771–1776, 2018.

[8] Farzaneh Mahdisoltani, Joanna Biega, and Fabian M Suchanek. Yago3 : A knowledge base from multilingual wikipedias. In *CIDR*, 2013.

[9] Vangipuram Radhakrishna, P. V. Kumar, and V. Janaki. A survey on temporal databases and data mining. In *Proceedings of the The International Conference on Engineering & MIS 2015*, ICEMIS '15, New York, NY, USA, 2015. Association for Computing Machinery.

[10] Denny Vrandečić and Markus Krötzsch. Wikidata : A free collaborative knowledgebase. *Commun. ACM*, 57(10) :78–85, September 2014.

[11] Jiapu Wang, Boyue Wang, Meikang Qiu, Shirui Pan, Bo Xiong, Heng Liu, Linhao Luo, Tengfei Liu, Yongli Hu, Baocai Yin, and Wen Gao. A survey on temporal knowledge graph completion : Taxonomy, progress, and prospects, 2023.

[12] Renjie Xiao, Zijing Tan, Haojin Wang, and Shuai Ma. Fast approximate denial constraint discovery. *Proc. VLDB Endow.*, 16(2) :269–281, oct 2022.

[13] Jiasheng Zhang, Shuang Liang, Yongpan Sheng, and Jie Shao. Temporal knowledge graph representation learning with local and global evolutions. *Knowledge-Based Systems*, 251 :109234, 2022.

[14] Jiasheng Zhang, Yongpan Sheng, Zheng Wang, and Jie Shao. Tkgframe : a two-phase framework for temporal-aware knowledge graph completion. In *Web and Big Data : 4th International Joint Conference, APWeb-WAIM 2020, Tianjin, China, September 18-20, 2020, Proceedings, Part I 4*, pages 196–211. Springer, 2020.

sETL: Outils ETL pour la construction de graphes de connaissances en exploitant la sémantique implicite des schémas de données

S. Ouelhadj[1,2], P. Champin[1], J. Gaillard[2]

[1] Univ Lyon, UCBL, CNRS, INSA Lyon, Centrale Lyon, Univ Lyon 2, LIRIS, UMR5205, F-69622 Villeurbanne, France
[2] Métropole de Lyon, Lyon, France

{firstname.lastname}@liris.cnrs.fr; jegaillard@grandlyon.com

Résumé

Nous présentons sETL, une nouvelle approche pour la construction de graphes de connaissances (GCs) en utilisant les technologies du Web Sémantique (WS). Nous abordons les défis rencontrés par la Métropole de Lyon - une collectivité territoriale française - pour assurer l'interopérabilité des données ouvertes de sa plateforme `data.grandlyon.com`*. sETL exploite la sémantique implicite des schémas de données, et fournit une boîte à outils aux praticiens de données pour enrichir sémantiquement leurs données sans requérir des connaissances spécifiques en WS. Ce travail formalise un Modèle Sémantique, introduit le concept de Bundles pour la transformation des données, et présente une implémentation Python d'opérateurs ETL de haut niveau. L'approche est comparée à celles de l'état de l'art, mettant en évidence ses caractéristiques uniques, et sa capacité à répondre aux exigences spécifiques identifiées dans l'étude.*

Mots-clés

Enrichissement sémantique, ETL, Graphe de connaissances, Schéma de données.

Abstract

We introduces sETL, a novel approach to build knowledge graphs (KGs) using Semantic Web (SW) technologies. It addresses the challenges faced by the Metropolis of Lyon - a French local authority - in ensuring interoperability of open data from its `data.grandlyon.com` *platform. sETL leverages the implicit semantics of schemas, and provides a toolkit for data practitioners to semantically lift their data without requiring specific SW knowledge. The paper formalizes a Semantic Model, introduces the concept of Bundles for data transformation, and presents a Python implementation of high-level ETL operators. The approach is compared to related works, highlighting its unique features and ability to meet specific requirements identified in the study.*

Keywords

Semantic Lifting, ETL, Knowledge Graph, Data Schema.

1 Introduction

Les données sont continuellement produites et publiées sur le web conformément à des normes et standards, afin d'améliorer leur compréhensibilité, interopérabilité et intégration [25]. La Métropole de Lyon - une collectivité territoriale française - et ses partenaires ont embrassé le mouvement des données ouvertes, et ont mis en place un point d'accès central aux données locales qu'ils produisent, appelé `data.grandlyon.com`. Les producteurs de données de la Métropole de Lyon font des efforts considérables pour améliorer continuellement la qualité des données. Parmi ces efforts figure la participation à la production et à l'utilisation de schémas de données partagés afin de normaliser les données.

Les schémas de données permettent de décrire le modèle de données des jeux de données. Ils fournissent des descriptions précises et non ambiguës des différents champs qui composent un jeu de données : les valeurs possibles, les types, le caractère obligatoire ou non du remplissage de ces champs, etc. La production de données conformes à un schéma présente de nombreux avantages tels que la validation des données, l'amélioration de leur interopérabilité et croisement, la génération automatique de documentations, et la pérennité des modèles de données [1]. Pour cette raison, plusieurs communautés de producteurs de données, de réutilisateurs, d'experts métiers et techniques ont été constituées pour le développement de schémas de données partagés [2].

Toutefois, les défis liés à l'amélioration de la compréhensibilité, l'interopérabilité et l'intégration des données sont toujours d'actualité, car la sémantique de ces schémas de données est largement implicite. En effet, si les schémas de données capturent une sémantique partagée, notamment dans les descriptions textuelles de leurs champs qui peuvent

1. https://guides.data.gouv.fr/publier-des-donnees/guide-qualite/maitriser-les-schemas-de-donnees/comprendre-les-benefices-dutiliser-un-schema-de-donnees
2. par ex. https://schema.data.gouv.fr, https://smart-data-models.github.io/data-models, https://www.futurocite.be/standardiser-les-donnees-ouvertes/

être comprises par les producteurs et consommateurs de données, cette sémantique n'est pas accessible aux machines, et est donc moins prometteuse pour l'interopérabilité et l'intégration des données à grande échelle. Un moyen de relever ces défis est de construire des graphes de connaissances (GCs) en utilisant les technologies du Web Sémantique (WS) [2], et en exploitant la sémantique implicite des schémas de données déjà disponibles et produits par les producteurs de données. Les technologies du WS nécessitent l'utilisation de RDF [27] comme modèle de données basé sur des graphes, et d'ontologies [4] pour définir formellement la sémantique. Cela peut constituer un obstacle pour les praticiens de données qui ne sont pas familiers avec les technologies du WS, mais qui manipulent généralement des données dans des formats (semi-)structurés (ex. JSON, CSV), en utilisant des schémas de données [1, 17] au lieu d'ontologies. Ces schémas sont basés sur des spécifications techniques telles que JSON Schema [32] ou Table Schema [31].

Par conséquent, notre objectif dans le contexte de la Métropole de Lyon, et plus largement pour toute organisation productrice de données, est de permettre aux praticiens de données de construire des GCs à partir de données dans des formats (semi-)structurés. Pour ce faire, nous avons identifié 5 exigences (Ri) basées sur les conclusions d'un atelier mené avec les producteurs de données de la Métropole de Lyon [24]. Ces exigences sont les suivantes :

(R1) exploiter la sémantique implicite des schémas de données existants : les participants de notre atelier connaissent bien les schémas de données, certains s'appuient déjà sur des dictionnaires de données internes, et veulent développer de bonnes pratiques pour améliorer la qualité des données ;

(R2) impliquer des praticiens de données sans compétences approfondies en WS : aucun des participants de notre atelier n'était familier avec les concepts du WS ;

(R3) être applicable à des structures de données hétérogènes : les données qu'ils produisent sont dans des formats variés (ex. CSV, JSON, GeoJSON) ;

(R4) être en mesure de s'aligner avec les ontologies existantes : ils s'intéressent aux vocabulaires partagés développés par les instituts nationaux et les organisations gouvernementales tels que l'IGN [3], afin de fournir une compréhension commune de la signification des termes utilisés (harmonisation des termes), et de relier les données en interne et en externe ;

(R5) être en mesure de produire des ontologies manquantes lorsqu'aucune n'est disponible pour décrire les données en question : ils sont moins intéressés par les vocabulaires à usage général tels que schema.org [4] et DBpedia, dont les termes sont souvent définis vaguement. Une nouvelle ontologie légère, basée sur les éléments du schéma, est considérée comme plus souhaitable pour leurs objectifs.

A partir de ces exigences, nous proposons dans cet article la boîte à outils sETL, pour « semantic ETL » (*Extract Transform Load*). Elle est basée sur des concepts et technologies d'ingénierie des données bien connus (UML, ETL, Python, Pandas dataframes) intégrés en une nouvelle approche pour tenter d'abaisser la barrière des compétences en WS requises dans la construction de GCs, permettant ainsi une exploitation plus efficace des capacités des technologies du WS. Les contributions de ce travail sont les suivantes :

1. la formalisation d'un *Modèle Sémantique* (MS) qui exprime la sémantique implicite des données à partir des schémas fournis, et l'expose en vue d'un raffinement ultérieur ;
2. la définition du concept de *Bundle* qui permet de décomposer le jeu de données et d'en associer chaque partie avec l'élément du MS lui correspondant ;
3. la mise en œuvre d'un ensemble d'opérateurs ETL de haut niveau qui permettent d'affiner les données et leur sémantique correspondante au niveau du bundle, afin de s'adapter aux contraintes des ontologies cibles et de gagner en expressivité, avant de charger ces bundles en un GC.

Dans la suite de cet article, nous commençons par présenter un exemple pour illustrer les caractéristiques de sETL, basé sur un jeu de données ouvert existant. Dans la section 2, nous détaillons la boîte à outils ETL proposée (sETL) où nous définissons les concepts de Modèle Sémantique, Bundle, Graphes de Bundles, et comment nous appliquons dans sETL les trois phases du paradigme ETL. Ensuite, dans la section 3, nous décrivons les aspects de mise en œuvre de la boîte à outils. Puis, dans la section 4, nous passons en revue l'état de l'art des approches de construction de GCs, et les comparons à sETL à travers nos 5 critères ci-dessus. Enfin, dans la section 5, nous présentons nos conclusions et les travaux futurs possibles.

Exemple fil conducteur

Cet exemple est basé sur un jeu de données ouvert à partir du Point d'Accès National français aux données de transport. Le jeu de données décrit l'emplacement géographique et les caractéristiques techniques des Infrastructures de Recharge pour Véhicules Electriques (IRVE) [5]. Il est publié au format CSV, avec un schéma [6] exprimé selon la spécification Table Schema [31]. Ce jeu de données contient 40 colonnes. Par souci de concision, nous ne considérons que 7 colonnes : nom_operateur, contact_operateur, telephone_operateur, id_station_local, nom_station, adresse_station, implantation_station. Selon le schéma fourni, implantation_station a un ensemble défini de valeurs autorisées : "Voirie", "Parking public", "Parking privé à usage public", "Parking privé réservé à la clientèle",

3. http://data.ign.fr/data.html
4. https://schema.org/
5. https://transport.data.gouv.fr/datasets/fichier-irve-gireve?locale=fr
6. https://schema.data.gouv.fr/schemas/etalab/schema-irve-statique/2.2.0/schema-statique.json

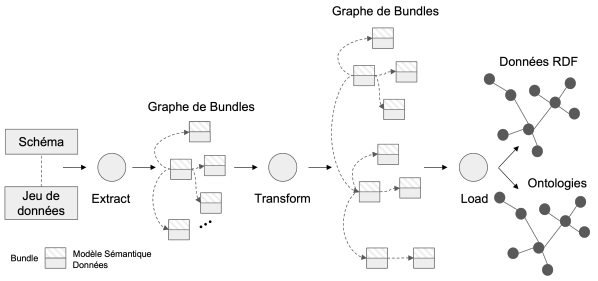


FIGURE 1 – Vue d'ensemble du processus de construction d'un graphe de connaissances avec la boîte à outils sETL.

"Station dédiée à la recharge rapide". Un échantillon du jeu de données est fourni dans le Tableau 1. Le jeu de données complet et son schéma sont disponibles dans le répertoire du projet [7].

2 sETL

La boîte à outils sETL vise à exploiter les schémas dans un processus de construction de GCs suivant le paradigme ETL. La Figure 1 présente une vue d'ensemble de ce processus. Au cours de ce processus, nous manipulons un nouveau type d'objets, appelés bundles, organisés au sein d'un graphe de bundles. Un bundle regroupe une partie des données avec son Modèle Sémantique. Ci-après, nous décrivons d'abord le Modèle Sémantique dans la section 2.1. Ensuite, nous définissons le Bundle et le Graphe de Bundles dans la section 2.2. Enfin, nous décrivons les trois phases du paradigme ETL dans le contexte de sETL. La phase *Extract* (Section 2.3) convertit un jeu de données et son schéma en un graphe de bundles. La phase *Transform* (Section 2.4) affine le graphe de bundles afin d'améliorer la sémantique des données. La phase *Load* (Section 2.5) exporte le graphe de bundles en une ontologie et des données RDF.

2.1 UML annoté : le Modèle Sémantique

Le modèle sémantique adopté dans sETL, également appelé UML annoté, est basé sur le diagramme de classes UML avec des ajouts détaillés ci-dessous. Les diagrammes de classes UML offrent un large éventail de types de composants. Pour cette première version de la boîte à outils, nous n'utilisons que les types de composants suivants : les classes (et leurs attributs), les énumérations (et leurs valeurs énumérées), et les associations. La Figure 2, partie C, illustre ces types de composants, dans un diagramme de classes UML capturant la sémantique implicite de notre exemple fil conducteur.

Ajouts de l'UML annoté

L'UML annoté étend les diagrammes de classes UML en permettant à chaque composant (classe, énumération, association, attribut et valeur énumérée) d'être annoté par un *IRI* et une *documentation textuelle* de ce composant. Les IRIs permettent de faire le lien entre l'UML annoté et les ontologies du WS. En outre, une énumération peut se voir attribuer un *type de données*, et chaque valeur énumérée peut être *alignée* avec des entités des GCs externes (ex. Wikidata, DBpedia).

Sur la base de ces considérations, nous définissons les notations suivantes pour les composants de l'UML annoté. Un Modèle Sémantique MS est un ensemble $\{MS_i \| \ i \in [0, n[\}$, où chaque MS_i est soit une $classe$, soit une $enumeration$, et où :

- chaque $classe$ a la forme $(nom,\ IRI,\ definition,\ attributs,\ associations)$;
- chaque $enumeration$ a la forme $(nom,\ IRI,\ definition,\ type,\ valeurs\_enumerees)$;
- chaque $attribut$ a la forme $(nom,\ IRI,\ definition,\ type,\ estIdentifiant)$;
- chaque $association$ a la forme $(nom,\ IRI,\ definition,\ destination)$, où $destination$ est une référence vers une $classe$ ou une $enumeration$ du MS ;
- chaque $valeur\_enumeree$ a la forme $(nom,\ IRI,\ definition,\ alignements)$;
- chaque $alignement$ a la forme $(autre\_entite,\ relation)$.

Comme tout diagramme de classes UML, un Modèle Sémantique (MS) peut être considéré comme un graphe orienté labellisé, dont les nœuds sont des classes et des énumérations, et dont les arêtes sont des associations.

2.2 Bundle et graphe de bundles

Dans sETL, les données ne sont jamais manipulées ou transformées de manière isolée : elles sont constamment liées à un Modèle Sémantique, dans des unités appelées *bundles*. En reliant chaque nœud (classe ou énumération) du Modèle Sémantique aux données correspondantes, nous obtenons une nouvelle structure : le *graphe de bundles*.

Considérons un bundle $b_i = (MS_i,\ D|MS_i)$ où MS_i est une classe ou une énumération du Modèle Sémantique, et $D|MS_i$ sont les données décrivant les instances de MS_i (notation inspirée de [3]). Un bundle peut être soit un bundle-classe ($b_i \in B_{class}$) si MS_i est une classe, soit un bundle-énumération ($b_i \in B_{enum}$) si MS_i est une énumération. Par conséquent, un graphe de bundles G_B est un graphe orienté labellisé où $B = \{(b_0, b_1, \cdots, bn);\ b_i = (MS_i,\ D|MS_i)\} = B_{class} \cup B_{enum}$.

Le Modèle Sémantique MS du graphe de bundles final représente l'ontologie sous-jacente du jeu de données en entrée. Ainsi, l'export de données RDF à partir du graphe de bundles final revient à peupler l'ontologie sous-jacente avec les données contenues dans chaque bundle du graphe de bundles. Par conséquent, nous remarquons que le graphe de bundles fournit une double vue, en faisant abstraction de la plupart des concepts du WS : une vue ontologique du jeu de données en entrée par le biais de l'UML annoté (MS), et une vue compacte et tabulaire des données RDF à produire par le biais des données contenues dans chaque bundle $D|MS_i$.

7. https://github.com/Sarra-Ouelhadj/SemanticLifting/tree/ic2024/Examples/Charging_Stations

TABLE 1 – Échantillon de données à partir de l'exemple fil conducteur

id_station_local	nom_station	adresse_station	nom_operateur	contact_operateur	telephone_operateur	implantation_station
756453	BornEco/ 63dcef1cde53 c3ec2928c1e	3 Place Maurice de Sully, Sully-sur-Loire 45600 France	Borneco | FR*BHM	technique.borneco @gmail.com	33123456789	Voirie
510419	WattzHub/ 625fc63fb907 c5cc90734800	40 Allée de la Mare Jodoin, Gif-sur-Yvette 91190 France	WattzHub | FR*SMI	contact @wattzhub.com	33185412867	Parking public

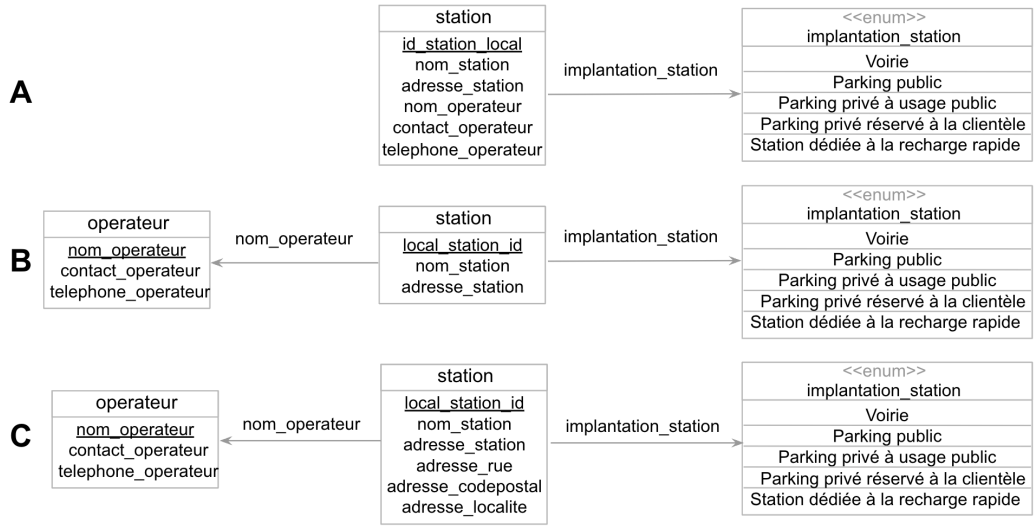


FIGURE 2 – Modèle Sémantique de notre exemple fil conducteur, à 3 phases du processus (A, B, et C).

2.3 Extracteurs

sETL extrait un graphe de bundles initial par le biais d'opérateurs appelées extracteurs à partir d'un schéma en entrée et de son jeu de données conforme. Les jeux de données dans différents formats de données sont conformes à des schémas définis à l'aide de différentes spécifications (ex. JSON Schema pour JSON, Table Schema pour CSV, etc.) Nous définissons donc un extracteur pour chaque paire de spécification de schéma et de format de données. Chaque extracteur met en correspondance chaque type d'élément du schéma source avec une construction correspondante dans notre Modèle Sémantique (MS).

Par exemple, notre extracteur Table Schema CSV appliqué sur l'exemple fil conducteur donne le résultat en Figure 2, partie A. Cet extracteur crée une classe unique pour l'ensemble de la table. Les colonnes déclarées par le schéma avec un type énuméré deviennent une association entre cette classe et une nouvelle énumération UML, également nommée d'après la colonne ("implantation_station" dans notre exemple). Toutes les autres colonnes deviennent des attributs de la classe. La $definition$ de chaque élément du MS est extraite à partir des descriptions textuelles fournies par le schéma (le cas échéant) ; il en va de même pour d'autres aspects du MS (par exemple $type$ ou $estIdentifiant$ pour les attributs). Les données liées à la classe sont la table entière. Les données liées à chaque énumération sont la colonne unique à partir de laquelle elle a été générée.

2.4 Transformeurs

Les transformeurs sont des opérateurs de raffinement disponibles pour l'utilisateur afin d'affiner le MS en un modèle plus complexe et riche sur le plan sémantique. Ces transformeurs sont appliqués sur les bundles, et sont listés dans le Tableau 2 où nous donnons de brèves descriptions de leurs effets. Par manque de place, nous ne pouvons pas donner une description détaillée de chacun d'entre eux, mais à titre d'exemple, nous donnons l'Algorithme 1 pour le transformeur *split*.

Dans l'exemple fil conducteur, nous appliquons 3 transformeurs : *split*, *apply*, et *annotate*. Premièrement, comme le bundle-classe "station" contient des données sur les opérateurs, qui sont sémantiquement différents des stations, nous divisons le bundle "station" afin d'ajouter un bundle-classe "opérateur" qui lui est lié. Les paramètres du transformeur *split* comprennent le nom (« opérateur » dans notre exemple) de la nouvelle classe, l'attribut servant d'identifiant à la nouvelle classe ("nom_operateur"), et les autres attributs et associations qui doivent être déplacés vers la nouvelle classe ("contact_operateur", et "telephone_operateur"). Le MS résultant est illustré dans la Figure 2, partie B. Les données associées à chaque bundle sont la projection correspondante du Tableau 1. Deuxièmement, nous affinons le contenu de l'attribut

TABLE 2 – Liste des transformeurs de sETL

Appliqué sur	Nom et Description
$B_{class} \cup B_{enum}$	**annotate** : assigner un IRI à un composant du MS_i. **document** : donner une définition textuelle à un composant du MS_i. **reconcile** : trouver les IRIs correspondant aux valeurs des colonnes spécifiées de chaque ligne de $D\|MS_i$ à partir des GCs externes (ex. Wikidata). **rename** : changer le nom d'un composant du MS_i, et mettre à jour les colonnes de $D\|MS_i$ en conséquence.
B_{class}	**apply** : appliquer une fonction aux valeurs des colonnes spécifiées de chaque ligne de $D\|MS_i$ et mettre à jour le MS_i en conséquence. Le(s) résultat(s) de la fonction est (sont) utilisé(s) pour remplir une (ou plusieurs) nouvelle(s) colonne(s) dans $D\|MS_i$, et le MS_i est enrichi des attributs correspondants. **mark_identifier** : marquer un attribut comme identifiant. **split** : diviser en 2 bundles-classes reliés entre eux par une association. **transform_attr_to_enum** : transformer un attribut en une énumération peuplée de toutes les valeurs de cet attribut dans $D\|MS_i$ en tant que B_{enum}.
B_{enum}	**add_value** : ajouter une valeur énumérée.

Algorithme 1 $b_1 = b_0.split(id,\ attributs,\ associations,\ nouv\_nom\_class)$

Entrées:

$b_0 = (MS_0,\ D_0|MS_0) \in B_{Class}$
$nouv\_nom\_class \in String$
$id \in MS_0.attributs$
$attributs \subset MS_0.attributs$
$associations \subseteq MS_0.associations$

soit MS_1 = class-vide
soit $D_1|MS_1$ = dataset-vide
$MS_1.nom \leftarrow nouv\_nom\_class$
$MS_1.definition \leftarrow id.definition$
$MS_0.attributs$.remove(id)
$id.estIdentifiant \leftarrow True$
$MS_1.attributs$.append(id)
$D_1|MS_1$.add_column($id.nom$)
pour $elem$ dans $attributs$ **faire**
    $MS_1.attributs$.append($elem$)
    $MS_0.attributs$.remove($elem$)
    $D_1|MS_1$.add_column($elem.nom$)
    $D_0|MS_0$.pop_column($elem.nom$)
pour $asso$ dans $associations$ **faire**
    $MS_1.associations$.append($asso$)
    $MS_0.associations$.remove($asso$)
    $D_1|MS_1$.add_column($asso.nom$)
    $D_0|MS_0$.pop_column($asso.nom$)
$b_1 \leftarrow$ BundleClass($MS_1,\ D_1|MS_1$)
$MS_0.associations$.append((nom = $id.nom$, definition = $id.definition$, destination = b_1))
retourne b_1

"adresse_station". Le schéma précise que les valeurs de cette colonne doivent contenir (1) le numéro et le nom de la rue, (2) le code postal et (3) la localité. Nous écrivons une fonction simple qui sépare ces trois composants et la transmettons au transformeur *apply*, avec les noms des trois colonnes à créer : "adresse_rue", "adresse_codepostal", "adresse_localite" (Figure 2, partie C).

Troisièmement, nous remarquons que plusieurs attributs de la classe "station" ont des termes correspondants dans l'ontologie schema.org. Nous utilisons l'opérateur *annotate* pour attacher l'IRI approprié à chacun de ces attributs. Il peut y avoir autant d'opérations de transformation que l'utilisateur le juge nécessaire. Dans notre exemple, l'utilisateur pourrait vouloir transformer les attributs de l'adresse postale en une classe distincte "adresse_postale" (en utilisant "adresse_station" comme clé primaire).

2.5 Loadeurs

Les loadeurs exportent le graphe de bundles final en un graphe de connaissances RDF. sETL fournit un loadeur d'ontologie et un loadeur de données RDF. Ces loadeurs attendent comme paramètres des espaces de noms, qui sont utilisés pour forger des IRIs pour les classes, les énumérations et les valeurs énumérées, et les instances.

Loadeur d'Ontologie. Lorsqu'un composant – classe, énumération, association, attribut ou valeur énumérée – n'est pas annoté explicitement par un IRI provenant d'une ontologie existante, un nouveau terme est créé à l'aide d'un template préconfiguré d'une ontologie (exemple dans Figure 3).

Loadeur de Données. Chaque ligne des données d'un bundle-classe représente une instance de cette classe. L'IRI de la classe est celui du MS s'il existe, sinon l'IRI forgé par le loadeur d'ontologie. Un triplet est généré pour chaque cellule du tableau des données du bundle-classe. Un exemple de triplets RDF est présenté dans le Listing 1. Le sujet de chaque ligne est l'IRI obtenu par la concaténation de l'espace de noms des instances avec la valeur de

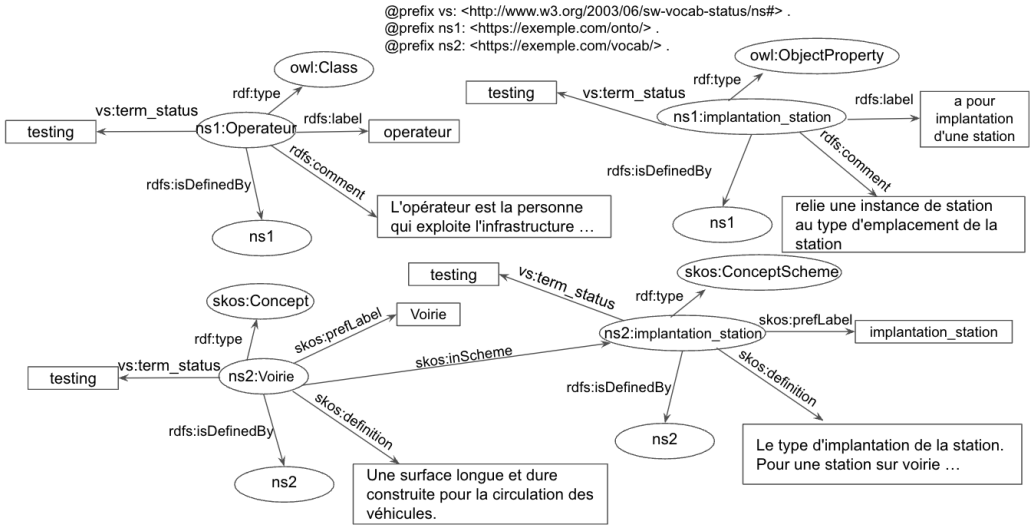


FIGURE 3 – Une partie de l'ontologie générée à partir du Modèle Sémantique du graphe de bundles illustré dans la Figure 2, partie C

```
@prefix ns1: <https://exemple.com/onto/> .
@prefix ns2: <https://exemple.com/vocab/> .
@prefix schema: <https://schema.org/> .

<https://exemple.com/id/station/756453> a ns1:
    Station ;
    schema:identifier "756453" ;
    schema:name "BornEco/63dcef1cde530c3ec2928c1e
    " ;
    schema:address "3  Place Maurice de Sully,
    Sully-sur-Loire  45600  France" ;
    schema:streetAddress "3  Place Maurice de
    Sully" ;
    schema:postalCode "45600" ;
    schema:addressLocality "Sully-sur-Loire" ;
    ns1:implantation_station ns2:Voirie ;
    ns1:nom_operateur <https://exemple.com/id/
    operateur/Borneco%20%7C%20FR%2ABHM> .

<https://exemple.com/id/operateur/Borneco%20%7C
    %20FR%2ABHM>
    a ns1:Operateur ;
    schema:name "Borneco | FR*BHM" ;
    schema:email "technique.borneco@gmail.com" ;
    schema:telephone  "33123456789" .
```

Listing 1 – Un exemple de triplets RDF générés

la colonne clé correctement échappée. Le prédicat est l'IRI de l'attribut ou de l'association correspondant à cette colonne, tel qu'annoté dans le MS ou forgé par le loadeur d'ontologie. Si la colonne correspond à un attribut, l'objet est un littéral, qui est la valeur de la colonne. Si la colonne correspond à une association, et que le bundle associé est un bundle-classe, l'objet est l'IRI identifiant l'instance correspondante au bundle-classe associé. Si la colonne correspond à une association, et que le bundle associé est un bundle-énumération, l'objet est l'IRI associé à la valeur énumérée dans la cellule actuelle.

3 Implémentation

La boîte à outils sETL est développée comme une bibliothèque Python. Les utilisateurs enrichissent leurs données en écrivant de simples scripts Python, en s'appuyant sur les classes (pour les bundles) et les fonctions (pour les opérateurs ETL) fournies par sETL. En interne, les données des bundles sont manipulées à l'aide de la bibliothèque Pandas [18] ; les loadeurs de données RDF et d'ontologies utilisent la bibliothèque RDFLib [8]. Un autre avantage de Python est la possibilité d'utiliser sETL dans une configuration interactive avec les Notebooks Jupyter [9], où les utilisateurs peuvent visualiser le graphe de bundles sous forme de diagrammes UML et inspecter les données de n'importe quel bundle, et à n'importe quelle étape de la transformation. Le code source est disponible en ligne [10], ainsi que des exemples de notebooks.

Nous avons développé 2 extracteurs jusqu'à présent : un pour les jeux de données GeoJSON conformes à JSON Schema, et un pour les jeux de données CSV conformes à Table Schema [31].

Nous avons testé avec succès sETL avec 3 jeux de données provenant de plateformes de données ouvertes françaises, chacun étant conforme à un schéma différent : les aménagements cyclables (GeoJSON/JSON Schema), lieux de stationnement (CSV/Table Schema), tous deux provenant de `schema.data.gouv.fr`, et les Infrastructures de Recharge pour Véhicules Électriques (CSV/Table Schema) provenant de `transport.data.gouv.fr`. Les résultats et pipelines correspondants sont disponibles dans le dépôt [11].

8. https://rdflib.readthedocs.io/
9. https://jupyter.org/
10. https://github.com/Sarra-Ouelhadj/SemanticLifting/
11. https://github.com/Sarra-Ouelhadj/SemanticLifting/tree/ic2024/Examples

4 Comparaison à l'état de l'art

De nombreuses approches d'enrichissement sémantique ont été proposées dans la littérature pour construire des graphes de connaissances (GCs). Ces approches peuvent être classées comme automatiques ou semi-automatiques. Les approches semi-automatiques impliquent une intervention humaine au cours d'étapes déterminées du processus de construction de GCS, tandis que les approches automatiques sont entièrement autonomes.

4.1 Approches semi-automatiques

Nous distinguons ici deux catégories : les approches déclaratives et les approches interactives. Dans la première, le mapping est fournie dans un langage déclaratif, tandis que dans la seconde, les utilisateurs construisent un mapping en interagissant avec le modèle de données.

Approches déclaratives

JSON-LD [12] et CSVW [29] permettent d'interpréter les jeux de données JSON et CSV, respectivement, en tant que RDF par le biais de fichiers de contexte. RML [7], qui étend la recommandation R2RML du W3C [6] pour prendre en charge des sources de données (semi-)structurées hétérogènes, expose les données en RDF par l'intermédiaire de mappings exprimés eux-mêmes en RDF. Le Linked data Modeling Language (LinkML) [21] est un framework de modélisation de données orienté objets basé sur une syntaxe de schéma YAML personnalisée inspirée de la structure de diagramme de classe UML. Il permet de schématiser une grande variété de formats de données en entrée, simplifiant ainsi la production de données RDF.

Au cours du processus d'enrichissement sémantique avec des approches déclaratives, des valeurs de données individuelles doivent parfois être transformées afin de s'adapter aux contraintes des ontologies cibles (ex. conversion d'unités, division ou fusion de plusieurs valeurs, etc.) Nous appelons cela une transformation fine. The Function Ontology (FnO) [20] est une description sémantique des fonctions. Elle décrit les paramètres d'une fonction, sa valeur de retour, et le problème qu'elle résout, ce qui permet de réutiliser la fonction. Les descriptions FnO s'intègrent parfaitement dans les représentations des mappings R2RML. La fonction FnO peut être utilisée comme une transformation fine des données ou comme une condition dans RML lors de l'exécution des mappings. SPARQL-Generate [16] étend le langage de requête SPARQL recommandé par le W3C [8] pour transformer des données (semi-)structurées hétérogènes en RDF à l'aide de modèles de graphes. Le mapping et les transformations fines sont exprimés grâce à l'expressivité de SPARQL. D-REPR [30] est un langage de mapping basé sur une syntaxe YAML personnalisée pour transformer des données (semi-)structurées hétérogènes en RDF. Il intègre des transformations fines de données pour le pré-traitement des données par le biais de fonctions personnalisables écrites en Python avec des paramètres codés en dur.

À l'exception de LinkML, aucune de ces approches n'exploite le schéma auquel les données en entrée peuvent se conformer (R1), ni n'est utilisable sans une bonne compréhension des technologies du WS (R2). Elles ne permettent pas non plus de générer des ontologies manquantes (R5). LinkML est différent en ce sens qu'il est basé sur un langage de schéma, qui pourrait en principe être généré à partir de schémas existants. Basé sur les concepts UML et la syntaxe YAML, il ne nécessite pas beaucoup de compétences en WS. Toutefois, les schémas LinkML sont étroitement liés à la structure des données d'origine (contrairement à nos graphes de bundles), ce qui rend difficile pour les praticiens de données de les adapter afin d'expliciter leur sémantique implicite. Par conséquent, LinkML ne satisfait que partiellement R1 et R2.

Approches interactives

OntoRefine [23] et Karma [14] permettent de faire des transformations fines des données de manière interactive à travers une représentation intermédiaire tabulaire des données en entrée. Bien que ce type de représentation intermédiaire soit familier aux praticiens de données qui n'ont pas d'expérience dans le domaine du WS, sa sémantique est pauvre par rapport aux représentations hiérarchiques ou en réseau qui mettent en évidence les concepts et leurs relations (R3), car souvent une seule ligne de structures tabulaires peut représenter plusieurs entités (voir notre exemple fil conducteur). Datalift [26], Csv2rdf4lod-automation [15], UnifiedViews [10], et LinkedPipes ETL [13] forment un autre groupe d'outils de manipulation de données qui convertissent systématiquement les données en entrée en RDF brut, puis appliquent d'autres processus d'enrichissement sémantique. Comme il faut manipuler des données RDF, l'interactivité de ces approches ne profite pas aux praticien des données qui n'ont pas de connaissances en WS (R2). En outre, aucune de ces approches n'exploite le schéma des données (R1), et ne permet de générer de nouvelles ontologies (R5).

4.2 Approches automatiques

Contrairement aux approches présentées ci-dessus, les approches automatiques visent à convertir les données sans aucune intervention humaine. Elles se répartissent en trois catégories : les mappings *hard-codés*, les mappings basés sur des schémas, et les mappings inférés.

Mappings hard-codés

Les outils dont les mappings sont hard-codés ciblent les données conformes à une spécification précise dont la sémantique a été codée en dur dans le système. gtfs-csv2rdf [12] prend en charge les données en entrée conformes à GTFS (General Transit Feed Specification) [13], et les convertit selon une ontologie prédéfinie (Linked GTFS vocabulary) [14] développée spécifiquement à cette fin. Guid-O-Matic [15] convertit les données en entrée conformes au stan-

12. https://github.com/OpenTransport/gtfs-csv2rdf
13. https://gtfs.org/
14. https://github.com/OpenTransport/linked-gtfs/blob/master/spec.md
15. https://github.com/baskaufs/guid-o-matic

dard Darwin Core [16] en RDF.

Mappings basées sur des schémas

Les outils avec mappings basés sur des schémas automatisent la génération de règles de mapping à partir de schémas. MIRROR[19], AutoMap4OBDA[28] et BOOTOX[11] sont des exemples de ces outils qui ciblent les bases de données relationnelles et, à notre connaissance, il n'existe aucune solution prenant en charge d'autres types de schémas largement utilisés.

Mappings inférés

Les solutions avec mappings inférés (également connues sous le nom d'annotation sémantique) automatisent la construction de GCs sans utiliser des règles de mapping. Au lieu de cela, elles infèrent automatiquement la correspondance des données avec un référentiel prédéfini. Par exemple, les travaux concourant au défi SemTab [17] [5, 22, 9] mettent en correspondance des éléments de données tabulaires (*i.e.*, cellules, colonnes, lignes) à des éléments sémantiques (*i.e.*, entités, classes, propriétés) provenant de GCs collaboratifs et généralistes (ex. Wikidata, DBpedia).

Aucune de ces approches automatiques n'implique d'intervention humaine (R2), ni n'est applicable à des structures de données hétérogènes (R3) puisqu'elles se concentrent sur une structure de données spécifique. Elles ne permettent pas non plus de générer de nouvelles ontologies (R5). De plus, les solutions avec des mappings inférés n'exploitent pas le schéma des données (R1). Cependant, les solutions basées sur des schémas satisfont partiellement R1 car elles sont basées sur les schémas sous-jacents des bases de données relationnelles.

4.3 Notre proposition

Nous montrons maintenant comment l'approche que nous proposons, sETL, répond aux 5 exigences présentées dans la Section 1. Premièrement, sETL tire son originalité de l'utilisation des schémas de données dans l'enrichissement sémantique des données (R1). Elle extrait le graphe de bundles initial - la représentation intermédiaire des données dans le système - à partir d'un jeu de données en entrée et de son schéma. Le Modèle Sémantique (MS) de ce graphe initial représente une transcription directe de la sémantique explicite du schéma. Deuxièmement, afin d'affiner la sémantique des données, sETL fournit un ensemble complet de transformeurs que les praticiens de données peuvent appliquer de manière interactive au graphe de bundles. Ces transformeurs garantissent la conformité continue entre le MS et les données de chaque bundle. Étant basée sur des concepts et des technologies d'ingénierie de données bien connus (UML, ETL, Python, Pandas), sETL a une faible barrière à l'entrée pour les praticiens n'ayant pas d'expérience en WS (R2). Troisièmement, le modèle de graphe de bundles est indépendant de tout format de jeux de données en entrée ou de toute spécification de schéma. Par conséquent, le système peut être étendu pour prendre en charge n'importe quelle structure de données en entrée (R3), ce qui convient à la nature hétérogène des données. Quatrièmement, sETL permet la réutilisation d'ontologies existantes en annotant le MS (R4), lorsque des vocabulaires avec la sémantique correspondante ont été identifiés, ou la génération de nouvelles ontologies légères (R5), dans les cas où aucun tel vocabulaire n'est disponible. Après l'initialisation du premier graphe de bundles au cours de la phase *Extract*, les données RDF et l'ontologie peuvent être matérialisées à tout moment. Combiné à l'utilisation interactive de sETL, cela permet un processus d'enrichissement sémantique incrémental, qui peut être utile pour des jeux de données complexes.

5 Conclusions et perspectives

Dans cet article, nous avons proposé la boîte à outils sETL qui relève le défi de l'enrichissement sémantique des données dans le contexte des organisations productrices de données n'ayant pas d'expertise étendue en WS. Cette approche a réussi à répondre aux 5 exigences que nous avons identifiées lors d'un travail antérieur réalisé avec un groupe de producteurs de données de la Métropole de Lyon. sETL permet la spécification d'un workflow complet d'enrichissement sémantique, et la production de données RDF qui en résultent. Elle comprend (1) un Modèle Sémantique étendant les diagrammes UML, pour capturer la sémantique implicite des schémas de données existants, (2) la notion de Bundle, qui groupe les éléments du MS avec leurs données correspondantes, et (3) des opérateurs ETL de haut niveau, qui font abstraction de la plupart des concepts du WS, pour le bénéfice des praticiens de données qui n'ont pas de connaissances approfondies en WS. Nous avons pu appliquer sETL à trois jeux de données complets provenant de plateformes de données ouvertes françaises.

Dans les travaux futurs, nous souhaitons étendre les fonctionnalités de la boîte à outils proposée à différents niveaux. Tout d'abord, nous souhaitons ajouter de nouveaux extracteurs pour d'autres spécifications de schémas et de formats de données (ex. les schémas de bases de données relationnelles), permettant aux utilisateurs de traiter des jeux de données plus variés et complexes. Nous souhaitons également enrichir les transformeurs. En particulier, nous souhaitons tirer parti, dans l'opérateur *reconcile*, des méthodes existantes de mapping inférés (annotation sémantique), pour faciliter la tâche d'alignement des valeurs dans les données tabulaires avec les concepts de GCs ouverts tels que Wikidata. Nous souhaitons également étudier l'ajout d'une interface graphique à sETL afin de faciliter son utilisation par les personnes qui ne maîtrisent pas l'écriture de scripts ou workflow de base sous forme de code. Enfin, nous souhaitons étudier l'intégration de sETL dans des outils existants (ex. des outils de nettoyage de données en amont, et des bases de données orientées graphe en aval) afin d'étendre les workflows existants, et de rendre sETL plus utilisable dans un environnement de production.

16. https://dwc.tdwg.org/
17. SW Challenge on Tabular Data to Knowledge Graph Matching

Références

[1] G. Aldebert and A. Augusti. schema.data.gouv.fr - An Open Data Schema Catalog for France, May 2020.

[2] S. Auer. Semantic integration and interoperability. In *Designing Data Spaces*, chapter 12, pages 195–210. Springer, Cham, 2022.

[3] F. Bariatti. *Mining Tractable Sets of Graph Patterns with the Minimum Description Length Principle*. Theses, Université de Rennes 1, November 2021.

[4] B. Chandrasekaran and et al. What are ontologies, and why do we need them ? *IEEE Intelligent Systems*, 14(01) :20–26, jan 1999.

[5] V. Cutrona and et al. Results of SemTab 2021. In *SemTab 2021*, volume 3103 of *CEUR-WS.org*, October 2021.

[6] S. Das and et al. R2RML : RDB to RDF Mapping Language. Technical report, W3C, 2012.

[7] A. Dimou and M. Vander Sande. RDF Mapping Language (RML). Technical report, RML.io, 2020.

[8] S. Harris and A. Seaborne. SPARQL 1.1 Query Language. Technical report, W3C, 2013.

[9] V-P. Huynh and et al. DAGOBAH : Table and Graph Contexts For Efficient Semantic Annotation Of Tabular Data. In *SemTab 2021*, volume 3103 of *CEUR-WS.org*, October 2021.

[10] K. Janowicz and et al. Unifiedviews : An etl tool for rdf data management. *Semantic Web*, 9(5) :661–676, jan 2018.

[11] E. Jiménez-Ruiz and et al. BootOX : Practical Mapping of RDBs to OWL 2. In *ISWC 2015*, pages 113–132. Springer, 2015.

[12] G. Kellogg and et al. JSON-LD 1.1. Technical report, W3C, July 2020.

[13] J. Klímek and et al. LinkedPipes ETL : Evolved Linked Data Preparation. In *Proc. ESWC 2016*. Springer, 2016.

[14] C. A. Knoblock and et al. Semi-automatically Mapping Structured Sources into the Semantic Web. In *Proc. ESWC 2012*, pages 375–390, Germany, 2012. Springer.

[15] T. Lebo and G.T. Williams. Converting governmental datasets into linked data. In *Proc. I-SEMANTICS 2010*, USA, 2010. ACM.

[16] M. Lefrançois and et al. A SPARQL extension for generating RDF from heterogeneous formats. In *Proc. ESWC 2017*, volume 10249, Slovenia, May 2017. Springer.

[17] Local Government Association. Open data | LG Inform Plus - Schemas, April 2023.

[18] W. McKinney. Data Structures for Statistical Computing in Python. In *Proc. SciPy*, 2010.

[19] L. F. de Medeiros and et al. MIRROR : Automatic R2RML Mapping Generation from Relational Databases. In *Proc. ICWE 2015*, pages 326–343. Springer, 2015.

[20] B. De Meester and et al. Implementation-independent function reuse. *Future Generation Computer Systems*, 110, 2020.

[21] S. Moxon and et al. The Linked Data Modeling Language (LinkML) : A General-Purpose Data Modeling Framework Grounded in Machine-Readable Semantics. In *Proc. ICBO 2021*, volume 3073 of *CEUR-WS.org*, 2021.

[22] P. Nguyen and et al. MTab4DBpedia : Semantic Annotation for Tabular Data with DBpedia. *Semantic Web Journal*, 2022.

[23] OntoText. Ontotext Refine, February 2023.

[24] S. Ouelhadj and et al. Méthode pour enrichir sémantiquement les données en utilisant l'UML annoté. EGC 2023, January 2023. Poster.

[25] M. Page and et al. *Open data maturity report 2023*. Publications Office of the European Union, LU, 2023.

[26] F. Scharffe and et al. Enabling linked data publication with the Datalift platform. In *AAAI Workshop on Semantic Cities*, Canada, July 2012. AAAI.

[27] G. Schreiber and Y. Raimond. RDF 1.1 Primer. W3C Working Group Note, W3C, June 2014.

[28] Á. Sicilia and G. Nemirovski. AutoMap4OBDA : Automated Generation of R2RML Mappings for OBDA. In *EKAW 2016*, page 577–592. Springer, 2016.

[29] J. Tennison. CSV on the Web : A Primer. Technical report, W3C, 2016.

[30] B. Vu and et al. D-repr : A language for describing and mapping diversely-structured data sources to rdf. In *Proc. K-CAP 2019*, USA, 2019. ACM.

[31] P. Walsh and R. Pollock. Table Schema. Technical report, Frictionless Standards, October 2021.

[32] A. Wright and et al. JSON Schema : A Media Type for Describing JSON Documents. Internet Draft 01, IETF, June 2022.

Fusion et intégration d'ontologies

Fusion d'ontologies biomédicales par des modèles siamois et validation par modèles de langue

S. Menad[1], S. Abdeddaïm[1], LF. Soualmia[1]

[1] Univ. Rouen Normandie, Normandie Univ, LITIS UR 4108, FR-76000 Rouen, France

{safaa.menad1,said.abdeddaim,soualfat}@univ-rouen.fr

Résumé

Dans cette étude, nous proposons de nous appuyer sur des modèles siamois afin d'intégrer dans une même ressource sémantique les ontologies les plus pertinentes dans le domaine de la santé. Un premier axe concerne les maladies, symptômes, médicaments et événements indésirables. Nos modèles neuronaux siamois sont entraînés sur des données biomédicales et génèrent de nouvelles relations sémantiques entre concepts. Nous avons exploité des ressources du domaine et un grand modèle de langue comme moyen de validation de ces nouvelles relations. Les résultats obtenus permettent d'envisager des expérimentations à plus large échelle avec d'autres ontologies du domaine.

Mots-clés

Fusion d'ontologies, Ontologies Biomédicales, Grands Modèles de Langue, Modèles Neuronaux Siamois.

Abstract

In this study, we propose to use Siamese models to integrate the most relevant ontologies in the biomedical field into a single semantic resource. The first focus is on diseases, symptoms, drugs, and adverse events. Our Siamese neural models are pre-trained on biomedical data and allow to generate new semantic relations between concepts. Domain knowledge resources and a large language model are used as a means of validating these new relationships. The results obtained allow us to envisage larger-scale experimentation with other ontologies of the domain.

Keywords

Ontology Merging, Biomedical Ontologies, Large Language Models, Siamese Neural Models.

1 Introduction

Les ontologies jouent un rôle essentiel dans la représentation, l'organisation et la compréhension des connaissances, notamment dans le domaine biomédical. Elles sont utilisées dans un grand nombre de tâches telles que la recherche d'informations, la normalisation et l'intégration de données hétérogènes. À mesure que le volume des données biomédicales augmente, les exploiter efficacement et les analyser à des fins de recherche devient de plus en plus difficile. Malgré la disponibilité d'ontologies biomédicales, elles peinent souvent à couvrir tous les concepts et relations pertinentes. Afin de combler le manque de ressource unifiée, nous proposons une méthode d'intégration d'ontologies biomédicales en utilisant l'approche sémantique SiMHOMer (Siamese Models for Health Ontologies Merging) que nous avons développée [20]. Un premier focus de l'étude s'articule autour des maladies, symptômes, médicaments et événements indésirables. La méthode s'appuie sur des modèles neuronaux siamois que nous avons spécifiquement entraînés sur des données biomédicales. L'objectif est d'identifier des relations significatives entre différents concepts et d'établir de nouvelles relations sémantiques, permettant d'obtenir une nouvelle ressource sémantique. Afin de vérifier la validité des relations générées par SiMHOMer, nous nous appuyons sur des ressources existantes, comme les relations issues du Metathesaurus de l'UMLS (Unified Medical Language System [1]) et son Semantic Network. Afin de compléter cette validation pour des relations qui n'existeraient pas dans l'UMLS, nous proposons d'exploiter un grand modèle de langue.

Les principales contributions de cette étude peuvent être résumées comme suit : i) Nous décrivons le modèle neuronal siamois que nous avons proposé dans [21] qui a montré sa performance sur d'autres tâches par rapport à d'autres modèles.. Ce modèle est entraîné sur des données biomédicales et permet de détecter les similarités sémantiques entre les concepts. Nous avions utilisé notre modèle pour intégrer l'ontologie des maladies et l'ontologie des médicaments dans une première étude [20] afin de permettre la proposition d'un médicament potentiel pour une maladie donnée ; ii) Nous développons notre approche pour générer de nouvelles relations entre d'autres ontologies (maladies et symptômes) et sources de données (OpenFDA pour les effets indésirables liés aux médicaments), iii) Enfin, nous décrivons comment nous validons les relations proposées, d'abord par l'utilisation du Metathesaurus de l'UMLS et son Semantic Network, puis par un grand modèle de langue (LLM).

2 État de l'Art

Plusieurs méthodes ont été proposées afin d'enrichir et d'intégrer des ontologies dans une même ressource. Les

1. https://www.nlm.nih.gov/research/umls/index.html

approches consistent à identifier des potentielles relations entre concepts, et vont de méthodes classiques (distance entre chaînes de caractères) [1, 6, 11] à des méthodes plus sophistiquées reposant sur l'apprentissage automatique. [4] exploitent les capacités des transformeurs pour la résolution de la tâche de correspondance d'entités, démontrant une amélioration significative par rapport aux approches classiques en apprentissage profond. Dans une démarche similaire, le système de mise en correspondance d'entités DITTO [15] propose une architecture complète, incluant des techniques de blocage et d'augmentation de données, s'appuyant sur des modèles basés sur les transformeurs. Les applications des transformeurs pour la tâche de fusion d'ontologies sont moins fréquemment utilisées que pour la tâche de mise en correspondance d'entités. [13] ont montré qu'en rajoutant un composant transformeurs dans le framework MELT [12] pour aligner deux ontologies permet d'obtenir de meilleurs résultats.

Avec l'avancée continue des techniques d'apprentissage automatique, particulièrement dans le domaine du traitement du langage naturel (TALN), les LLMs ont aussi émergé comme des alternatives majeures. Ces modèles sophistiqués, construits sur des architectures d'apprentissage profond et entraînés sur de vastes corpus de données textuelles, ont révolutionné divers domaines d'étude. Par exemple, dans certaines études des réseaux neuronaux permettent de compléter les graphes de connaissances [5]. Les LLMs conçus à des fins générales, tels que BERT [7] et GPT [24], ont fait l'objet de recherches approfondies en raison de leur efficacité dans diverses tâches liées au langage. Dans le domaine biomédical, ils ont été appliqués pour aligner des concepts sources locaux avec des terminologies cliniques standard, telles que SNOMED-CT [16] et LOINC [28]. Cependant, ces travaux étaient limités à la relation de subsomption (is-a). Dans une recherche antérieure, nous avions proposé d'utiliser nos modèles neuronaux siamois pour fusionner l'ontologie des maladies et l'ontologie des médicaments [20, 21]. Dans ce travail, nous proposons d'étendre notre approche sur d'autres ontologies du domaine de santé.

3 Approche Proposée

3.1 Modèles Siamois

Les transformeurs représentent une architecture révolutionnaire en apprentissage profond, particulièrement éminente en traitement du langage naturel. Contrairement aux modèles séquentiels traditionnels, tels que les réseaux neuronaux récurrents (RNN), les transformeurs s'appuient sur des mécanismes d'auto-attention, ce qui leur permet de capturer efficacement les dépendances globales dans les séquences d'entrée. Ce mécanisme permet à chaque mot dans une séquence d'observer tous les autres mots, permettant une meilleure compréhension contextuelle sans être limité par un traitement séquentiel. Les transformeurs ont considérablement avancé diverses tâches de NLP, y compris la traduction automatique, la génération de texte, etc.

Les transformeurs traditionnels utilisent généralement une architecture de type cross-encoder, nécessitant la combinaison de deux phrases en une seule entrée pour prédire la variable cible. Cependant, cette approche devient peu pratique lorsqu'il s'agit de traiter de nombreuses comparaisons par paires.

Les Sentence-transformeurs [26] comblent cette limitation en introduisant une approche qui génère des plongements (embeddings) pour les phrases d'entrée. Ces embeddings encapsulent des informations sémantiques sur les phrases, garantissant que deux textes ayant des significations similaires sont positionnés à proximité dans l'espace d'embedding. La méthode implique d'entraîner simultanément deux modèles de transformeurs, en utilisant une architecture de réseau siamois, permettant l'extraction de représentations de phrases significatives favorables à l'évaluation de similarité sémantique, et facilitant des tâches de NLP. Pour chaque entrée, le modèle produit un vecteur de taille fixe (u et v). La fonction objectif est choisie de telle sorte que l'angle entre les deux vecteurs u et v soit plus petit lorsque les entrées sont similaires. La fonction objectif utilise le cosinus de l'angle :

$$cos(u, v) = \frac{u.v}{||u||||v||} \quad (1)$$

Si $cos(u, v) = 1$, les phrases sont similaires et si $cos(u, v) = 0$, les phrases n'ont aucun lien sémantique.

3.2 Modèles Proposés

Les transformeurs siamois fonctionnent bien dans le domaine général, mais pas dans les domaines de spécialité, comme le domaine biomédical. Nous avons donc besoin de modèles entraînés sur des données biomédicales.

Nous avons proposé un nouveau modèle siamois BioS-Transformers [21] pré-entraîné sur le corpus d'articles scientifiques en anglais PubMed. Les transformeurs siamois ont été initialement conçus pour transformer des phrases (de taille similaire) en vecteurs. Dans notre approche, nous proposons de transformer les termes du thésaurus MeSH (Medical Subject Headings), les titres et les résumés des articles PubMed dans le même espace vectoriel en entraînant un modèle de transformeur siamois sur ces données. Nous voulons assurer un espace de correspondance entre le texte court et le texte long dans ce même vecteur. Le modèle est entraîné avec des paires d'entrées (titre, terme MeSH) et (résumé, terme MeSH). Sur la base de ces données, nous avons construit notre modèle basé sur un transformeur pré-entraîné sur des données biomédicales.

Dans cette étude, nous utilisons le modèle spécifique entraîné basé sur le Bio_ClinicalBERT [2] pour la tâche de fusion d'ontologies. La construction de BioSTransformers s'est inspirée du modèle Sentence-BERT [26] en remplaçant BERT par d'autres transformeurs. Nous avons utilisé des transformeurs qui ont été entraînés sur des données biomédicales (bio-transformers) pour créer des transformeurs siamois en ajoutant une couche de pooling et en modifiant la fonction objectif. La couche de pooling calcule le vecteur moyen des vecteurs de sortie du transformeur (embeddings de tokens). Les deux textes d'entrée passent successivement à travers le transformeur, produisant deux vecteurs u et v à

la sortie de la couche de pooling, qui sont ensuite utilisés par la fonction objectif.

Dans un Sentence-transformeur, les données supervisées sont représentées par des triplés (*phrase 1, phrase 2, score de similarité*) où le *score de similarité* est calculé entre les deux phrases *phrase 1* et *phrase 2*. Dans notre cas, comme il n'y a pas de score, ni pour les résumés, ni pour les titres et les termes MeSH correspondants, nous avons considéré :

- qu'un résumé, un titre et les termes MeSH associés au même article (identifié par un PMID) sont similaires, et le score est égal à 1 ;
- qu'un résumé (ou un titre) avec des termes MeSH non associés au même article ne sont pas similaires, et le score est égal à 0.

Nous avons utilisé une fonction objectif d'apprentissage contrastif auto-supervisé basée sur la fonction de perte Multiple Negative Ranking Loss (MNRL) dans le paquetage Sentence-Transformers [2]. La fonction MNRL nécessite des couples positifs en entrée (le titre ou le résumé et un terme MeSH associés au même article dans notre cas). Pour un couple positif (titre_i ou résumé_i, MeSH_i) la fonction MNRL considère que chaque couple (titre_i or résumé_i, MeSH_j) avec $i \neq j$ dans le même batch est négatif. Comme un article peut être associé à plusieurs termes MeSH, nous nous assurons que dans le batch de génération qu'un titre (ou résumé) associé à un terme MeSH dans un article PubMed ne soit jamais considéré comme négatif.

4 Fusion d'ontologies

4.1 Éléments Clés

Nous nous sommes inspirés des définitions [23, 8, 22] et les avons adaptées ci-dessous pour notre contexte de fusion d'ontologies biomédicales.

Définition d'ontologie : une ontologie O_i est un ensemble de vocabulaire défini au moyen de taxonomies pour décrire un domaine d'intérêt donné. Ce vocabulaire est considéré comme un ensemble d'éléments $e_i =< C_i, R_i, I_i >$; avec C_i étant l'ensemble des classes, R_i agrégeant les relations pour relier les classes, et I_i rassemblant l'ensemble des instances pour interpréter les classes et les relier avec R_i. Une ontologie O_i est également enrichie sémantiquement avec X_i pour définir des axiomes qui formalisent les classes en utilisant des langages logiques tels que les logiques de description.

Alignement d'ontologies : un alignement décrit l'ensembles des correspondances entre deux ontologies. Formellement, étant donné deux ontologies O_1 et O_2, nous limitons la définition d'un alignement A à un ensemble de triplés. Chaque triplé est spécifié par la terminologie de la relation binaire $r(e_1, e_2)$; où r représente la relation entre les deux éléments $e_1 \in O_1$ et $e_2 \in O_2$. En conséquence, l'alignement est le processus de recherche de ces ensembles de correspondances. Un score de confiance c peut également être ajouté au triplé de correspondance pour mesurer la similarité entre e_1 et e_2 (par exemple, la valeur de $c \in$ [0,1]).

Processus d'alignement : il est défini comme une fonction d'alignement ayant plusieurs paramètres calculant la similarité entre les entités. $F_m(O_1, O_2, A_j, P_c, B)$ est une fonction d'alignement avec P_c comme paramètre qui contient la valeur de confiance de similarité et B l'ensemble des ressources externes utilisées pour identifier un possible alignement A_j entre l'élément e_1 et e_2.

Fusion d'ontologies : suivant le travail présenté dans [22], nous définissons la fusion d'ontologies comme l'enrichissement sémantique d'une ontologie cible O_1 en utilisant des éléments d'une ontologie source O_2. Le résultat obtenu est une nouvelle ontologie O_3 grâce à l'alignement $A =< r_j, e_{1,j}, e_{2,j}, c_j >$

4.2 Portée des travaux

Dans la section suivante, nous décrivons les ontologies de santé que nous avons utilisées pour cette étude.

4.2.1 Ontologie des maladies

L'ontologie des maladies humaines (DOID) [17] décrit les maladies et le vocabulaire médical grâce à l'alignement de plusieurs ressources externes. Elle a été initialement construite en utilisant la Classification Internationale des Maladies (CIM-9) comme vocabulaire fondamental. Les premières versions ont été largement réorganisées par processus, système affecté et causes (troubles génétiques, maladies infectieuses, troubles métaboliques). Les révisions ultérieures ont été améliorées avec la réorganisation de DOID basée sur les concepts des maladies de l'UMLS en conjonction avec les mappings de concepts de termes vers l'ontologie SNOMED-CT (Systematized Nomenclature of Medicine- Clinical terms) et la classification CIM-9 [17]. Ces mappings se basent sur les identifiants uniques de concepts (CUI) de l'UMLS de chaque terme de maladie. Son développement a été motivé par la nécessité de représenter les connaissances avec une richesse sémantique qui permet de lier des données biomédicales à des gènes et des maladies. DOID permet ainsi d'identifier, d'intégrer et de relier des concepts de maladies synonymes qui sont inclus dans le MeSH, la SNOMED-CT, OMIM (Online Mendelian Inheritance in Man qui relie maladies et gène), et la CIM-9. Les mappings de vocabulaire sont mis à jour deux fois par an à partir d'une extraction des CUI de termes du fichier de mapping de vocabulaire MRCONSO.RRF de l'ULMS. Elle contient 13 910 concepts.

4.2.2 Ontologie des médicaments

L'ontologie des médicaments (DRON) [10] est un dictionnaire d'entités moléculaires décrivant des composants chimiques. Les entités moléculaires en question sont soit des produits naturels, soit des produits synthétiques. En plus des entités moléculaires, ChEBI (Chemical Entities of Biological Interest) contient des groupes (parties des entités moléculaires) et des classes d'entités. Ce dictionnaire comprend donc une classification ontologique, dans laquelle les relations entre les entités moléculaires ou les classes d'entités et leurs parents et/ou enfants sont spécifiées. Elle

2. https://www.sbert.net/docs/package_reference/losses.html\#multiplenegativesrankingloss

```
<owl:Class rdf:about="http://purl.obolibrary.org/obo/DOID_0112374">
    <rdfs:subClassOf rdf:resource="http://purl.obolibrary.org/obo/DOID_0050557"/>
    <obo:IAO_0000115>A congenital muscular dystrophy characterized by muscular dystrophy resulting from defective
    glycosylation of dystroglycan.</obo:IAO_0000115>
    <oboInOwl:hasDbXref>ICD10CM:G71.2</oboInOwl:hasDbXref>
    <oboInOwl:hasDbXref>ORDO:370953</oboInOwl:hasDbXref>
    <oboInOwl:hasExactSynonym>CMD due to dystroglycanopathy</oboInOwl:hasExactSynonym>
    <oboInOwl:hasExactSynonym>MDDG</oboInOwl:hasExactSynonym>
    <oboInOwl:hasExactSynonym>congenital muscular dystrophy due to dystroglycanopathy</oboInOwl:hasExactSynonym>
    <oboInOwl:hasOBONamespace>disease_ontology</oboInOwl:hasOBONamespace>
    <oboInOwl:id>DOID:0112374</oboInOwl:id>
    <oboInOwl:inSubset rdf:resource="http://purl.obolibrary.org/obo/doid#DO_rare_slim"/>
    <rdfs:label>muscular dystrophy-dystroglycanopathy</rdfs:label>
</owl:Class>
```

FIGURE 1 – Exemple d'une classe de l'ontologie de maladies DOID.

contient 8 282 concepts.

4.2.3 Ontologie des symptômes

L'ontologie des symptômes [3] (SYMP) a été développée comme une ontologie normalisée pour les symptômes des maladies humaines, à l'École de médecine de l'université du Maryland, à l'Institut des sciences du génome. Elle contient des symptômes, avec leur définition, libellés et synonymes. Elle est composée de 1 013 concepts.

4.2.4 Événements indésirables dûs aux médicaments

À notre connaissance, il n'existe aucune ontologie intégrant à la fois les médicaments et leurs effets indésirables. Par conséquent, nous avons choisi d'utiliser la base de données OpenFDA pour combler cette lacune. L'OpenFDA [4] [14] est une initiative de la Food and Drug Administration (FDA) des États-Unis qui offre un accès public à des ensembles de données et à des API liées aux produits réglementés par la FDA. Elle vise à promouvoir la transparence des données, à faciliter la recherche et l'analyse, à surveiller la sécurité des produits et à encourager le développement d'applications. Elle propose notamment de nombreuses API : événements indésirables liés aux médicaments, événements indésirables liés aux dispositifs médicaux, étiquettes de médicaments, classifications de dispositifs, rappels de produits et rapports d'application de la loi. En interrogeant ces APIs, il est possible d'accéder aux informations sur les événements indésirables liés aux médicaments. Pour répondre à nos questions de recherche, nous avons utilisé l'API des événements indésirables liés aux médicaments à partir de `https://open.fda.gov/apis/drug/event/`. Nous avons par exemple utilisé les requêtes suivantes :

```
https://api.fda.gov/drug/event.json?
    search=patient.drug.activesubstance.
    activesubstancename:"Collagen"&limit
    =1

https://api.fda.gov/drug/event.json?
    search=patient.drug.openfda.
    brand_name:"concerta"&limit=1
```

3. https://www.ebi.ac.uk/ols4/ontologies/symp?viewMode=list
4. https://open.fda.gov/

La première requête permet de rechercher un médicament en utilisant le nom de sa substance active, tandis que la seconde utilise son nom de marque. Étant donné que l'ontologie des médicaments est composée de classes contenant des noms de médicaments et d'entités chimiques, nous avons ainsi pu récupérer des médicaments en utilisant leur substance active principale. Par exemple, le terme *Collagène* concerne le champ ciblé que nous avons cherché à récupérer, qui est la substance active, et *Concerta* désigne le nom spécifique du médicament que nous recherchons.

4.3 SiMHOMer

Notre étude vise à fusionner des éléments des ontologies DOID, DRON et SYMP et à les aligner avec les données des événements indésirables de l'OpenFDA. Le résultat de ces alignements constitue une nouvelle ressource sémantique enrichie dans laquelle (i) chaque maladie est associée à un médicament potentiel et à un symptôme potentiel, et (ii) chaque médicament est lié à un ou plusieurs événements indésirables. Cette contribution s'inscrit dans le cadre du projet Predibioontlo (Predicting Clinical Diagnoses with Biomedical Ontologies and Language Models), dont l'objectif est d'utiliser le deep learning et les bases de connaissances biomédicales pour développer un outil visant à aider les médecins dans la prédiction des diagnostics et la proposition de médicaments pour leurs patients. Les phases listées dans [22] ont été adoptées pour le processus d'alignement.

La Figure 2 illustre l'approche globale permettant de générer les nouvelles relations entre les différentes ontologies. Nous commençons avec les trois ontologies distinctes que nous voulons fusionner, ainsi que la base de données OpenFDA contenant des informations sur les événements indésirables. La première étape (prétraitement) consiste à extraire les données textuelles des ontologies, y compris les définitions, les étiquettes et les synonymes. Dans la deuxième étape (fusion), ces données sont ensuite transformées en plongements (embeddings) en utilisant les modèles siamois. Ces embeddings permettent de calculer les similarités entre les différents éléments. En parallèle, un extracteur d'événements indésirables de médicaments est utilisé pour récupérer les événements indésirables associés aux médicaments en entrée dans DRON. Dans la dernière étape du processus, des correspondances sont établies entre

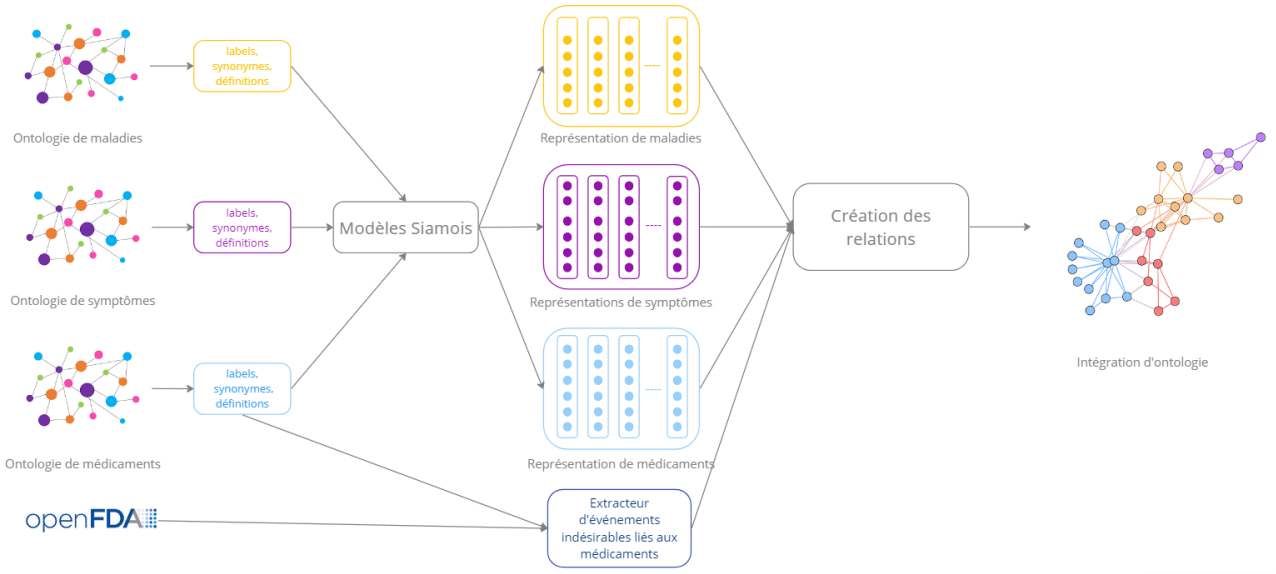


FIGURE 2 – Approche générale de la fusion des ontologies DRON, DOID, SYMP et alignement avec l'OpenFDA.

les relations suggérées par les modèles à travers les différentes ontologies et les données des événements indésirables de l'OpenFDA, créant ainsi une ontologie englobant ces diverses entités.

4.3.1 Prétraitement

Toutes les données textuelles ont été extraites des trois ontologies DOID, DRON et SYMP via la bibliothèque Owlready2 [5] [6]. Ces données sont liées :

- à la définition décrivant une maladie [7], les libellés et les synonymes ;
- aux métadonnées de ChEBI à partir desquelles l'ontologie DRON a été décrite. Ces métadonnées représentent des informations sur une maladie via une définition de propriété de données (métadonnées de ChEBI [8]), les libellés et les synonymes des médicaments ;
- à la définition décrivant un symptôme, son libellé et ses synonymes.

4.3.2 Fusion

Notre modèle est utilisé comme fonction de fusion, dans laquelle les bases de connaissances externes représentent les données sur lesquelles le modèle est entraîné : d'abord sur le corpus PubMed, puis sur la base de données clinique MIMIC III (une base de données contenant des dossiers patient électroniques).

Afin de trouver des relations entre maladies, symptômes et médicaments, nous avons procédé selon différentes manières. En effet, dans notre précédente étude [20], nous n'avions uniquement considéré que les noms des maladies de l'ontologie DOID ($rdfs$: $label$) et calculé les similarités entre ces éléments et les métadonnées de l'ontologie DRON (obo : $IAO_0000115$). Nous avons ensuite amélioré le processus en considérant deux approches différentes qui prennent en compte d'autres éléments constitutifs de DOID. En effet, le nom de la maladie n'étant pas suffisant, nous utilisons soit la concaténation des éléments de l'ontologie, soit nous ne considérons qu'un seul élément à la fois (celui représentant la similarité maximale).

Dans la concaténation de plusieurs éléments de l'ontologie DOID, ces éléments correspondent au nom de la maladie (rdf : $label$), à sa définition (obo : $IAO_0000115$) et à plusieurs noms de maladies synonymes ($oboInOwl$: $hasExactSynonym$). Nous appelons cette stratégie "multi-label". La concaténation est considérée comme une entrée pour notre modèle. Dans la seconde stratégie, nous ne considérons qu'un seul élément à la fois de DOID. Plus précisément, nous prenons en compte soit le nom de la maladie (rdf : $label$), soit la définition de la maladie (obo : $IAO_0000115$), soit un seul nom de maladie associé ($oboInOwl$: $hasExactSynonym$) dans chaque calcul de similarité. Ainsi, dans cette approche, pour chaque élément de DRON ou SYMP considéré par notre modèle, la correspondance est établie avec un élément de DOID, en choisissant le score de similarité maximum entre les métadonnées de DRON ou SYMP (obo : $IAO_0000115$), et l'une des métadonnées de DOID (rdf : $label$ ou $oboInOwl$: $hasExactSynonym$ ou obo : $IAO_0000115$).

4.3.3 Génération des Relations

Les alignements générés sont des correspondances entre un seul concept de DOID et un seul concept de DRON ou SYMP (alignement un à un). Une nouvelle relation est définie entre la maladie, le concept DRON et le concept SYMP. Cette nouvelle relation permet la génération d'une ontolo-

5. https://owlready2.readthedocs.io/en/v0.42/
6. https://github.com/arieme/OM_with_BioSTransformers
7. http://purl.obolibrary.org/obo/
8. http://purl.obolibrary.org/obo/$IAO_0000115$

gie plus complète (ontologie d'intégration) enrichie par les ontologies DRON, SYMP et DOID. Nous avons nommé ces nouvelles relations *has_drug* et *has_symptom*. Nous avons choisi un simple label qui décrit d'une manière très généraliste la relation. En raison des nuances et des questions très vaste qui existent dans le domaine de la médecine, par exemple : une médicament prescrit pour un certain type de patient qui a un certain âge ou sexe, etc.
Nous avons sélectionné la définition de la deuxième ontologie, SYMP, à des fins de comparaison, car elle contient un nombre significatif de synonymes nuls. Dans les cas où les définitions sont absentes, nous avons utilisé le libellé à la place.

4.3.4 Extraction des effets indésirables

Dans cette étape, nous avons extrait tous les effets indésirables liés à un médicament, soit en utilisant son nom de marque s'il est disponible, soit en utilisant sa substance active s'il n'existe pas dans la base de données. Ensuite, pour chaque maladie, nous avons créé la relation *has_side_effect*. Enfin, nous intégrons tout cela avec les relations *has_drug* et *has_symptom* dans un graphe que nous illustrons dans la figure 3.
La Figure 3 montre un exemple de relations générées entre les différentes ontologies. La maladie (en rouge) *Acquired angioedema* a comme symptôme (en marron) *anaphylaxis* et peut être traitée par le médicament (en vert) *icatibant* qui a plusieurs effets indésirables (en rose) comme *Laryngeal oedema* et *Stridor*.

5 Validation des Relations

Dans cette section, nous décrivons comment nous validons les alignements obtenus. À cette fin, nous nous appuyons sur l'utilisation de l'UMLS, le système de représentation des connaissances le plus utilisé dans le domaine biomédical. Nous utilisons également un grand modèle de langue.

5.1 UMLS

L'UMLS est une ressource exhaustive composée d'un Metathesaurus et d'un réseau sémantique (Semantic Network) développés par National Library of Medicine (NLM). Il sert de ressource unifiée pour les resources termino-ontologiques biomédicales, fournissant un moyen de relier divers vocabulaires et classifications biomédicales. Il comprend une vaste gamme de termes, concepts et relations provenant de sources diverses telles que la littérature médicale, les dossiers de santé électroniques, les terminologies cliniques et les ontologies.

5.1.1 Le Metathesaurus de l'UMLS

C'est le plus grand composant de l'UMLS. Il s'agit d'un vaste thésaurus biomédical organisé par concept, qui relie les noms similaires pour le même concept provenant de près de 200 vocabulaires différents. Le Metathesaurus identifie également les relations utiles entre les concepts et préserve les significations, les noms de concepts et les relations de chaque vocabulaire. Chaque terme dans le Metathesaurus est attribué un identifiant de concept unique (CUI), permettant l'interopérabilité et la liaison entre différents systèmes biomédicaux.
Nous avons identifié les relations issues du Metathesaurus les plus adaptées à notre cas d'étude, à savoir :

```
may_be_treated_by
treated_by
has_sign_or_symptom
sign_or_symptom_of
```

Parmi les relations que nous souhaitons valider, seuls certaines sont disponibles dans le Metathesaurus de l'UMLS. Par exemple, dans le cas de la relation *may_be_treated_by*, nous n'avons identifié que quelques relations correspondant aux maladies que nous recherchions. Les relations restantes concernant d'autres maladies ne sont pas incluses dans notre ontologie DOID ou qui existent dans notre ontologie mais pas dans UMLS. Cependant, ces 22 relations générées par notre approche sont incluses dans le Metathesarus, ce qui nous permet de les valider (Tableau 1).
En revanche, pour la relation *has_sign_or_symptom*, nous avons identifié plusieurs relations correspondant à celles que le modèle propose. Des exemples de ces relations trouvées dans le Metathesaurus sont présentés dans le Tableau 2. Il faut noter que le Metathesaurus ne fournit pas de relations spécifiques entre les maladies et les symptômes. Par exemple, pour *Irregular Heartbeat* (*Battement de cœur irrégulier*), il mentionne uniquement le symptôme *CIRC BLOOD*, qui est un terme trop générique. Les symptômes dans le Metathesaurus sont à un niveau élevé d'abstraction, même si nous sommes descendus au niveau maximum de l'arborescence au niveau des atomes, ce sont encore des termes généraux. Cependant, avec notre modèle, nous pouvons identifier le symptôme *Palpitation* qui a une relation avec *CIRC BLOOD* mais qui est plus spécifique.

5.1.2 Le Réseau Sémantique de l'UMLS

Le réseau sémantique (Semantique Network) se compose d'un ensemble de catégories, ou Types Sémantiques, qui fournissent une catégorisation cohérente de tous les concepts représentés dans le Metathesaurus de l'UMLS et d'un ensemble de relations sémantiques reliant les Types Sémantiques. Le réseau Sémantique contient 127 types sémantiques et 54 relations sémantiques. Pour valider nos relations, nous avons considéré les relations suivantes :

```
Pharmacologic Substance|treats|Sign or
    Symptom
Sign or Symptom|diagnoses|Pathologic
    Function
Virus|causes|Pathologic Function
Sign or Symptom|associated_with|Disease
     or Syndrome
Finding|associated_with|Disease or
    Syndrome
Finding|associated_with|Pathologic
    Function
```

Dans l'UMLS, les relations au sein du Metathesaurus sont idéalement alignées sur celles du Semantic Network. De

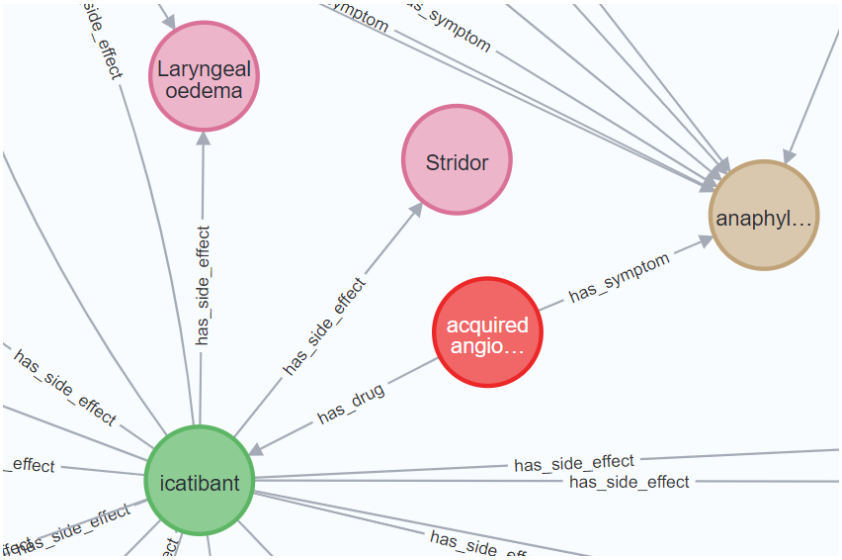


FIGURE 3 – Exemple de relations générées (has_side_effect ; has_drug ; has_symptom).

TABLE 1 – Exemples de la relation : *may_be_treated_by* proposées par notre modèle et retrouvées dans le Metathesaurus.

Maladie DOID	**CUI**	**Médicament DRON**	**CUI**
Obstructive sleep apnea	C0520679	Modafinil	C0066677
Enterocolitis, Pseudomembranous	C0014358	Fidaxomicin	C0065023
Pulmonary Fibrosis	C0034069	Pirfenidone	C0298067
Epilepsy	C0014544	Felbamate	C0060135
Ovarian neoplasm	C0919267	Niraparib	C2744440

nombreuses études ont été menées pour évaluer la pertinence de ces relations lorsqu'elles sont projetées vers le Metathesaurus [19, 29]. Dans notre étape de validation, nous supposons que des relations pourraient être déduites. En conséquence, le tableau 3 montre le nombre de relations trouvées dans le Réseau Sémantique qui correspondent à nos relations proposées, en fonction des relations sélectionnées. Une prochaine étape consistera à identifier toutes les autres relations potentielles pour une exploration plus approfondie.

5.2 Grands Modèles de Langage

Les progrès récents dans les grands modèles de langage (LLM) offrent de nouvelles opportunités pour le domaine biomédical. Ce sont des modèles puissants construits à l'aide de réseaux neuronaux contenant des milliards de paramètres. Ils sont entraînés sur de vastes volumes de texte généré par des humains [30] qui permettent ensuite de générer des textes, répondre à des questions, etc [9].

5.2.1 Le Modèle LlaMA

LlaMA [27] est un LLM publié par Meta en 2023. Plusieurs évaluations ont montré le potentiel de ces LLM dans le domaine biomédical [25]. Nous utilisons LlaMA 2 comme chatbot pour évaluer les relations que nous avons proposées. Bien que les LLM démontrent des capacités remarquables en zéro-shot, ils rencontrent des limitations lorsqu'ils sont confrontés à des tâches plus complexes. Pour y remédier, la sollicitation en quelques étapes peut faciliter l'apprentissage en contexte [3].

5.2.2 Les Prompts

Un prompt fait référence à l'entrée initiale fournie au modèle pour générer une sortie souhaitée. Il sert de repère ou d'instruction pour le modèle, le guidant pour produire du texte qui correspond à la tâche ou au contexte souhaité.

Diverses études ont examiné l'effet de l'ingénierie des prompts sur les performances des modèles de langage, ainsi que la variation des réponses générées en sortie pour une même tâche lors de l'utilisation de différents prompts. [18, 31] ont montré que la réponse générée par le modèle de langage présente une forte corrélation avec le prompt fourni en entrée.

Dans [9], les auteurs ont testé différents prompts pour évaluer les performances de ChatGPT dans sa réponse aux questions médicales et ont démontré que l'inclusion d'informations contextuelles supplémentaires a le potentiel d'influencer les réponses de ChatGPT.

Nous proposons d'utiliser différents prompts pour évaluer leur efficacité. Nous avons développé le prompt de manière itérative, en étudiant les résultats obtenus au fur et à mesure. Nous avons observé que le fait de fournir du contexte

TABLE 2 – Exemples de la relation : *has_sign_or_symptom* proposées par notre modèle et retrouvées dans le Metathesaurus.

Maladie DOID	**Symptôme SYMP**	**UMLS Disease**	**UMLS Symptom**
Obstructive sleep apnea	sleep apnea	Sleep apnea syndrome	Finding of sleep rest pattern
Esophageal varix	obsolete portal hepatitis	Varices	CIRC BLOOD
alcohol use disorder	concentration difficulty	Alcohol abuse	AOD use
spermatogenic failure 17	obsolete impotence	Infertility	Reproductive function
dilated cardiomyopathy 1X	muscle weakness	Muscle Weakness	Neuro-musculo-skeletal function
atrial fibrillation	palpitation	Irregular Heartbeat	CIRC BLOOD
Prinzmetal angina	tachycardia	Angina Pectoris	CIRC BLOOD
Perlman syndrome	renal involvement	Ascites	Digestion-hydration

TABLE 3 – Nombre de relations trouvées qui sont dans le Semantic Network.

Nom de relation	**Nombre**
Antibiotic | *treats* | Disease or Syndrome	191
Sign or Symptom | *associated_with* | Disease or Syndrome	907

Prompt 1 :

```
Please say if there is a symptom-disease association for me:
When confirming, please return clear answers like 'true' or 'false'. Please provide
    the result in JSON format, with the following structure:
{'symptom': 'symptom name',
'disease': 'disease name',
'answer': 'true or false'}.

Text:
the disease {disease} defined by:
{Definiton}
the symptom {symptom} defined by:
{Definiton}
```

FIGURE 4 – Contenu du Prompt 1.

au modèle lui permettait de générer une réponse pertinente pour notre scénario. Le contenu du prompt en relation avec nos types de données et de relations est présenté dans Figure 4 et 5 :

5.3 Résultats

Nous avons testé une liste de relations validées via l'utilisation du Metatesaurus ou du Semantic Network de l'UMLS pour vérifier si le LLM renvoit le même résultat ou pas. Certaines relations non disponibles dans l'UMLS sans source ont également été considérées comme vraies.

La Table 4 montre que, parmi toutes les relations que nous lui avons fournies, le LLM a validé avec succès les relations présentes dans le Metathesaurus et le Semantic Network de l'UMLS, ainsi que des relations supplémentaires absentes de l'UMLS. Il n'a remis en question aucune des relations proposées.

6 Conclusion

Dans cette étude, nous avons présenté une approche novatrice visant à consolider plusieurs ontologies pertinentes pour le domaine de la santé dans un cadre d'intégration d'ontologies unifiée. En exploitant nos modèles entraînés sur des données biomédicales, nous avons proposé des relations potentielles entre des concepts provenant d'ontologies disparates.

Pour valider les relations proposées, nous les avons croisées avec celles qui existent dans le Metathesaurus de l'UMLS et son Semantic Network. Cependant, nous n'avons pas pu valider toutes nos relations avec cette ressource, qui s'est avérée insuffisante. Nous avons également utilisé les capacités des grands modèles de langue (LLM) pour compléter ce processus de validation. Cette validation initiale confirme l'exactitude de nos relations proposées par le LLM.

Nos premiers résultats sont très prometteurs, même si nous n'avons qu'une partie des relations proposées qui

Prompt 2 :

```
Please say if there is a drug-disease association for me:
When confirming, please return clear answers like 'true' or 'false'. Please provide
    the result in JSON format, with the following structure:
{'drug': 'drug name',
'disease': 'disease name',
'answer': 'true or false'}.

Text:
the disease {disease} defined by:
{Definiton}
the symptom {drug} defined by:
{Definiton}
```

FIGURE 5 – Contenu du Prompt 2.

TABLE 4 – Relations testées avec le modèle LlaMA.

Maladie DOID	Symptôme ou Médicament	Relation	Source	Réponse LlaMA
angiosarcoma	spontaneous ecchymoses	_	_	true
Obstructive sleep apnea	sleep apnea	has_sign_or_symptom	Metathesaurus	true
obstructive sleep apnea	modafinil	may_be_treated_by	Metathesaurus	true
amodiaquine allergy	Exanthema	associated_with	Semantic Network	true
Zollinger-Ellison syndrome	Vomiting	associated_with	Semantic Network	true

est validée. Nos travaux futurs se concentreront sur un processus de validation plus approfondi, potentiellement en utilisant des ressources externes plus complètes telles que celles incluses dans le service de recherche d'ontologies (OLS) de l'Institut de bioinformatique européen de l'EMBL (`https://www.ebi.ac.uk/ols4`), ou encore par la sollicitation d'experts du domaine. Les nouvelles relations dérivées du cadre SiMHOMer offrent des avantages potentiels significatifs pour les spécialistes de la santé, promettant une intégration et une accessibilité des connaissances enrichies du domaine. Nous envisageons de comparer nos modèles, qui ont déjà démontré leur efficacité sur d'autres tâches, à d'autres modèles spécifiques à cette tâche (dans OAEI par exemple).

Références

[1] Ismail Akbari, Mohammad Fathian, and Kambiz Badie. An improved mlma+ and its application in ontology matching. In *2009 Innovative Technologies in Intelligent Systems and Industrial Applications*, pages 56–60. IEEE, 2009.

[2] Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. Publicly available clinical BERT embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA, June 2019. Association for Computational Linguistics.

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33 :1877–1901, 2020.

[4] Ursin Brunner and Kurt Stockinger. Entity matching with transformer architectures-a step forward in data integration. In *23rd International Conference on Extending Database Technology, Copenhagen, 30 March-2 April 2020*, pages 463–473. OpenProceedings, 2020.

[5] Xiang Chen, Ningyu Zhang, Lei Li, Shumin Deng, Chuanqi Tan, Changliang Xu, Fei Huang, Luo Si, and Huajun Chen. Hybrid transformer with multi-level fusion for multimodal knowledge graph completion. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 904–915, 2022.

[6] William W Cohen, Pradeep Ravikumar, Stephen E Fienberg, et al. A comparison of string distance metrics for name-matching tasks. In *IIWeb*, volume 3, pages 73–78, 2003.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT : Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186, 2019.

[8] Jérôme Euzenat, Pavel Shvaiko, et al. *Ontology matching*, volume 18. Springer, 2007.

[9] Zhenxiang Gao, Lingyao Li, Siyuan Ma, Qinyong Wang, Libby Hemphill, and Rong Xu. Examining the potential of chatgpt on biomedical information retrieval : Fact-checking drug-disease associations. *Annals of Biomedical Engineering*, pages 1–9, 2023.

[10] Josh Hanna, Eric Joseph, Mathias Brochhausen, and William Hogan. Building a drug ontology based on rxnorm and other sources. *Journal of biomedical semantics*, 4 :44, 12 2013.

[11] Wei He, Xiaoping Yang, and Dupei Huang. A hybrid approach for measuring semantic similarity between ontologies based on wordnet. In *International Conference on Knowledge Science, Engineering and Management*, pages 68–78. Springer, 2011.

[12] Sven Hertling, Jan Portisch, and Heiko Paulheim. Melt-matching evaluation toolkit. In *International conference on semantic systems*, pages 231–245. Springer International Publishing Cham, 2019.

[13] Sven Hertling, Jan Portisch, and Heiko Paulheim. Matching with transformers in melt. *arXiv preprint arXiv :2109.07401*, 2021.

[14] Taha A Kass-Hout, Zhiheng Xu, Matthew Mohebbi, Hans Nelsen, Adam Baker, Jonathan Levine, Elaine Johanson, and Roselie A Bright. Openfda : an innovative platform providing access to a wealth of fda's publicly available data. *Journal of the American Medical Informatics Association*, 23(3) :596–600, 2016.

[15] Yuliang Li, Jinfeng Li, Yoshihiko Suhara, Jin Wang, Wataru Hirota, and Wang-Chiew Tan. Deep entity matching : Challenges and opportunities. *Journal of Data and Information Quality (JDIQ)*, 13(1) :1–17, 2021.

[16] Hao Liu, Yehoshua Perl, and James Geller. Concept placement using bert trained by transforming and summarizing biomedical ontology structure. *Journal of Biomedical Informatics*, 112 :103607, 2020.

[17] Schriml Lynn, Cesar Arze, Suvarna Nadendla, Yu-Wei Wayne Chang, Mark Mazaitis, Victor Felix, Gang Feng, and Warren Kibbe. Disease ontology : A backbone for disease semantic integration. *Nucleic acids research*, 40 :D940–6, 11 2011.

[18] Ggaliwango Marvin, Nakayiza Hellen, Daudi Jjingo, and Joyce Nakatumba-Nabende. Prompt engineering in large language models. In *International Conference on Data Intelligence and Cognitive Informatics*, pages 387–402. Springer, 2023.

[19] Alexa T McCray and Olivier Bodenreider. A conceptual framework for the biomedical domain. In *The semantics of relationships : an interdisciplinary perspective*, pages 181–198. Springer, 2002.

[20] Safaa Menad, Wissame Laddada, Saïd Abdeddaïm, and Lina F Soualmia. Biostransformers for biomedical ontologies alignment. In *Proceedings of the 15th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, IC3K 2023, Volume 2 : KEOD*, pages 73–84, 2023.

[21] Safaa Menad, Wissame Laddada, Saïd Abdeddaïm, and Lina F Soualmia. New siamese neural networks for text classification and ontologies alignment. In *International Conference on Complex Computational Ecosystems*, pages 16–29. Springer, 2023.

[22] Inès Osman, Sadok Ben Yahia, and Gayo Diallo. Ontology integration : Approaches and challenging issues. *Information Fusion*, 71 :38–63, Jul 2021.

[23] Jan Portisch, Michael Hladik, and Heiko Paulheim. Background knowledge in ontology matching : A survey. *Semantic Web*, pages 1–55, 09 2022.

[24] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8) :9, 2019.

[25] Arya Rao, Michael Pang, John Kim, Meghana Kamineni, Winston Lie, Anoop K Prasad, Adam Landman, Keith Dreyer, and Marc D Succi. Assessing the utility of chatgpt throughout the entire clinical workflow : development and usability study. *Journal of Medical Internet Research*, 25 :e48659, 2023.

[26] Nils Reimers and Iryna Gurevych. Sentence-BERT : Sentence embeddings using Siamese BERT-networks. In *Proceedings of (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China, November 2019. Association for Computational Linguistics.

[27] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama : Open and efficient foundation language models, 2023.

[28] Tao Tu, Eric Loreaux, Emma Chesley, Adam D Lelkes, Paul Gamble, Mathias Bellaiche, Martin Seneviratne, and Ming-Jun Chen. Automated loinc standardization using pre-trained large language models. In *Machine Learning for Health*, pages 343–355. PMLR, 2022.

[29] Li Zhang, Michael Halper, Yehoshua Perl, James Geller, and James J Cimino. Relationship structures and semantic type assignments of the umls enriched semantic network. *Journal of the American Medical Informatics Association*, 12(6) :657–666, 2005.

[30] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv :2303.18223*, 2023.

[31] Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. Large language models are human-level prompt engineers, 2023.

Intégration de SOSA/SSN, SAREF, et TD dans l'ontologie CoSWoT

M. Lefrançois[1], C. Roussey[2], F.-Z. Hannou[3], V. Charpenay[1]

[1] Mines Saint-Etienne, Univ Clermont Auvergne, INP Clermont Auvergne, CNRS, UMR 6158 LIMOS, Saint-Étienne, France
[2] MISTEA, INRAE Centre Occitanie, Montpellier, France
[3] EDF R&D, Palaiseau, France

maxime.lefrancois@emse.fr, catherine.roussey@inrae.fr, fatma-zohra.hannou@edf.fr, victor.charpenay@emse.fr

Résumé

Cet article décrit l'ontologie CoSWoT construite au cours du projet « Constrained Semantic Web of Things » à l'aide de la méthodologie ACIMOV [13]. Cette ontologie réconcilie trois ontologies de référence du Web des Objets : « Semantic Sensor Network » (SSN/SOSA) [11] et « Thing Description » (TD) [3] du W3C, « Smart Applications REFerence » (SAREF) de ETSI [9], à travers la proposition d'un nouveau patron de conception architectural « Kinds of X and X of Interest ». Son objectif est de permettre à des dispositifs distribués sur le Web d'échanger des données sans ambiguïté. Nous illustrons les éléments de cette ontologie par un cas d'usage en bâtiment intelligent.

Mots-clés

Web des Objets Sémantique, ontologie modulaire, description de service, capteur, actionneur, raisonnement distribué, patron de conception d'ontologies.

Abstract

This article describes the CoSWoT ontology built during the "Constrained Semantic Web of Things" project using the ACIMOV [13] methodology. This ontology reconciles three reference ontologies from the Web of Things world : "Semantic Sensor Network" (SSN/SOSA) [11] and "Thing Description" (TD) [3] of the W3C, "Smart Applications REFerence" (SAREF) from ETSI [9], through the definition of a new ontology design pattern "Kinds of X and X of Interest". Coswot's goal is to enable devices distributed on the Web to exchange data unambiguously. We will illustrate the elements of this ontology with a smart building use case.

Keywords

Semantic Web of Objects, ontology, service description, sensor, actuator, distributed reasoning, design pattern.

1 Introduction

Le projet Constrained Semantic Web of Things (CoSWoT) [1] s'attaque aux verrous liés à l'utilisation du Web sémantique dans les objets contraints de l'Internet des Objets, en particulier ceux liés à l'interopérabilité sémantique et au raisonnement distribué.

Le projet CoSWoT a pour objectif de proposer une architecture logicielle distribuée et embarquée sur des dispositifs contraints du Web des Objets (WoT). Cette architecture a deux caractéristiques principales : (1) elle utilisera des modèles de connaissances à base de graphes pour déclarer la logique applicative des dispositifs et la sémantique des messages échangés ; (2) les dispositifs auront des capacités de raisonnement afin de répartir les tâches de traitement entre eux. Le développement d'applications WoT sera simplifié : notre plateforme permettra le développement et l'exécution d'applications intelligentes et décentralisées du WoT malgré l'hétérogénéité des dispositifs connectés. La plateforme proposée sera testée sur plusieurs cas d'utilisation dans le bâtiment intelligent et en agriculture numérique.

La contribution principale présentée dans cet article est l'ontologie CoSWoT, qui comprend les connaissances communes à tous les composants de la plateforme CoSWoT et les connaissances spécifiques à certains domaines d'application. Le développement de l'ontologie CoSWoT vise à satisfaire les objectifs suivants :

- **O1** L'ontologie doit être alignée à des ontologies de référence du WoT et doit réutiliser des ontologies de référence des domaines visés ;
- **O2** L'ontologie doit être modulaire, avec des modules qui couvrent les connaissances communes à tous les composants de la plateforme WoT ;
- **O3** L'ontologie doit avoir une structure homogène et prédictible, de sorte que des concepts similaires définis dans des domaines différents soient décrits de la même manière ;
- **O4** Des sous-ensembles de l'ontologie CoSWoT, ou *vues*, devraient pouvoir être embarquées dans des dispositifs avec des contraintes de ressources, pour manipuler, raisonner et échanger des données de capteurs pour une application spécifique.

Nous illustrons le développement de cette ontologie pour une application du domaine du bâtiment intelligent, qui porte sur le bâtiment Espace Fauriel (EF) de Mines Saint-Étienne [8]. Notre exemple implique plusieurs dispositifs

1. https://coswot.gitlab.io/

déployés dans le bureau 429 :

- un radiateur avec un thermomètre, piloté à distance ;
- un noeud près de la fenêtre avec un thermomètre, un capteur qui évalue le taux de dioxyde de carbone de l'air (capteur CO2), un bras actionnable permettant d'ouvrir la fenêtre et de donner son statut, des capacités de communications, de calcul et de raisonnement ;
- un noeud près du tableau avec un thermomètre, un capteur de CO2 et des capacités de communication ;
- un noeud près de la porte avec un thermomètre, un capteur de CO2 et des capacités de communication.

Dans notre exemple, les capteurs de CO2 mesurent toutes les 5 secondes, les noeuds envoient leurs mesures au noeud de la fenêtre pour calculer la mesure maximale de concentration de CO2. A partir de cette mesure, ce noeud déduira la qualité de l'air du bureau. L'objectif est d'automatiser l'ouverture de la fenêtre en fonction de la qualité de l'air.
Le reste de l'article est organisé ainsi : la section 2 donne un aperçu des évolutions récentes des ontologies de référence du domaine du WoT. La section 3 propose un patron de conception architectural qui permet de mitiger certains problèmes d'hétérogénéité identifiés dans ces ontologies. La section 4 présente la méthode de développement de l'ontologie CoSWoT pour satisfaire les objectifs **O1** à **O4**. La section 5 donne un aperçu de l'ontologie CoSWoT et présente quelques modules illustrés sur notre exemple. La section 6 conclue notre article.

2 Évolution des ontologies du WoT

Avec l'émergence du Semantic Web of Things (SWoT), les standards incluent des ontologies, de façon à promouvoir l'utilisation de quelques ontologies de référence, plutôt que d'observer un morcellement et la création de nouvelles ontologies avec chaque nouveau projet. Les organismes de standardisation qui contribuent aux ontologies du SWoT sont notamment : (1) Le World Wide Web Consortium (W3C), organisme international de standardisation du Web et du Web sémantique. (2) L'European Telecommunication Standards Institute (ETSI), organisme européen de standardisation pour les télécommunications. (3) oneM2M, consortium international rassemblant des organismes de standardisation (dont l'ETSI), des organismes de recherche et des industriels autour d'un standard pour l'Internet of Things (IoT). Nous avons présenté à IC 2019 un positionnement sur le Web Sémantique des Objets, dans lequel nous discutions des évolutions récentes des ontologies standardisées pour le WoT [19, Section 2]. Nous reprenons cette section ici et présentons une vision actualisée de leur évolution.

2.1 OGC&W3C SOSA/SSN

Le groupe d'incubation *Semantic Sensor Network* du W3C a publié en 2011 un rapport analysant les différents modèles conceptuels existants pour décrire les capteurs et leurs observations, ainsi qu'une proposition d'ontologie SSNX [18]. L'ontologie SSNX a été massivement réutilisée par d'autres ontologies et jeux de données. Parallèlement, l'Open Geospatial Consortium (OGC) publiait le modèle UML Observations and Measurements (O&M) V2.0 [4]. En 2017, le groupe de travail Spatial Data on the Web Working Group (SDW-WG) commun aux organismes de standardisation OGC et W3C a publié une mise à jour de cette ontologie en 2017, nommée **SOSA/SSN** [14, 12]. Celle-ci spécifie la sémantique des capteurs et actionneurs, entre autres. Elle permet de décrire notamment les capteurs, les propriétés des entités d'intérêt qu'ils observent, les observations qu'ils font et le résultat de ces observations. De manière analogue, elle permet de décrire les actionneurs, les propriétés des entités d'intérêt sur lesquelles ils peuvent agir, et les actionnements qu'ils font et le résultat de ces actionnements. SSN se compose d'un noyau léger appelé SOSA (Sensor, Observation, Sample, and Actuator), d'un module d'extension plus expressif SSN, d'un module SSN-Systems séparé pour les capacités du système et d'un ensemble de modules d'alignement avec d'autres ontologies. Le travail sur SOSA/SSN s'est poursuivi du côté du W3C avec des propositions d'extension pour les collections d'observation, et à l'OGC avec la publication récente d'une nouvelle version V3.0 du standards O&M (maintenant : *Observations, measurements and samples*). Dans la foulée, le groupe SDW-WG a planifié le développement d'une nouvelle version de l'ontologie SOSA/SSN pour 2024 [2].

2.2 W3C Thing Description

Le W3C comprend un autre groupe de travail qui contribue directement au SWoT : le W3C Web of Things working group. Un *Thing* est défini dans [17] comme l'abstraction d'une entité (virtuelle ou physique) pouvant être manipulée par une application IoT, *e.g.* un objet, un service, ou une entité logique telle qu'une pièce ou un bâtiment. L'ontologie **Thing Description** [15] décrit les *affordances d'interaction* des objets, c'est à dire les fonctionnalités qu'on peut invoquer pour manipuler son état, ou déclencher un processus. TD peut être utilisé conjointement avec l'ontologie des contrôles hypermédias HCTL [1] et le vocabulaire RDF pour JSON Schema [2] pour décrire précisément comment formuler une requête (HTTP, CoAP, MQTT, ...) pour interagir avec l'objet [16]. Un *Servient* –contraction de *Server* et de *Client*– est une suite logicielle qui expose ou consomme des Things.

2.3 SAREF

Le développement de l'ontologie Smart Applications Reference (**SAREF**) [3] [5, 9], a débuté en 2014/2015 avec une étude demandée par la Commission européenne pour comparer et aligner les modèles de connaissances en lien avec l'IoT, afin de limiter la fragmentation de ce marché. SAREF consiste aujourd'hui en un module noyau, et une dizaine d'extensions pour des domaines d'application comme l'énergie, les bâtiments, ou l'agriculture. Initialement centrée sur le concept de *Device* en tant qu'objet physique remplissant une ou plusieurs fonctions, l'ontologie SAREF a été progressivement amélioré au fil des versions. La ver-

2. https://github.com/w3c/sdw-sosa-ssn/
3. https://saref.etsi.org/

sion V3.2.1 de SAREF Core [7] a été publiée en janvier 2024, et démontre une volonté forte de convergence vers SOSA/SSN. Par exemple, la classe *Measurement* est déprécié en faveur de la classe *Observation*, reprise de SOSA/SSN. Des choix de modélisation de SAREF sont suggérés au groupe de travaille SDW-WG, et certains ont déjà été intégrés comme la classe *ProcedureExecution* qui généralise les observations et actuations, et la classe *PropertyOfInterest* qui décrit les qualités observables intrinsèquement liées à une entité d'intérêt. De plus, SAREF définit un ensemble de patrons d'ontologie de référence [6] qui contraignent son usage. Ces patrons indiquent quelles sont les classes et propriétés qui peuvent ou ne peuvent peut être spécialisées, et dans quel contexte. L'année 2024 devrait voir la publication d'une nouvelle version majeure (V4.1.1) de SAREF Core où les concepts dépréciés dans V3.2.1 seront supprimés, et une nouvelle version majeure (V2.1.1) de chacune des extensions de domaines conforme à la nouvelle version de SAREF Core et aux différents patrons d'ontologie de référence [4].

3 Positionnement pour CoSWoT

3.1 Convergence et influence

L'évolution parallèle et actuelle de SOSA/SSN, TD, et SAREF avec des gouvernances distinctes pose des défi concernant l'objectif **O1** pour l'ontologie CoSWoT. Nous contribuons à chacune de ces ontologies, mais il est difficile d'anticiper leur trajectoire d'évolution, du fait qu'elles résultent de décisions plus ou moins consensuelles dans leurs communautés respectives. On note néanmoins un désir de convergence, et des questions de modélisation similaires (voir [10, Section 7]). Par exemple, la question de savoir si une instance de la classe *Property* doit être générique (ex. température ambiante) ou spécifique à une entité d'intérêt (ex. la température du bureau 429) a été longtemps débattue. Il existe des usages pour les deux façons de modéliser [5]. La décision a été prise dans SAREF de restreindre l'emploi de *Property* seulement pour les propriété génériques, et d'introduire *PropertyOfInterest* pour décrire les propriétés spécifiques à une entité d'intérêt. SOSA/SSN a adopté ce choix également. De même, certains utilisateurs décrivent avec *sensor* des types de capteurs (ex. DHT22), alors que d'autres décrivent des instances (ex. le capteur DHT22 dans une pièce). SAREF introduit *FeatureKind* pour décrire des archétypes d'entités d'intérêt, mais pas la sous-classe *DeviceKind* ou *SensorKind*. Le nouveau choix de modélisation est discuté dans SOSA/SSN [6]. Ainsi, parfois le spécifique est défini par le suffixe « OfInterest », parfois le générique est défini par le suffixe « Kind », rendant difficile la compréhension de ces ontologies par un nouvel utilisateur. De plus, la compatibilité arrière est systématiquement rompue avec les usagers minoritaires d'un choix de modélisation. Enfin, il n'est pas encore clair quand et comment ces classes doivent être spécialisées. Doit-on privilégier la définition d'une instance de « Kind », ou d'une sous-classe de « OfInterest » ? Ces entités doivent-elles être définies par paire ? Doit-on utiliser le *punning* systématiquement ? [7] Est-ce pertinent de définir des sous-classes de la classe « Kind » ? Dans l'ontologie CoSWoT, nous proposons un patron de conception architectural unifié « *Kinds of X and X of Interest* », qui mitige ces problèmes et apporte des réponses à ces questions.

3.2 Le patron de conception architectural « *Kinds of X and X of Interest* »

L'ontologie CoSWoT prend comme référence les ontologies SOSA/SSN, SAREF et TD. Les classes importantes de SOSA/SSN et SAREF (FeatureOfInterest, Device, Sensor, Property, State, Observation, ...) sont redéfinies dans l'espace de nom de CoSWoT `https://w3id.org/coswot/`. Pour satisfaire l'objectif **O3** nous proposons une modélisation légèrement différente, en suivant systématiquement un patron de conception que l'on nomme « *Kinds of X and X of Interest* », illustré dans la figure 1.

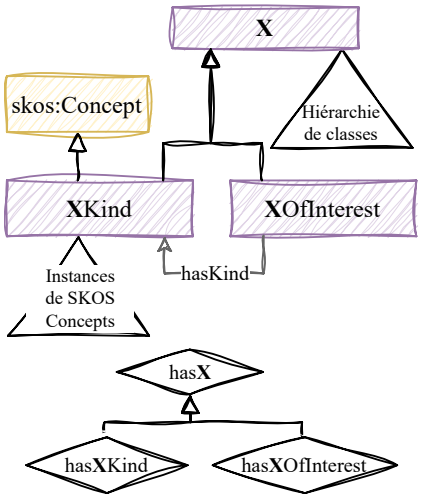


FIGURE 1 – Patron de conception architectural « *Kinds of X and X of Interest* », et l'extension « *has X* »

La classe **X** (ex. *Feature*, *Property*, *Device*) représente la classe des entités dont le type est indéterminé (ni générique, ni spécifique). Elle peut être spécialisée dans une hiérarchie de classe. La classe **X** est équivalente à l'union disjointe de **X**Kind et **X**OfInterest, qui représentent la classe des archétypes de **X**, et des **X** spécifiques, respectivement. **X**Kind est une sous-classe de *skos:Concept*, et les instances qui peuplent cette classe sont organisées en un modèle SKOS à l'aide des propriétés *skos:broader* et *skos:narrower* par exemple. Des restrictions locales sur **X**Kind forcent les concepts plus spécifiques et plus génériques à également être de type **X**Kind.
La propriété hasKind lie une entités spécifique à son type. Une restriction locale sur **X**OfInterest force tout objet de la

4. `https://portal.etsi.org/tb.aspx?tbid=726`
5. Voir sdw-sosa-ssn#106
6. Voir sdw-sosa-ssn#107, sdw-sosa-ssn#209
7. *punning* en OWL 2 fait référence à l'utilisation d'une même IRI pour identifier une classe, une propriété, et/ou un individu, sans que ces entités n'aient formellement de lien.

propriété hasKind à être de type **X**Kind. De plus, hasKind est sous-propriété de la chaîne hasKind o skos:broader, de sorte qu'une instance de **X**OfInterest hérite des **X**Kind plus génériques. La propriété hasKind étant « non-simple » [8] de par cet axiome, elle ne doit pas faire l'objet de restrictions de cardinalité. On peut néanmoins ajouter une restriction locale existentielle **X**OfInterest $\sqsubseteq$ $\exists$hasKind.**X**Kind, et une restriction universelle **X**Kind $\sqsubseteq$ $\forall$hasKind.$\bot$. Cette dernière restriction interdit l'utilisation de hasKind sur les instance de type **X**Kind.

Lorsqu'une sous-classe de **X** est définie, il est possible, mais pas obligatoire, d'utiliser le même patron de conception architectural. Par exemple, nous aurons *Sensor*, *SensorKind*, et *SensorOfInterest*, qui seront respectivement des sous-classes de *Device*, *DeviceKind*, et *DeviceOfInterest*.

Finalement, il est commun de devoir modéliser qu'une instance du patron pour **Y** fait référence à une instance du patron pour **X**. Un **X**OfInterest sera spécifique à un unique **Y**OfInterest. Par exemple, une propriété d'intérêt est spécifique à une entité d'intérêt. Le patron propose donc une extension « *has X* » qui définit six propriétés has**X**, is**X**Of, has**X**Kind, is**X**KindOf, has**X**OfInterest, et is**X**OfInterestOf, cette dernière étant fonctionnelle. Trois axiomes de type subPropertyChainOf permettent d'inférer les types :

has**X**Kind $\sqsubseteq$ skos:broader o has**X**Kind

has**X**Kind $\sqsubseteq$ hasKind o has**X**Kind

has**X**Kind $\sqsubseteq$ has**X**OfInterest o hasKind

Même si has**X**Kind est non-simple, la classe **X**OfInterest peut recevoir une restriction local de cardinalité =1 sur is**X**OfInterestOf sans que l'ontologie ne viole les contraintes du profile OWL2 DL.

Ce patron facilite la séparation des préoccupations : les développeurs d'extensions de CoSWoT pourront privilégier la définition de sous-classes de **X** ; les développeurs de taxonomies, thésaurus, et catalogues en ligne pourront privilégier l'instanciation des classes **X**Kind ; les développeurs d'applications pourront privilégier l'instanciation des classes **X**OfInterest.

4 Méthode de développement

Le développement de l'ontologie CoSWoT est réalisé en application de la méthodologie d'ingénierie d'ontologies ACIMOV (Agile Continuous Integration for Modular Ontologies and Vocabularies) [13]. ACIMOV offre aux équipes de développement d'ontologie un cadre agile, et un ensemble de ressources permettant d'accompagner l'entièreté du cycle de vie de l'ontologie, telles que des outils de validation, ou de développement/intégration continus. ACIMOV offre un support à (1) la réutilisation des ontologies de référence, (2) la modularité, (3) la collaboration.

Réutilisation des ontologies de référence. Dans CoSWoT, les ontologies SAREF, SOSA/SSN et TD ont été identifiées comme répondant aux besoins exprimés dans les cas d'usages. Elles représentent des références dans le domaine du WoT, ayant un maintien régulier, une documentation riche, et bénéficiant d'une large adoption de la communauté du domaine (besoin **O1**).

Modularité. CoSWoT est conçue comme une ontologie modulaire pour permettre une gestion efficace de la complexité du domaine du WoT, et des domaines d'applications sous-jacents. Cette modularité est d'abord horizontale : à l'image des ontologies de référence réutilisées, des modules (appelés modelets dans ACIMOV) sont créés pour capturer des représentations de connaissances complémentaires, chacun conçu autour d'un concept clé, et défini pour répondre à des questions de compétences issues des cas d'application CoSWoT (besoin **O2**). D'autres part, une hiérarchie des modelets est défini ; des modelets core sont créés pour consolider les connaissances du domaine du WoT génériques/communes à plusieurs domaines d'applications, et sont par la suite spécialisés selon les spécificités des applications (besoin **O3**). Cette modularité permet aussi de faciliter la réutilisation de certaines parties de l'ontologie, identifiées comme pertinentes pour un cas d'usage particulier à travers la création de vues (besoin **O4**).

Les développeurs d'ontologie constituent un « *backlog* » de modelet, qu'ils prennent en charge de manière parallèle afin d'optimiser le processus de développement.

Collaboration. comme introduit plus tôt, le périmètre du projet CoSWoT s'étend à deux domaines d'applications (agriculture numérique et bâtiment intelligent). Les experts de ces domaines ainsi que les « *product owners* » en charge de développement des cas d'usages sont impliqués dans le processus de développement de l'ontologie CoSWoT. En application de ACIMOV, la stratégie d'acquisition des connaissances démarre via la collecte des spécifications des cas d'usage par les experts de domaines, et se poursuit tout au long du cycle de vie de l'ontologie, en affinant les besoins grâce à des réunion récurrentes permettant de valider les modules ontologiques et d'assurer une intégration régulière des feedbacks des différentes parties prenantes de CoSWoT. La collaboration concerne aussi l'équipe de développement d'ontologies qui organise des sessions de revue régulières pour valider les modelets développés.

5 Ontologie CoSWoT Core

5.1 Aperçu

Le noyau de l'ontologie CoSWoT est compatible OWL2 DL et définit 61 classes, 69 propriétés objet, et 6 propriétés de données. Il contient 14 modelets, 8 instances du patron « *Kinds of X and X of Interest* » dont 4 fois avec l'extension « *has X* ». La figure 2 illustre les relations d'import entre ces modelets. Chaque modelet est accessible à son URL, et la fusion de tous ces modelets est accessible avec négociation de contenu à l'URL `https://w3id.org/coswot/core/`. Chaque terme est défini dans l'espace de nom coswot: (enregistré sur le service prefix.cc), et un HTTP GET à son URL redirige vers le modelets qui le définit. Dans le reste de cette section, nous présentons quelques-uns de ces modelets.

8. Les propriétés non-simple font l'objet de restrictions globales pour satisfaire le profile OWL2 DL

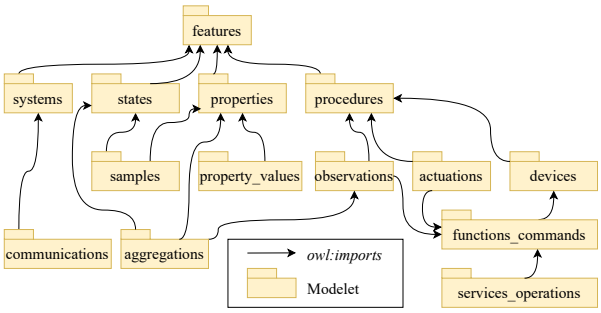


FIGURE 2 – Aperçu des dépendances entre les modelets de l'ontologie CoSWoT

5.2 Features

Le modelet **coswot:core/features** décrit les entités représentant toute entité du monde réel ayant une propriété ou un état qui sera observé ou contrôlé. Une instance de *coswot:Feature* est soit une instance de *coswot:FeatureOfInterest*, c'est à dire une entité spécifique du monde réel, soit une instance de *coswot:FeatureKind*, c'est à dire un archétype d'entité.

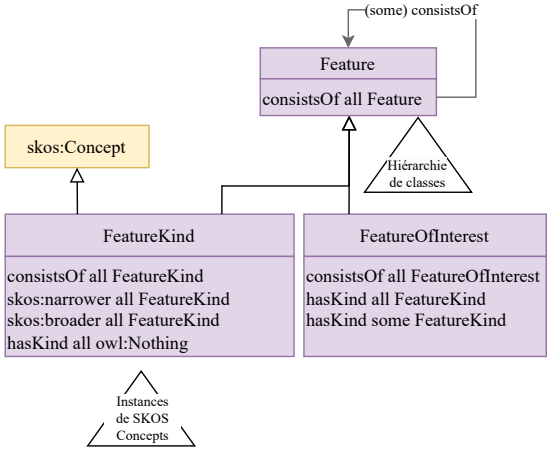


FIGURE 3 – Diagramme Chowlk du modelet Features. NOTE : les diagrammes peuvent être téléchargés et visualisés avec l'application draw.io : `https://drive.usercontent.google.com/uc?id=1YLwbnZobNIHKCXJfScjohD-5jGThZfpf&export=download`

Ainsi dans notre exemple, l'entité objet de l'étude est le bureau 429 du bâtiment EF. Comme le montre la figure 4, la représentation du bureau est une instance de la classe *coswot:FeatureOfInterest*, car cette ressource identifie un objet unique du monde réel. Elle est aussi instance de la classe *coswot:OfficeRoom* sous classe *coswot:Feature*. La ressource indique que ce bureau peut aussi être classé dans la catégorie bureau de 20 m2.

5.3 Devices

Le modelet **coswot:core/devices** décrit les appareils ou dispositifs, dont les capteurs et les actionneurs. Comme

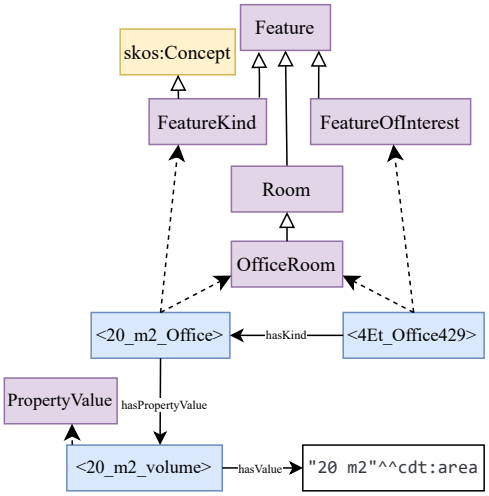


FIGURE 4 – Diagramme Chowlk du bureau 429

les appareils sont des entités du monde réel, il spécialise le modelet Features. Une représentation d'un appareil est une instance de *coswot:Device*, sous classe de *coswot:Feature*. Cette représentation est soit une instance de *coswot:DeviceOfInterest*, c'est à dire un appareil identifié du monde réel, soit une instance de *coswot:DeviceKind*, c'est à dire un archétype d'appareils.

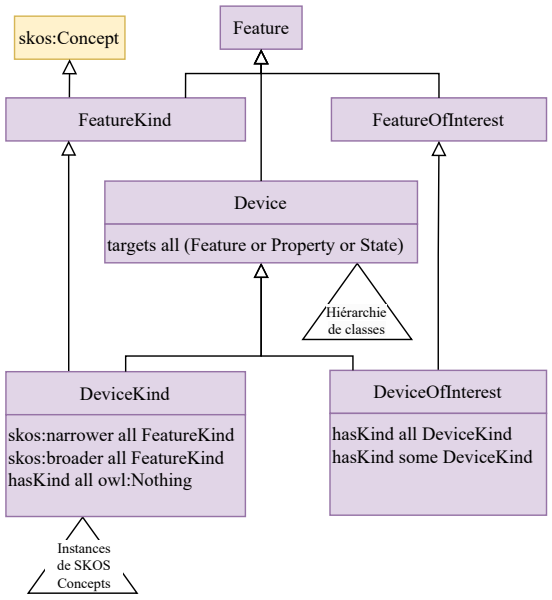


FIGURE 5 – Diagramme Chowlk du modelet Devices

Ainsi dans notre exemple, le bureau 429 est équipé de plusieurs appareils : le radiateur, la fenêtre à ouverture automatique, les capteurs, etc. La figure 6 présente le servient de la fenêtre qui contient deux capteurs. Ainsi, les représentations du servient, des deux capteurs sont des instances de *coswot:DeviceOfInterest* car ces ressources identifient un objet unique du monde réel. Elles sont aussi instances de sous classes *coswot:Device* : *coswot:Servient*,

coswot:Thermometer, *coswot:GazSensor*. La ressource représentant le thermomètre indique que ce capteur à une précision de 0.1%.

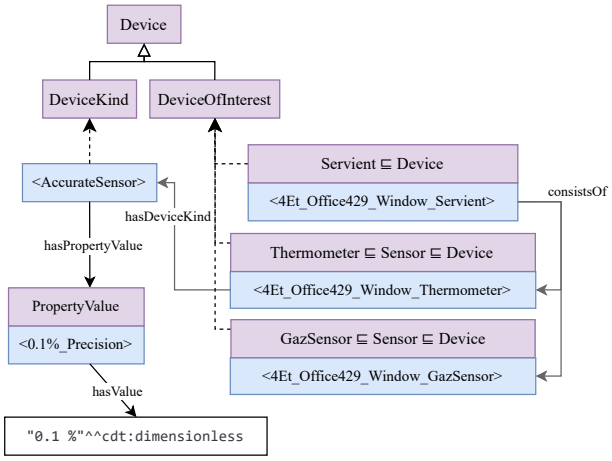


FIGURE 6 – Diagramme Chowlk du servient de la fenêtre

5.4 Properties

Le modelet **coswot:core/properties** décrit les propriétés des entités. Les propriétés font référence aux qualités identifiables des entités que des dispositifs peuvent observer. Une instance de *coswot:Property* est soit une instance de *coswot:PropertyOfInterest*, c'est à dire une caractéristique spécifique à une entité identifiée du monde réel, soit une instance de *coswot:PropertyKind*, c'est à dire un archétype de propriété qui peut s'appliquer à différentes entités.

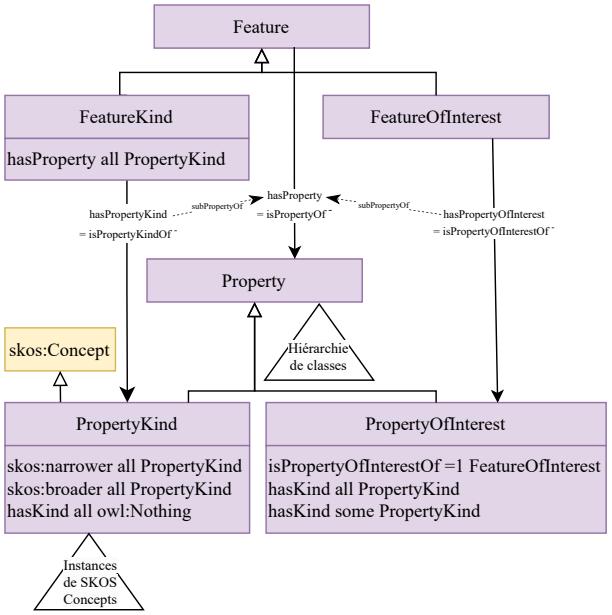


FIGURE 7 – Diagramme Chowlk du modelet Properties

Ainsi dans notre exemple, la propriété que nous modélisons est la température de l'air du bureau 429 du bâtiment EF. Comme le montre la figure 8, cette ressource est une instance de la classe *coswot:PropertyOfInterest* car elle qualifie un objet unique du monde réel. Elle est de type *coswot:AirTemperature* qui est une instance de *coswot:Property* que l'on pourrait trouver dans une taxonomie en ligne. La figure 8 indique les différentes cibles (« target ») du thermomètre.

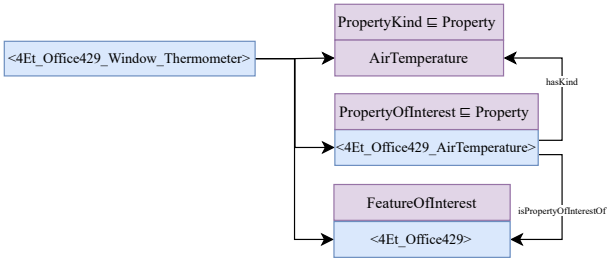


FIGURE 8 – Diagramme Chowlk de la témpérature de l'air du bureau 429

5.5 Property Values

Le modelet **coswot:core/property_values** décrit comment indiquer la valeur d'une grandeur physique. Il spécifie également comment associer une grandeur physique à une propriété d'une entité. Une valeur de propriété peut être utilisée pour définir des hiérarchies SKOS dans les archétypes. Une instance de *coswot:PropertyValue* est liée à une valeur et à une unité. Nous recommandons d'utiliser *cdt:ucum* pour associer directement l'unité à la valeur.

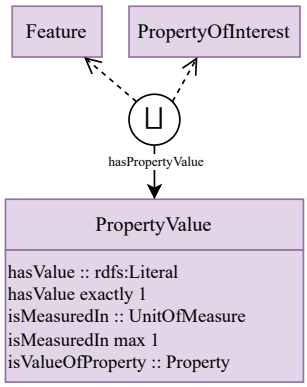


FIGURE 9 – Diagramme Chowlk du modelet Property values

Ainsi, dans nos exemples précédents nous avons utilisé plusieurs fois des instances de *coswot:PropertyValue*, pour définir l'archétype des bureaux de 20 m2 4, l'archétype des capteurs de précision 11.

5.6 Samples

Le modelet **coswot:core/samples** décrit les échantillons des entités d'intérêt c'est à dire les parties de l'entité d'intérêt (en général des prélèvements) sur lesquelles les observations seront effectuées. Une instance de *coswot:Sample* est forcément une instance de *coswot:FeatureOfInterest*. Dit

autrement, un échantillon est forcément un objet identifié du monde réel.

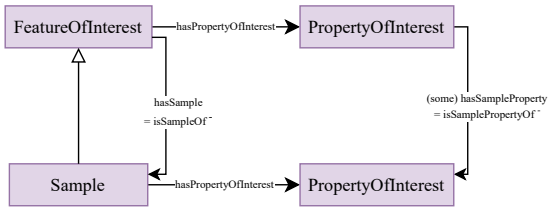


FIGURE 10 – Diagramme Chowlk du modelet Samples

Ainsi dans notre exemple, les capteurs qui mesurent la température de l'air ou la concentration de CO2 contenue dans l'air, n'observent que l'air situé dans leur environnement et non pas l'intégralité de l'air dans le bureau 429. Par convention des constructeurs de capteurs, l'échantillon d'air observé correspond à un volume d'air d'1 m3 situé autour de la sonde [9]. Comme le montre la figure 11, un échantillon est créé pour chaque capteur, instance de la classe *coswot:Sample* avec ses propriétés associées.

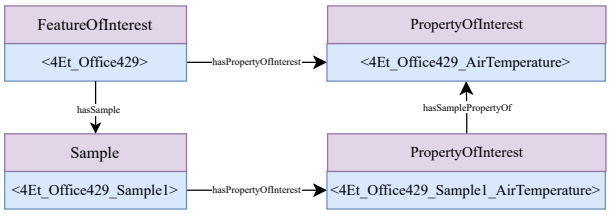


FIGURE 11 – Diagramme Chowlk d'un échantillon d'air observé par un capteur de CO2 dans le bureau 429

5.7 Procedures

Le modelet **coswot:core/procedures** décrit les procédures et leurs exécutions. Les exécutions de procédures généralisent les observations, les prédictions, et les actuations. Il s'agit d'un patron général représentant toute exécution d'une procédure définie qu'elle soit exécutée par un appareil ou par un humain. Une instance de *coswot:ProcedureExecution* a une date de début et une date de fin. Elle est liée à l'entité *coswot:FeatureOfInterest* qui l'exécute et à sa procédure *coswot:Procedure*. Une procédure peut prendre la forme d'un algorithme pour un appareil, ou d'un guide méthodologique pour un humain.

5.8 Observations

Le modelet **coswot:core/observations** définit les capteurs et décrit l'exécution de procédures d'observations. Il spécialise le modelet procédure et fonctions_commandes. Une instance de *coswot:Observation* doit forcément observer au moins une chose que ce soit une propriété *coswot:Property*, une entité *coswot:Feature* ou un état *coswot:State*.

9. Nous pourrions construire un archétype de capteur ayant un échantillon d'1 m3 qui serait associé à tous les capteurs du bâtiment, ce qui n'est pas très discriminant

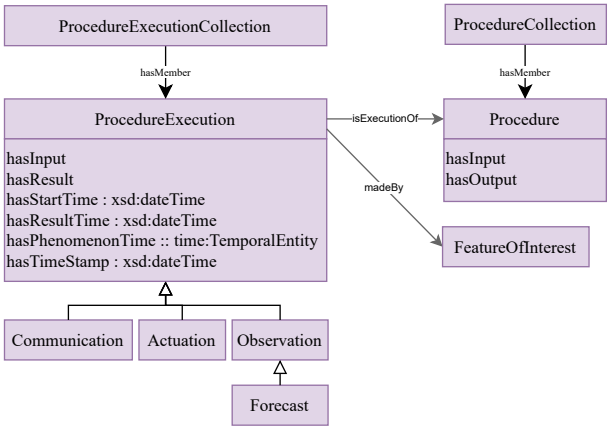


FIGURE 12 – Diagramme Chowlk du modelet Procedures

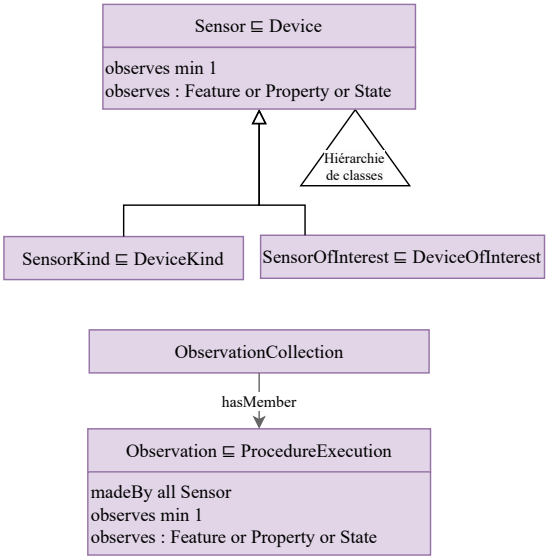


FIGURE 13 – Diagramme Chowlk du modelet Observations

Nous prenons l'exemple de la figure 14 qui décrit la mesure effectuée par le capteur de CO2 de la fenêtre. La représentation du capteur est instance des classes *coswot:DeviceOfInterest* et *coswot:GazSensor*. Cette mesure porte sur l'échantillon d'air localisé autour de la fenêtre, dont la représentation est instance de la classe *coswot:Sample*. cet échantillon a une propriété identifiée, instance des classes *coswot:PropertyOfInterest* et *coswot:CarbonDioxydeConcentrationInAir*. Cette mesure a pour résultat 1500 ppm. Il s'agit d'une mesure instantanée effectuée le 6 juillet 2021 à 09 :30 et 02 secondes.

5.9 Aggregation

Le modelet **coswot:core/aggregations** décrit les agrégations qui sont une spécialisation des observations. Une instance de *coswot:Aggregation* prend en entrée un ensemble d'observations représentée par une instance de la classe *coswot:ObservationCollection*, et produit un résultat, représenté par une instance de *coswot:PropertyValue*.

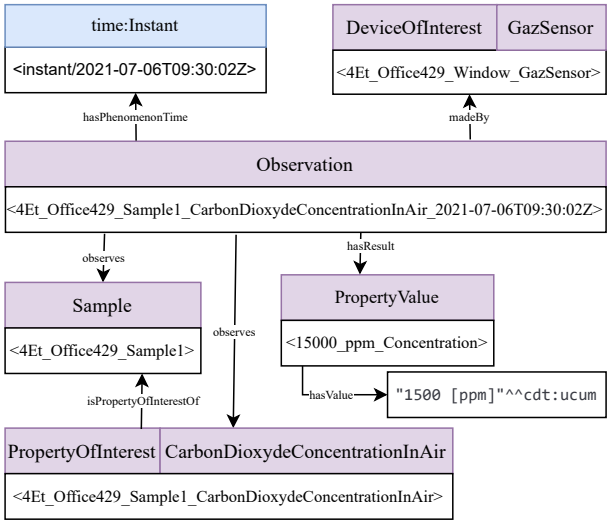


FIGURE 14 – Diagramme Chowlk d'une mesure du capteur de CO2 de la fenêtre

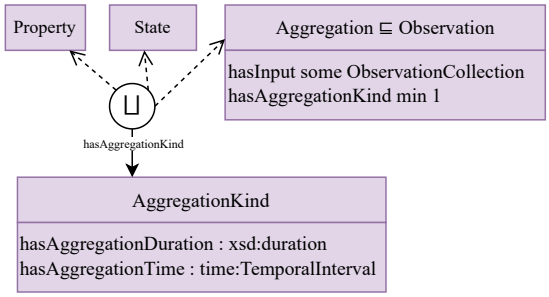


FIGURE 15 – Diagramme Chowlk du modelet aggregations

Dans notre exemple représenté sur la figure 16, notre application agrège les mesures instantanées des 3 capteurs de CO2. Ces capteurs mesurent tous à une fréquence de 5 secondes et les servients de la porte et du tableau envoient instantanément leur mesure au servient de la fenêtre. Ainsi, le servient de la fenêtre agrège les mesures produites dans une fenêtre temporelle de 5 secondes. Cette agrégation prend en entrée les trois mesures de capteurs de CO2 regroupées dans une collection d'observations non représentées dans le diagramme. Cette agrégation observe le bureau 429 et plus particulièrement la propriété identifiée, instance des classes *coswot:PropertyOfInterest* et *coswot:CarbonDioxydeConcentrationInAir*.

5.10 Communication

Le modelet **coswot:core/communications** vise à décrire les communications, et les systèmes de communications. Il spécialise le modelet procedures. Les servients de l'architecture CoSWoT représentent des systèmes de communications qui s'échangent des messages, chacun représentant l'identifiant d'un graphe nommé, comportant l'information échangée.

Dans notre exemple de la figure 18, le servient de la porte envoie l'identifiant du graphe nommé contenant ces deux

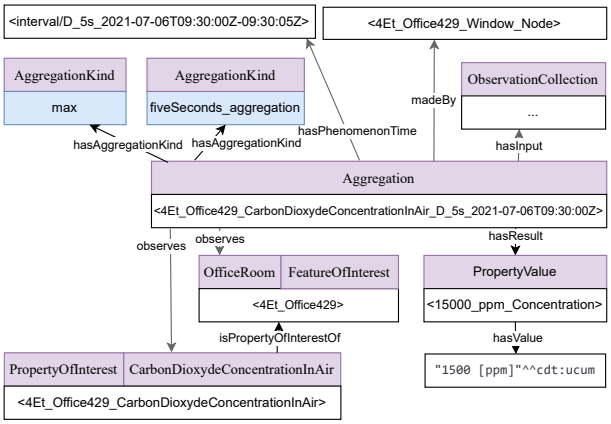


FIGURE 16 – Diagramme Chowlk de l'agrégation des mesures des capteurs de CO2 du bureau 429

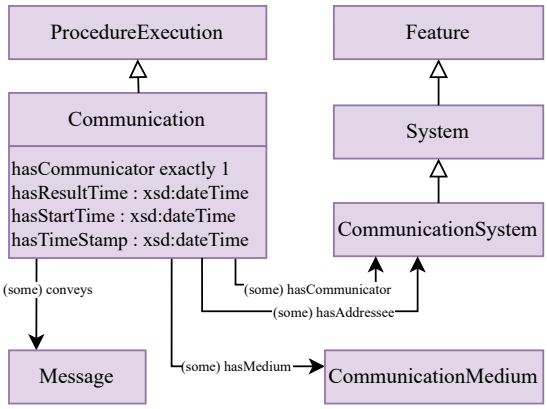


FIGURE 17 – Diagramme Chowlk du modelet communications

dernières mesures : mesure de concentration de CO2 et température. Cette communication a deux destinataires, le servient du radiateur et le servient de la fenêtre. Le communication se fait sur le réseau eduroam.

6 Conclusion

Cet article présente l'ontologie CoSWoT, qui vise à améliorer l'interopérabilité sémantique dans le domaine du Web des Objets, en considérant le contexte des applications constituées d'objets distribués sur le Web. Dans ce cadre, la conception de l'ontologie CoSWoT réutilise les ontologies de références du domaine du WoT : SSN/SOSA, TD, et SAREF. Le développement de l'ontologie CoSWoT prend appui sur une méthodologie agile d'ingénierie d'ontologies, ACIMOV, pour permettre la réutilisation des ontologies références, tout en garantissant le respect des principes de modularité, d'agilité, et bénéficier d'outils d'intégration et de développement continus. L'un des objectifs de l'ontologie CoSWoT est d'intégrer les ontologies références, tout en réconciliant la définition des concepts clés du domaines. Ce travail a donnée lieu au patron de conception architectural « *Kinds of X and X of Interest* », qui différencie les

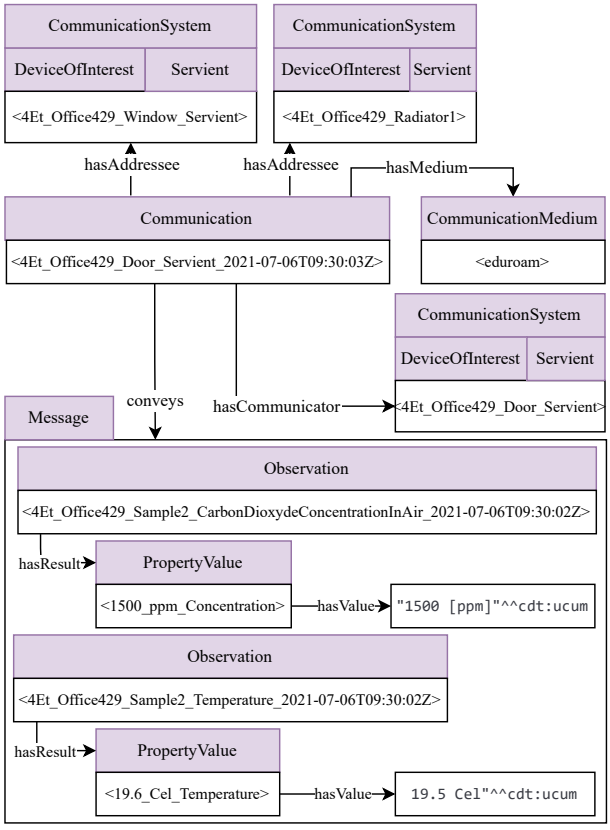


FIGURE 18 – Diagramme Chowlk de la communication entre deux servients

concepts clés de ces ontologies (ex. *FeatureOfInterest*, *Device*,..), en distinguant leurs instances spécifiques de celles génériques. L'usage de ce patron a pour but de clarifier la représentation des concepts et de faciliter l'usage des propriétés associées. Sur la base de ce patron, CoSWoT est conçue comme un ensemble de modelets (modules) core regroupant les concepts WoT communs à tous les domaines d'applications verticaux. Des vues de l'ontologie sont par la suite réalisées en spécialisant une partie des modelets, pour permettre de répondre aux besoins du domaine d'application objet de l'étude. Cela a notamment été illustré dans l'article sur le cas du bâtiment intelligent EF.

Le travail sur l'ontologie CoSWoT se poursuit pour concrétiser son usage dans plus de cas d'usage (issus de l'agriculture numériques par exemple). Il permettrait à terme de formuler des recommandations quant à l'intégration des spécificités des objets contraints dans les ontologies de référence, renforçant l'interopérabilité sémantique au delà des architectures centralisées.

Remerciements

Le projet CoSWoT est financé par l'agence nationale de la recherche sous la référence ANR-19-CE23-0012.

Références

[1] V. Charpenay and M. Kovatsch. Hypermedia Controls Ontology, Editor draft, 10 May 2023. W3c working group draft, W3C, May 2023.

[2] V. Charpenay, M. Lefrançois, and M. Poveda Villalón. JSON Schema in RDF, Editor draft, 10 May 2023. W3c working group draft, W3C, May 2023.

[3] V. Charpenay, M. Lefrançois, M. Poveda Villalón, and S. Käbisch. Thing Description (TD) Ontology, Editor draft, 10 May 2023. W3c working group draft, W3C, May 2023.

[4] Simon Cox. Observations and Measurements. OGC project document 10-025r1, OGC, March 2011.

[5] Laura Daniele, Frank den Hartog, and Jasper Roes. Created in close interaction with the industry : the smart appliances reference (saref) ontology. In *FOMI 2015*, pages 100–112. Springer, 2015.

[6] ETSI TC SmartM2M. SmartM2M ; SAREF reference ontology patterns. Technical Specification ETSI TS 103 548 V1.2.1, ETSI, January 2024.

[7] ETSI TC SmartM2M. SmartM2M ; Smart Applications ; Reference Ontology and oneM2M Mapping. Technical Specification ETSI TS 103 264 V3.2.1, ETSI, January 2024.

[8] Isaac Fatokun, Arun Raveendran Nair Sheela, Thamer Mecharnia, Maxime Lefrançois, Victor Charpenay, Fabien Badeig, and Antoine Zimmermann. Modular knowledge integration for smart building digital twins. In *LDAC'23*, 2023.

[9] 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. In *Energy Smart Appliances : Applications, Methodologies, and Challenges*. Wiley, 2023.

[10] Armin Haller, Krzysztof Janowicz, Simon J D Cox, Danh Le Phuoc, Kerry Taylor, and Maxime Lefrançois. Semantic Sensor Network Ontology. W3C Recommendation, W3C, October 19 2017.

[11] Armin Haller, Krzysztof Janowicz, Simon JD Cox, Maxime Lefrançois, Kerry Taylor, Danh Le Phuoc, Joshua Lieberman, Raúl García-Castro, Rob Atkinson, and Claus Stadler. The sosa/ssn ontology : a joint w3c and ogc standard specifying the semantics of sensors, observations, actuation, and sampling. *Semantic Web-Interoperability, Usability, Applicability an IOS Press Journal*, 56, 2019.

[12] Armin Haller, Krzysztof Janowicz, Simon J.D. Cox, Maxime Lefrançois, Kerry Taylor, Danh Le Phuoc, Josh Lieberman, Raúl García-Castro, Rob Atkinson, and Claus Stadler. Sosa : A lightweight ontology for sensors, observations, samples, and actuators. *Semantic Web Journal*, 2018.

[13] Fatma-Zohra Hannou, Victor Charpenay, Maxime Lefrançois, Catherine Roussey, Antoine Zimmermann, and Fabien Gandon. The ACIMOV Methodology : Agile and Continuous Integration for Modular Ontologies and Vocabularies. In *2nd Workshop on Modular Knowledge associated with FOIS 2023*, 2023.

[14] Krzysztof Janowicz, Armin Haller, Simon J.D. Cox, Danh Le Phuoc, and Maxime Lefrançois. Sosa : A lightweight ontology for sensors, observations, samples, and actuators. *Journal of Web Semantics*, 2018.

[15] Sebastian Kaebisch, Takuki Kamiya, Michael McCool, and Victor Charpenay. Web of Things (WoT) Thing Description. Candidate Recommendation, W3C, May 16 2019.

[16] M. Koster and E. Korkan. Web of Things (WoT) Binding Templates. W3c working group note, W3C, January 2020.

[17] Ryuichiand Kovatsch, Matthiasand Matsukura, Michael Lagally, Toru Kawaguchi, Kunihiko Toumura, and Kazuo Kajimoto. Web of Things (WoT) Architecture. Candidate Recommendation, W3C, May 16 2019.

[18] Laurent Lefort, Cory Henson, and Kerry Taylor. Semantic Sensor Network XG Final Report. W3C Incubator Group Report, W3C, June 28 2011.

[19] Nicolas Seydoux, Maxime Lefrançois, and Lionel Médini. Positionnement sur le Web Sémantique des Objets. In *IC 2019*, pages 8–25, July 2019.

Posters et démonstration

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

J. Mba Kouhoue[1,3], M. Lefrançois[2], A. Lesage[3], J. Lonlac[1], A. Doniec[1], S. Lecoeuche[1].

[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.lecoeuche@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 SAREF 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 physiques, 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

2. `https://docs.brickschema.org/`

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/testt/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 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 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

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 Rokonuzzaman Reza. Chatgpt vs. bard : a comparative study. *Authorea 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.

EpiStrat-Eval: outil d'évaluation des stratégies d'extraction d'informations spatiales pour la veille en épidémiologie

Y. Mahdoubi[1], S. Valentin[2,3], N. Idrissi[1], M. Roche[2,3]

[1]Faculté des Sciences et Techniques Béni Mellal, Maroc
[2]TETIS, Univ Montpellier, AgroParisTech, CIRAD, CNRS, INRAE, Montpellier, France
[3]CIRAD, UMR TETIS, Montpellier F-34398, France

Résumé

Les articles de presse publiés en ligne jouent un rôle essentiel dans la détection précoce des foyers de maladies. Toutefois, l'extraction et le traitement de l'information spatiale à partir de ces données textuelles demeurent un défi majeur. C'est dans cette optique que l'application EpiStrat-Eval, en lien avec l'outil de surveillance PADI-web, offre une interface cartographique permettant aux épidémiologistes d'évaluer diverses stratégies d'extraction d'entités spatiales. Cette initiative contribue ainsi à l'amélioration continue des systèmes de surveillance.

Mots-clés

données textuelles, entités spatiales, évaluation

Abstract

Online news articles play a vital role in the early detection of disease outbreaks. However, extracting and processing spatial information from this textual data remains a significant challenge. To address this, EpiStrat-Eval, in collaboration with the PADI-web monitoring tool, offers a mapping interface that enables epidemiologists to evaluate various strategies for extracting spatial entities. This initiative contributes to the continuous improvement of monitoring systems.

Keywords

textual data, spatial information, evaluation

1 Introduction

1.1. Surveillance basée sur les événements et information spatiale

Ces dernières années, de nombreux travaux se sont intéressés à l'utilisation de données textuelles issues de sources informelles pour la surveillance des épidémies. Alors que les organisations telles que l'Organisation mondiale de la santé animale diffusent des notifications officielles, les sources informelles telles que les journaux en ligne fournissent des informations de niveaux de fiabilité variables pouvant s'avérer plus précoces [1].

Les outils de surveillance événementielle, tels que HealthMap [2] et PADI-web [3], permettent d'automatiser différentes étapes de la chaîne de traitement des données textuelles en y intégrant des approches basées sur l'Intelligence Artificielle et l'apprentissage automatique, de la collecte de sources jusqu'à la production d'informations épidémiologiques pouvant être analysées. L'identification précise des localisations joue un rôle important dans l'évaluation des risques épidémiologiques, permettant aux épidémiologistes de détecter une possible émergence ou d'appréhender la propagation d'une maladie. Cette spatialisation est indispensable, *in fine*, à la mise en place de mesures de lutte et de protection adaptées. Cependant, cette tâche se heurte à des défis méthodologiques propres à l'extraction automatique d'information spatiale à partir de données textuelles dans un contexte événementiel. D'une part, la détection des entités spatiales peut produire des erreurs en raison d'ambiguïtés fréquentes, telles que l'utilisation de noms de lieux génériques ou d'abréviations, la confusion entre des noms de personne et de lieux, etc. D'autre part, les articles de presse peuvent contenir des informations spatiales qui ne sont pas associées aux foyers de maladie. Si ces informations connexes sont trop nombreuses, le processus d'extraction génère une importante quantité d'information non-pertinente, augmentant la tâche de filtre par l'expert et posant le risque de générer de fausses alertes. Enfin, la tâche de *geocoding*, indispensable à la représentation cartographique des entités, ajoute une couche supplémentaire de complexité à ce processus d'identification en raison des nombreuses homonymies et ambiguïtés [4].

1.2. Motivation

Face à ces verrous, différentes stratégies d'extraction de l'information spatiale combinant heuristiques simples et approches fondées sur l'Intelligence Artificielle peuvent être intégrées dans les outils de veille. Notre objectif consiste à proposer une application permettant d'évaluer la performance de ces différentes stratégies. Nous nous adossons aux travaux dédiés à l'outil PADI-web, un outil automatique pour la veille en épidémiologie animale basé sur les articles publiés en ligne [3]. Nous proposons EpiStrat-Eval[1], une interface conviviale pour aider les épidémiologistes dans leur processus décisionnel lors de la validation de la localisation des foyers de maladie[2].

[1] https://github.com/ysfmh14/EpiStratEval.git
[2] Vidéo de démonstration

1.3. Travaux connexes: interfaces et évaluation

L'avènement des technologies numériques a révolutionné le domaine de la surveillance épidémiologique, en offrant de nouveaux outils pour collecter, analyser et visualiser les données de santé. De nombreuses plateformes interactives, principalement appliquées en santé humaine, permettent de visualiser et d'interagir avec des données spatio-temporelles, afin de faciliter la détection d'épidémies à partir de données officielles [6, 7]. Cependant, ces approches s'appuyant sur des bases de données officielles, la notion de validation spatiale n'y est pas intégrée. Dans le cadre de la surveillance des sources informelles, HealthMap [2] propose une carte interactive intégrant les diverses sources d'information. Chaque article est associé à une localisation (ville, région ou pays) et est représenté sur la carte via un cercle dont la taille et la couleur représentant le niveau d'alerte. Les utilisateurs enregistrés peuvent évaluer le niveau de risque de chaque alerte grâce à un score de 1 à 5. EpidNews [8] propose une interface utilisateur intuitive permettant d'explorer les données épidémiologiques des maladies animales issues à la fois des sources officielles et non officielles. L'objectif de l'outil est de mettre en évidence les alertes issues de sources non-officielles dans des régions non couvertes par les données officielles.

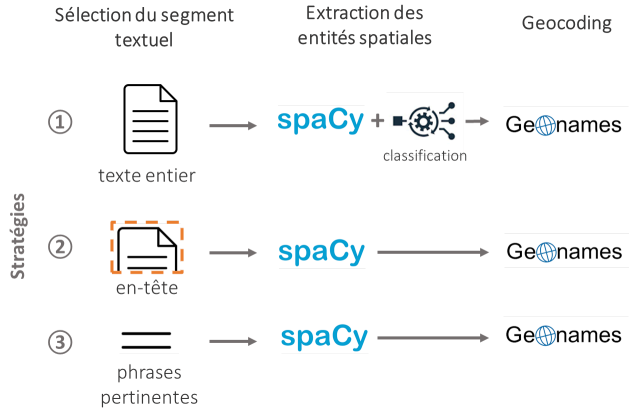
2 Données spatiales issues d'un système de veille

Dans cette section, nous présentons les différentes stratégies de spatialisation des foyers de maladie par PADI-web, et les étapes de leur préparation pour l'outil EpiStrat-Eval.

2.1. Stratégies d'extraction de PADI-web

L'interface de PADI-web permet d'extraire les informations spatiales relatives aux épidémies à partir d'articles publiés en utilisant six stratégies distinctes. Nous présentons dans la Figure 1 trois de ces stratégies, qui sont appliquées sur les articles automatiquement classés comme "Déclaration de foyer".

Figure 1. Stratégies d'extraction d'entités spatiales dans l'outil PADI-web.



La première étape consiste à préfiltrer le segment textuel à partir duquel vont être extraites les localisations : le texte entier (stratégie 1), le titre et les 300 premiers caractères (stratégie 2), les phrases classées comme pertinentes, c'est–à-dire relatives à un foyer récemment déclaré, grâce à une approche fondée sur l'apprentissage automatique détaillée dans [9] (stratégie 3). L'extraction des entités spatiales est ensuite réalisée en utilisant un modèle pré-entraîné de la librairie spaCy [10]. Dans la stratégie 1, une étape de sélection automatique des localisations pertinentes (associées à un foyer) via un module de classification est appliquée [5]. Le *geocoding*, qui permet de d'affecter des coordonnées géographiques à chaque entité spatiale, est réalisé grâce à la base de données géographiques GeoNames [11].

2.2. Préparation et nettoyage des données

Avant d'intégrer les données fournies par PADI-web dans notre application, nous effectuons plusieurs traitements pour obtenir un ensemble de données bien organisé, exempt de données erronées ou manquantes. Voici les traitements effectués :

(1) Suppression des lignes comportant des valeurs de latitude et de longitude nulles (ne pouvant pas être représentées sur la carte).
(2) Pour un même article, suppression des lignes représentant la même entité géographique, afin de réduire les redondances sur la carte.
(3) Ajout de la colonne *validation* afin d'enregistrer les validations des entités spatiales par les experts.

Nous intégrons également la colonne *location type* afin d'indiquer la hiérarchie de la localisation (i.e. continent, pays, région, ville), en utilisant le code et la classe géographique associés à l'identifiant GeoNames de chaque entité.

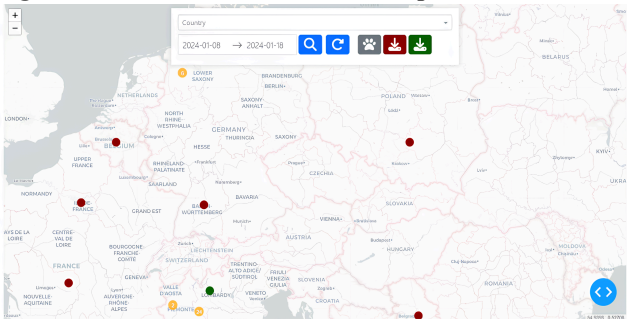
3 EpiStrat-Eval

Notre application permet de rendre les données extraites par PADI-web visuellement accessibles, afin d'accompagner la prise de décision des épidémiologistes. Cependant, l'objectif principal de notre application est de fournir aux experts la possibilité de comparer les différentes stratégies d'extraction d'informations spatiales, pour, *in fine*, identifier la ou les plus pertinentes.

3.1 Description générale de l'interface

L'interface principale de notre application, illustrée dans la Figure 2, se divise en deux composants majeurs. Le premier est une carte affichant les positions correspondant aux détections des maladies, tandis que la seconde est une barre de navigation contenant plusieurs composants. Parmi ces éléments, l'outil propose de filtrer les données en sélectionnant un pays et/ou en définissant une fenêtre temporelle. Cette barre offre également la possibilité d'afficher la liste des hôtes (espèces animales) qui sont extraits (bouton gris), ainsi que la possibilité de rafraîchir les données en cas de modification, ou encore de charger un fichier contenant des données déjà évaluées.

Figure 2. Visualisation de l'interface EpiStrat-Eval



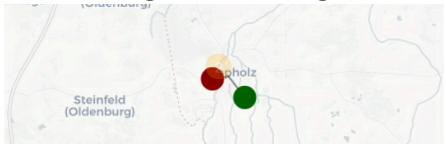
3.2 Représentation de l'information spatiale

Pour représenter les entités spatiales nous avons opté pour une représentation cartographique, dans laquelle les données sont symbolisées par des cercles de deux couleurs distinctes : les cercles rouges signalent les localisations de foyers extraites par la première stratégie, tandis que les cercles verts représentent les extractions de la deuxième stratégie. De plus, des cercles oranges sont utilisés pour condenser les informations lorsqu'un grand nombre de détections se situent dans une même zone. Cette zone, matérialisée en bleu, est établie en spécifiant une distance maximale entre les détections : toutes celles qui se trouvent à une distance égale ou inférieure les unes des autres sont regroupées dans une même région (Figure 3). Cette distance est un paramètre pouvant être adapté au contexte épidémiologique. Les cercles oranges offrent une représentation abstraite de la région en indiquant le nombre de détections présentes. Le détail des détections dans une région identifiée est affiché en cliquant sur la zone.

Figure 3. Représentation de détections multiples

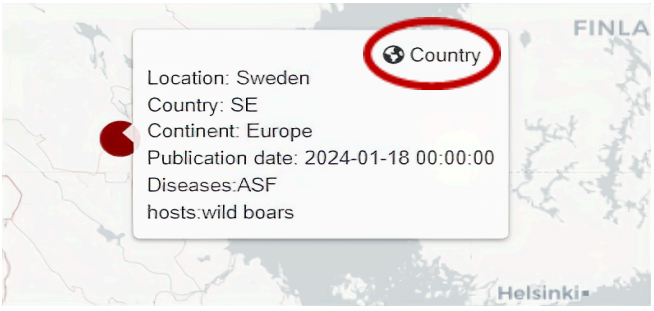
Les cercles orange mentionnés dans le paragraphe précédent sont également utilisés pour résoudre les problèmes de superposition, lorsqu'un foyer est détecté plusieurs fois dans la même localisation, soit par une seule stratégie, soit par les deux. Cela entraîne le chevauchement des cercles qui rend difficile la distinction des détections dans cette localisation. Les cercles orange permettent de clarifier cette situation en fournissant une représentation plus claire des détections multiples dans une même zone (Figure 4).

Figure 4. Visualisation d'une localisation identifiée par la stratégie 1 et la stratégie 2



Nous permettons également aux experts d'accéder à diverses informations concernant chaque détection en plaçant le curseur sur le cercle correspondant à la détection (Figure 5). Ces informations comprennent la valeur de la localisation dans le texte, le pays correspondant, la date de publication de l'article, et le nom de la maladie et le ou les hôtes extraits à partir de l'article. Le type de localisation (ville, région, pays ou continent) est également indiqué et permet de contextualiser la position d'une détection : si une région administrative est extraite, ses coordonnées géographiques associées vont être son centroïde.

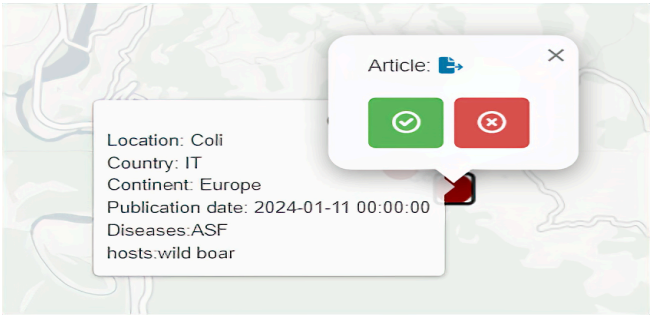
Figure 5. Visualisation des métadonnées associées à une détection



3.3 Evaluation

Nous allons à présent détailler l'aspect central de notre outil, permettant aux experts de valider ou non les localisations identifiées par chaque stratégie.

Figure 6. Validation de la pertinence d'une information spatiale



Pour pouvoir interpréter la pertinence d'une localisation, EpiStrat-Eval propose une redirection vers l'article dont l'entité spatiale a été extraite. L'expert peut ensuite valider ou non la pertinence de l'entité spatiale (Figure 6). Une fois ce choix effectué, un ticket est associé aux détections qui ont reçu une évaluation et le label (valide ou non valide) est ajouté à une colonne appelée "Validation". Les fichiers générés peuvent être téléchargés et ré-utilisés comme fichiers d'entrée pour reprendre une évaluation en cours ou modifier des valeurs existantes.

3.4. Outils

Les fonctionnalités d'EpiStrat-Eval reposent sur plusieurs outils. La bibliothèque Python *Dash* a été utilisée pour

concevoir des tableaux de bord réactifs. Cette approche permet aux composants de partager des informations entre eux grâce à l'utilisation de "callbacks" et permet d'intégrer des éléments interactifs. Nous avons également utilisé la bibliothèque Python *Folium* pour créer une carte interactive. Enfin, la bibliothèque Python *Flask* a été intégrée pour consommer des API, émises depuis un code JavaScript associé à la carte.

4 Cas d'étude

Nous avons réalisé un cas d'étude sur un échantillon d'articles extraits de l'outil PADI-web. L'évaluation a été réalisée par une épidémiologiste en santé animale. Nous avons sélectionné les articles de PADI-web classés comme des déclarations de foyers de maladies animales publiés entre les 22/02/2024 et le 26/02/2024 (31 articles). Nous avons comparé la stratégie 1 (extraction à partir de l'ensemble de l'article puis sélection automatique) avec la stratégie 2 (extraction limitée à l'en-tête de l'article, sans sélection automatique). Nous avons choisi de comparer ces deux stratégies car elles répondent à deux niveaux de complexité méthodologique : la stratégie 2 repose sur l'hypothèse que les entités spatiales associées à un événement sont présentes en début d'article, tandis que la 1ère stratégie évalue la pertinence de chaque entité spatiale quelle que soit leur position dans le texte. Avant filtrage, les deux stratégies génèrent 157 et 110 entités spatiales, respectivement. Après filtrage (section 2.2), 93 entités spatiales sont obtenues via la stratégie 1 *versus* 37 entités via la stratégie 2.

Pour être labellisées comme valides, les localisations doivent répondre à deux critères : (1) correspondre à une entité spatiale citée dans l'article et (2) correspondre à un foyer de maladie. L'outil ne permet pas d'évaluer l'étape de geoparsing (attribution de coordonnées spatiales), qui est une étape indépendante de l'identification de localisations pertinentes. Dans ce cas d'étude, nous avons donc considéré comme valides les localisations répondant aux deux critères précédents, même associées à des coordonnées spatiales incorrectes.

Tableau 1. Evaluation des stratégies d'extraction 1 et 2

		Stratégie 1	**Stratégie 2**
Articles	Localisés	93.5% (29/31)	61% (22/31)
	Non localisés	6.5% (2/31)	29% (9/31)
Localisations	Valides	75.3% (70/93)	94.6% (35/37)
	Non valides	24.7% (23/93)	5.4% (2/37)

La stratégie plus conservatrice (stratégie 2) permet d'obtenir une excellente précision (94.6% des entités extraites correspondent à des foyers mentionnés dans un article). Cependant, 29% des articles ne sont pas associés à une localisation (dans ces articles, aucune entité spatiale n'a été détectée par la librairie spaCy dans l'en-tête de l'article). La stratégie 1 génère davantage de localisations non valides, mais permet détecter un plus grand nombre absolu d'entités spatiales valides, et notamment de spatialiser la quasi-totalité des articles (93.5%). Les deux articles non spatialisés contiennent des entités spatiales bien détectées par GeoNames, mais qui n'ont pas été considérées comme liées à un foyer suite à la classification automatique. Dans ce cas d'étude, la stratégie 1 est à privilégier afin d'éviter un trop grand nombre de faux négatifs (localisations pertinentes associées à des foyers non détectés par la stratégie 1).

L'extension de ce cas d'étude à l'ensemble des stratégies d'extraction de PADI-web et en combinant l'évaluation de plusieurs experts, permettra d'évaluer de manière plus représentative les informations spatiales issues d'approches automatiques. Cette évaluation est indispensable à l'intégration de solutions fondées sur l'Intelligence Artificielle dans le cadre d'activités quotidiennes de veille épidémiologique.

5 Références

.[1] C. Paquet, D. Coulombier, R. Kaiser, and M. Ciotti. Epidemic intelligence : a new framework for strengthening disease surveillance in Europe. Eurosurveillance, 11(12) :5–6, December 2006.

[2] C. C. Freifeld, K. D. Mandl, B. Y. Reis, and J. S. Brownstein. HealthMap : Global Infectious Disease Monitoring through Automated Classification and Visualization of Internet Media Reports. Journal of the American Medical Informatics Association, 15(2) :150–157, March 2008.

[3] S. Valentin, E. Arsevska, J. Rabatel, S. Falala, A. Mercier, R. Lancelot, and M. Roche. PADI-web 3.0 : A new framework for extracting and disseminating fine-grained information from the news for animal disease surveillance. One Health, 2021.

[4] M. Gritta, M. T. Pilehvar, and N. Collier. A pragmatic guide to geoparsing evaluation. Language Resources and Evaluation, 54(3) :683–712, September 2020.

[5] E. Arsevska, S. Valentin, J. Rabatel, J. de Goër de Hervé, S. Falala, R. Lancelot, and M. Roche. Web monitoring of emerging animal infectious diseases integrated in the French Animal Health Epidemic Intelligence System. PLOS ONE, 13(8) :e0199960, August 2018.

[6] R. Arias-Carrasco, J. Giddaluru, L. E. Cardozo, F. Martins, V. Maracaja-Coutinho, and H. I. Nakaya. OUTBREAK : a user-friendly georeferencing online tool for disease surveillance. Biological Research, 54, 2021. Publisher : BMC.

[7] L. N. Carroll, A. P. Au, L. T. Detwiler, T.-c. Fu, I. S. Painter, and N. F. Abernethy. Visualization and analytics tools for infectious disease epidemiology : A systematic review. Journal of Biomedical Informatics, 51 :287–298, October 2014.

[8] R. Goel, S. Valentin, A. Delaforge, S. Fadloun, A. Sallaberry, M. Roche, and P. Poncelet. EpidNews : Extracting, exploring and annotating news for monitoring animal diseases. Journal of Computer Languages, 56 :100936, February 2020.

[9] S. Valentin, E. Arsevska, A. Vilain, V. De Waele, R. Lancelot, and M. Roche. Elaboration of a new framework for fine-grained epidemiological annotation. Scientific Data, 9(1) :655, October 2022.

[10] spaCy · Industrial-strength Natural Language Processing in Python.

[11] D. Ahlers. Assessment of the Accuracy of GeoNames Gazetteer Data. In Proceedings of the 7th Workshop on Geographic Information Retrieval, pages 74–81, New York, NY, USA, 2013. ACM.

Implanter une approche hybride dans une démarche d'ingénierie de la connaissance pour manager les avis techniques relatifs au retour d'expérience d'exploitation d'un équipement sensible complexe

S. Boblet[1], T. Cartié[1], A. Berger[2], J-P. Cotton[2], F. Vexler[2]

[1] TechnicAtome,
Lieu-dit « Les Hautes Rives », Route de Saint Aubin,
91190 Villiers-le-Bâcle, France - www.technicatome.com
{sebastien.boblet, thierry.cartie}@technicatome.com
[2] Ardans SAS,
6 rue Jean Pierre Timbaud, « Le Campus » Bâtiment B1,
78180 Montigny-le-Bretonneux, France - www.ardans.fr
{aberger, jpcotton, fvexler}@ardans.fr,

3 juin 2024

Résumé

*Comment manager les avis techniques relatifs au retour d'expérience d'exploitation de manière efficiente dans une organisation qui n'a jamais fait appel aux techniques et méthode de l'ingénierie de la connaissance ? Cet article précise comment un industriel du nucléaire et du secteur de la défense s'est approprié une telle démarche adaptée à son contexte organisationnel « TA KM » qui s'inscrit dans le cadre de l'*ISO30401*, pour construire un système complet avec une application «* SARBACANES *» pour supporter son process métier et pour pérenniser son savoir-faire et ses expertises dans une base de connaissance. Au-delà du très classique transfert de connaissance entre expert et spécialiste métier,* SARBACANES *révèle aussi la capacité d'une telle ingénierie à offrir comme résultat une exploitation polyfonctionnelle. La modélisation a été accélérée par la mise en œuvre d'un outil adapté à ce type d'opération : la plateforme Ardans Knowledge Maker® .*

Mots-clés

SARBACANES *, Ingénierie de la connaissance, Gestion de Retour d'expérience, Modélisation de process métier, Recueil d'expertise, Gestion et Management, Sûreté nucléaire, Exploitation documentaire, Transfert de connaissance,* TA KM *, Ardans Knowledge Maker® ,* ISO30401.

Abstract

How can technical advice on operating experience feedback be managed efficiently in an organization that has never used knowledge engineering techniques and methods ? This article explains how an industrial company in the nuclear and defense sectors adopted such an approach, adapted to its « TA KM *» organizational context and falls within the* ISO30401 *framework, to build a complete system with a «* SARBACANES *» application to support its business processes and perpetuate its know-how and expertise in a knowledge base. know-how and expertise in a knowledge base. Over and above the classic transfer of knowledge between experts and business specialists,* SARBACANES *also reveals the ability of this type of engineering to deliver multi-functional operation. Modeling was accelerated by the use of a tool adapted to this type of operation : the Ardans Knowledge Maker® platform*

Keywords

SARBACANES *, Knowledge engineering, lessons learnt, feedback management, Business process modelling, Collection of expertise, Nuclear safety, Document exploitation, Knowledge Transfer,* TA KM*, Ardans Knowledge Maker® ,* ISO30401.

1 Introduction

Depuis 2019, les sociétés TechnicAtome et Ardans travaillent sur la mise en place d'un « *système métier* » qui allie méthode d'ingénierie de la connaissance, plate-forme de gestion de la connaissance, intégration au système d'information de production, conduite de changement pour un meilleur service rendu au client final fondé sur la meilleure efficience opérationnelle des spécialistes et experts de la société (Cf. Cartié & al. [6]). Cet article intègre la vision opérationnelle, la méthode d'élaboration du dispositif, la nature de l'approche hybride retenue (Cf. Boblet & al. [5]) et le retour d'expérience de la mise en exploitation et de la dynamique qui en découlent.

2 La description du contexte opérationnel

L'acteur opérationnel est un industriel du secteur de la défense, la société TechnicAtome . Le domaine concerné est la gestion du retour d'expérience (REX) (cf. Malvache & al. [10]) d'exploitation des équipements techniques que TechnicAtome conçoit et livre aux forces.

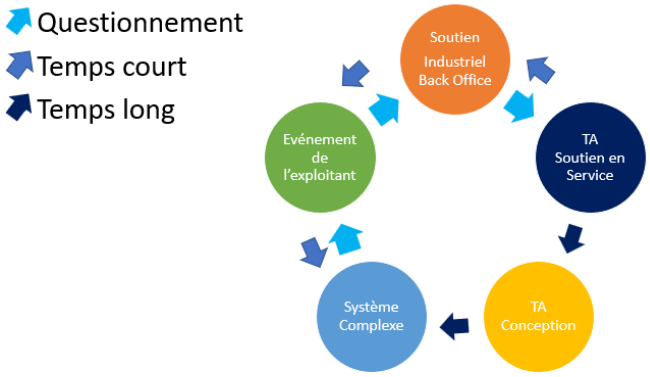


FIGURE 1 – Le retour d'expérience

Le sujet est le suivant : à partir d'un événement en exploitation sur l'équipement qui est un système complexe, la caractérisation du fait technique remonte à deux niveaux : soutien industriel puis soutien en service. Ainsi que le représente la Figure 1, les experts doivent retourner dans un « *temps court* » à l'exploitant un avis technique. Par ailleurs, dans un « *temps long* », cette analyse est complétée avec les concepteurs pour éventuellement gérer une intervention voire intégrer des évolutions lors d'une opération de maintenance future. Il s'agit de considérer alors l'impact sur le système de soutien ou bien la prise en compte des évolutions en conception dans le cas d'un impact sur le système principal. Les objectifs opérationnels sont nombreux : depuis l'optimisation des processus de capitalisation de REX jusqu'à l'analyse proprement dite du REX. Il y a aussi une priorité sur l'efficience dans l'analyse de l'événement en exploitation par rapport à tous les événements déjà survenus et analysés. Il y a bien sûr le fait de disposer d'un meilleur accès à la connaissance collective, et une fluidité dans les échanges entre exploitants et mainteneurs, comme entre mainteneurs et concepteurs. Enfin, l'évolution culturelle est un enjeu sous-jacent, avec la mise en place d'une approche collaborative permanente pour obtenir une amélioration continue effective.
In fine, l'objectif est de fournir la garantie de la qualité et de la fidélité du REX pour tous les acteurs impliqués.

3 La méthode TA KM pour conduire l'opération dans le temps

"Construire en commun un objet inconnu" (Cf. Grundstein [9]) nécessite un processus de confiance pour agréger au fil de l'eau cette matière "connaissance". Avant de s'interroger sur la technologie informatique d'IA il y a la question de la méthode et du phasage de l'opération afin d'être dans un processus d'appropriation et d'adhésion de la démarche "étape par étape".

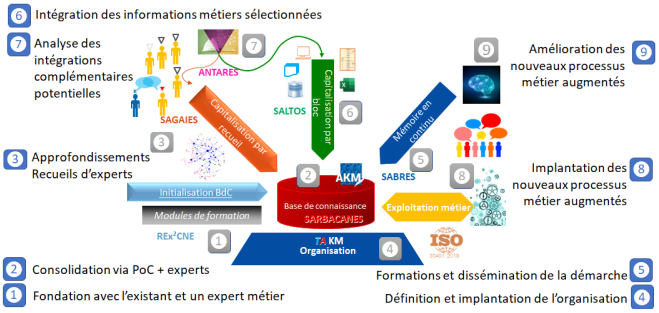


FIGURE 2 – Les différentes étapes de la méthode TA KM et ses facettes

Initialisation de la modélisation, structuration de la connaissance, nourrissage des éléments de la base de connaissance, validation de ces mêmes contenus, enrichissement par de nouveaux éléments d'expertise, injection d'éléments de connaissance en masse, formation des utilisateurs pilotes, élicitation du processus métier complet et implantation dans l'environnement informatique de production : voilà en quelques activités le squelette de ce que porte la méthode TA KM qui a conduit et irrigué l'opération depuis son lancement en 2019 (cf. Figure 2) et selon les principes de l'ISO30401(Cf. Berger [2]). On considère à ce stade que TA KM porte le « *système de gestion de la connaissance* » sur le métier, ce qui s'entend comme le *KMS* (Le KMS pour « *Knowledge Management System* » intègre la plate-forme logicielle à l'organisation de la gouvernance de la connaissance) au sens de l'ISO30401 [12].

4 Le positionnement de l'approche hybride

L'approche hybride s'appuie sur une méthode d'ingénierie des connaissances outillée éprouvée : Ardans *make*® et Ardans Knowledge Maker® . Afin d'apprécier Ardans Knowledge Maker® , la méthode Ardans *make*® et l'approche hybride *symbolique & sémantico syntaxique* le lecteur est invité à parcourir les articles ayant déjà traité de ces sujets à savoir : Mariot & al. [11], Besson & al. [4], Berger [1], Vexler & al. [13], Mary & al. [3] et Fourtout & al. [8].
L'hybridation est relative d'une part à une **modélisation symbolique** où des éléments de connaissances (on parle aussi d'articles ou de fiches) sont produits à partir de modèles, reliés à une ontologie élaborée au fil de la démarche, complétée par les liages inter-fiches validés par les sachants et une gestion de droits d'accès, et d'autre part à une **approche sémantico syntaxique** réalisé par apprentissage incrémental sur les contenus des éléments présents dans la base de connaissance et enrichi lors de la complétion d'une nouvelle fiche.
Le côté hybride a été nécessaire afin de satisfaire une ergonomie cognitive pour deux fonctionnalités essentielles (cf.

le point d'interrogation de l'article A13 avec ses deux types de liens vers les items de l'ontologie ou vers les autres Figure 3).

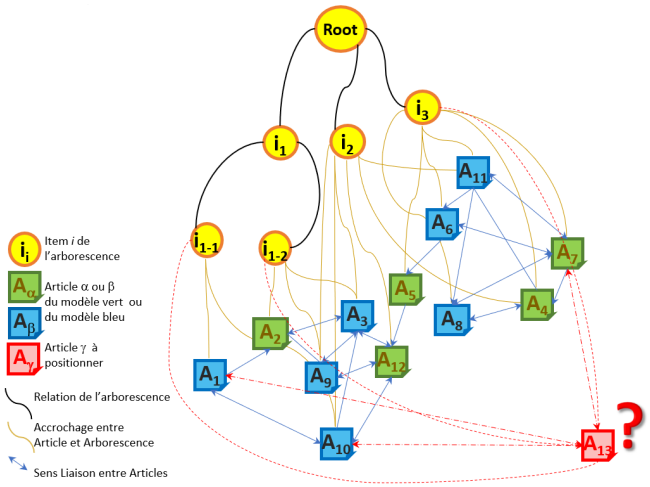


FIGURE 3 – La puissance de l'approche hybride : la modélisation associée à une approche sémantico syntaxique

La première est : comment être sûr de trouver les événements les plus proches de celui qu'il convient d'étudier et qui ont déjà été analysés et expertisés par le passé ?
La seconde est : comment garantir que l'élément de connaissance qui va être créé, est positionné correctement avec le liage pertinent (La « *pertinence* » est une conséquence de l'application de la méthode qui impose que les liens entre les éléments posés soient validés par les experts) au sein du réseau qui contient plusieurs dizaine de milliers d'éléments ?

5 Le retour d'expérience après la mise en service et la première phase de déploiement

5.1 Le calendrier de SARBACANES

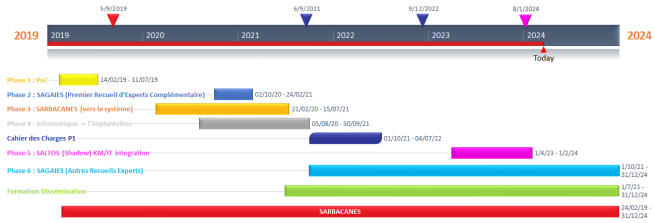


FIGURE 4 – Les éléments majeurs du calendrier de l'opération industrielle

La conduite de ce type d'opération de management de REX (cf. Figure 4) soulève des aspects de complexité initialement cachés. La réalité temporelle d'un projet basé sur de l'ingénierie de la connaissance et celle d'une appropriation à l'aune de la taille de l'organisation. La question de la confiance avec un système d'IA hybride s'est ici construite incrémentalement ainsi que celle de l'appropriation quasi immédiate des acteurs métiers adhérents à la démarche. La réussite de ce projet est aussi une implication forte de sponsor métier expérimenté et d'un moral à tout épreuve. La satisfaction enfin de constater qu'après la mise en exploitation, de nouvelles fonctionnalités "métier" connexes sont demandées puis intégrées à la base de connaissance initiale pour être exploitées très naturellement dans la vie courante.

5.2 Des éléments de modélisation métier de SARBACANES

Voici la méta-modélisation sous-jacente à la base de connaissance SARBACANES .

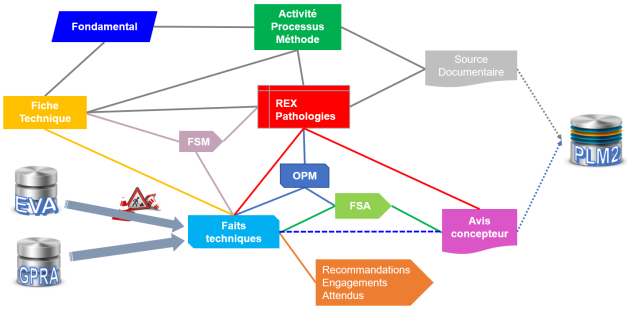


FIGURE 5 – La structuration des modèles et les différentes étapes de son déploiement

Si l'on prend en compte les premiers types de modèles pour analyser un « *Fait Technique* », quelles sont les connaissances manipulées ?

Nom du Modèle	Description : Exemple
Fondamental	Un principe ou phénomène : Ex. Corrosion des métaux
Activité Processus	Une action métier : Ex. Évolution du plan de maintenance
Fiche Technique	Un équipement ou une solution technique : Ex. Architecture du contrôle-commande
Source Documentaire	Une référence documentaire PLM : Ex. Procédure TA-6253301A
REX Pathologie	L'état de l'art sur un sujet : Ex. Phénomènes vibratoires
Fait Technique	Un événement d'exploitation : Ex. Alarme sur circuit AUGM24
Avis Concepteur	Diagnostic & Prescription sur une situation : Ex. Possibilité de déroger à AB.SB.TC02 pour l'IPER

Par exemple, lorsqu'un « *Fait Technique* » est annoncé, il concerne un équipement renseigné par une « *Fiche Tech-*

nique » au cours d'une « *Activité (Processus)* » le tout étant conçu sur la base de « *Fondamentaux* ». Les événements déjà instruits sur un sujet de même nature ont fait l'objet d'« *Avis Concepteur(s)* » et les analyses antérieures ont été consolidées dans des « *REX Pathologie* » référencé(s) comme « *Source documentaire* » dans des documents stockés dans l'outil de « *PLM* ».
Si la réponse n'est pas forcément la solution, avec SARBACANES les ingénieurs en charge de l'analyse du nouveau « *Fait Technique* », savent ainsi immédiatement « *positionner* » cette situation dans le patrimoine de TechnicAtome et débutent leur instruction en toute connaissance de cause.

5.3 La vie de SARBACANES

La connaissance s'agrège, se sédimente, s'incrémente au fil de l'eau et des retours d'expérience ou travaux nouveaux au plus grand plaisir des acteurs qui au quotidien voient leur travail prendre de la hauteur et un intérêt croissant.
Cette opération transcende l'équation de Davenport & Pruzak [7] car ici on observe que :

Knowledge Transfer =
Transmission + Absorption & Use + Enrichment

Il s'avère que le retour d'exploitation de SARBACANES confirme le fait que le système a non seulement un intérêt dans l'usage attendu dans la **production d'analyse temps court**, mais aussi dans le **transfert de connaissance** pour l'accompagnement des utilisateurs nouveaux dans le métier à s'approprier l'antériorité des analyses déjà produites, et dans l'enrichissement par la consolidation de l'expertise dans la **production d'analyse temps long**. Cette analyse temps long correspond à une réflexion *a posteriori* des acteurs : après la **Transmission**, l'**Appropriation** et l'**Usage**, il y a un véritable **Enrichissement** de la connaissance !

6 Perspectives

Très concrètement, le projet se développe progressivement par les apports pragmatiques qui simplifient le travail quotidien des utilisateurs. La fait d'avoir accroché les connaissances au process métier et le lien aux sources qui justifient la réponse à la question posée génère un cercle vertueux de confiance non pas simplement d'un utilisateur mais de toute l'équipe métier. Résultat, des processus de gestion métier sont basculés et déployés au travers de SARBACANES ce qui fait évoluer les pratiques et ravi les acteurs qui disposent de plus de temps pour se concentrer sur le cœur de leur métier.
On observe enfin que la rigueur du processus de validation de cette connaissance collective relève d'une véritable « *hygiène d'ingénierie* » pour l'organisation.
Une autre question est évoquée et est en cours d'analyse aujourd'hui : celle de l'apport effectif de l'IA générative (LLM+RAG) à ce stade. Faut-il risquer de perdre en qualité de performance et de confiance en introduisant un risque potentiel après avoir construit une base de connaissance robuste et riche forte de plusieurs dizaine de milliers d'actifs tangibles et validés humainement ?

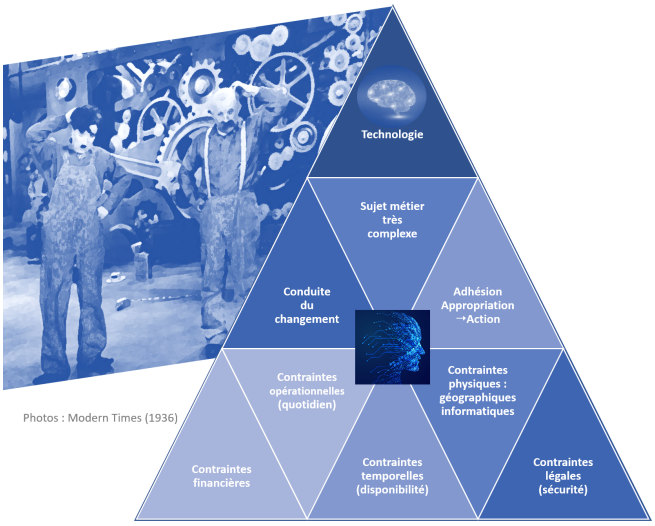


FIGURE 6 – Où positionner la technologie, la connaissance, la confiance & l'humain ?

Les derniers échanges avec la communauté HyCHA (cf. `https://hycha24.sciencesconf.org/`) [5] confirment l'aspect délicat comme la complexité technique et humaine dans l'implantation opérationnelle (cf. Figure 6) d'une technologie IA hybride dans le milieu industriel.

7 Conclusion

A la question du « *passage à l'échelle pour un système industriel, où positionner le point dur entre la technologie, la connaissance, la confiance & l'humain ?* » finalement la difficulté n'est pas celle que l'on attend !
L'approche hybride pour réaliser un tel système technique et opérationnel ne constitue pas *in fine* le point le plus difficile pour garantir le succès de ce type de projet ambitieux qui est d'abord une aventure humaine incroyable : ne dit-on pas que « *La prudence est mère de la sûreté* » ?
Ce qui reste certain, c'est que la démarche TA KM a permis de construire la base SARBACANES et le système de gestion de la connaissance du maintien en condition en service de TechnicAtome est fondé sur la base d'une somme de petits succès. La dissémination de cette méthode d'ingénierie et de son ancrage dans le métier est un processus lent qui est en tous les cas particulièrement prometteur.

8 Remerciements

Nous remercions vivement TechnicAtome d'avoir autorisé cette communication qui démontre tout l'intérêt d'une approche outillée de l'ingénierie de la connaissance appliquée aux retours d'expérience dans les métiers de l'ingénierie de la maintenance (et plus d'ailleurs), qui traitent en particulier ceux relatifs à la sûreté et la sécurité nucléaire.

Références

[1] Alain Berger. Évolution dans l'industrie du métier d'ingénieur cogniticien ou d'ingénieur de la connais-

sance entre 1985 et 2015. In *1st Conférence Nationale sur les Applications Pratiques de l'Intelligence Artificielle (APIA 2015) at the Plate-forme Intelligence Artificielle*, pages 23–33, Rennes, France, Juil 2015.

[2] Alain Berger. Regard sur l'ingénierie de la connaissance face à l'ISO30401. In *34es Journées francophones d'Ingénierie des Connaissances (IC 2023)*, volume https://hal.science/hal-04152777/file/PFIA2023IC_IC_%26_Iso30401_Alain_Berger.pdf, Strasbourg, France, July 2023. Plate-Forme Intelligence Artificielle (PFIA 2023) and AfIA & iCube.

[3] Alain Berger, François Vexler, Corentin Mary, and Jean-Pierre Cotton. Réflexion sur le choix d'un classifieur sémantique destiné à aider le cogniticien dans l'élaboration d'une base de connaissance et la garantie de sa consistance dans le temps. In *6ème Conférence Nationale sur les Applications Pratiques de l'Intelligence Artificielle, APIA 2020*, volume http://pfia2020.fr/wp-content/uploads/2020/08/Actes_CH_PFIA2020_V3.pdf, pages 66–73, Angers, France, 2-3 juillet 2020.

[4] Vincent Besson and Alain Berger. To initiate a corporate memory with a knowledge compendium : ten years of learning from experience with the ardans method. In *15èmes Journées Francophones Extraction et Gestion des Connaissances, EGC 2015, https://editions-rnti.fr/?inprocid=1002103*, volume RNTI-E-28, pages 401–412, Luxembourg, Luxembourg, 27-30 Janvier 2015. Hermann-Éditions.

[5] Sébastien Boblet, Thierry Cartié, Alain Berger, Jean-Pierre Cotton, and François Vexler. Mettre en place une approche hybride pour manager les avis techniques relatifs au retour d'expérience d'exploitation d'un équipement complexe sensible. In *HyCHA'24 - Journées d'Intelligence Artificielle Hybride : de l'intégration des connaissances et de l'humain à l'explication des modèles*, volume https://hycha24.sciencesconf.org/531283, Gif-sur-Yvette, France, 27-28 Mars 2024. CentraleSupelec.

[6] Thierry Cartié and Alain Berger. Construire avec confiance une base de connaissance ou savoir traiter un sujet technique complexe. In *6ème Forum Industriel de l'IA - IA DE CONFIANCE : Responsabilité, Robustesse, Transparence*, Le Totem, Paris, France, Octobre 2021. AfIA.

[7] Thomas Davenport and Laurence Prusak. *Working Knowledge : How Organizations Manage what They Know*, volume https://www.researchgate.net/publication/229099904_Working_Knowledge_How_Organizations_Manage_What_They_Know of *EBSCO eBook Collection*. Harvard Business School Press, 1998.

[8] Céline Fourtout, Patrick Prieur, Alain Berger, Jean-Pierre Cotton, Aline Belloni, and Daniel Marx. Épione : Formaliser un processus métier par une démarche d'ingénierie de la connaissance : retour d'expérience sur le déclassement dans le nucléaire. In *9èmes Conférence Nationale sur les Applications Pratiques de l'Intelligence Artificielle APIA2023, Strasbourg*, volume https://pfia23.icube.unistra.fr/conferences/apia, 6&7 Juillet 2023.

[9] Michel Grundstein. Développer un système à base de connaissance : un effort de coopération pour construire en commun un objet inconnu. In *Acte de la journée Innovation pour le travail en groupe*. CP2I, Novembre 1994.

[10] Pierre Malvache and Patrick Prieur. Mastering Corporate Experience with the REX Method, Management of Industrial and Corporate Memory. In *International Symposium on the Management of Industrial and Corporate Knowledge (ISMICK'93)*, pages pp.33–41, Compiègne, France, June 1993.

[11] Pierre Mariot, Christine Golbreich, Jean-Pierre Cotton, François Vexler, and Alain Berger. Méthode, modèle et outil ardans de capitalisation des connaissances. In *7èmes journées francophones Extraction et Gestion des Connaissances, EGC 2007*, volume https://editions-rnti.fr/render_pdf.php?p=1000709, pages 187–206, Namur, Belgique, 23-26 janvier 2008.

[12] ISO Central Secretary. Knowledge management systems — requirements iso30401 :2018. In *https://www.iso.org/standard/68683.html*, International Organization for Standardization. Geneva, CH, 2018.

[13] François Vexler, Corentin Mary, Alain Berger, and Jean-Pierre Cotton. Management des connaissances augmenté : usage d'un classifieur sémantique pour l'aide à l'élaboration et au maintien en cohérence d'une base de connaissance. In *20èmes Journées Francophones Extraction et Gestion des Connaissances, EGC 2020*, volume RNTI-E-36 * https://editions-rnti.fr/?inprocid=1002598, pages 393–400, Bruxelles, Belgique, 29-31 Janvier 2020.

Articles déjà publiés

HHT : Une ontologie pour représenter les dynamiques territoriales pour les humanités numériques

W. Charles[1], N. Hernandez[1,2]

[1] IRIT, CNRS, Université de Toulouse, Toulouse INP, Toulouse, France
[2] Université Toulouse 2 Jean Jaurès, Toulouse, France

prénom.nom@irit.fr

Résumé

La plupart des ontologies territoriales se concentrent sur les territoires actuels, s'appuient sur une représentation spatiale et n'ont pas l'intention d'englober l'impact des acteurs sur cet espace. Afin de représenter les territoires historiques, nous avons proposé l'ontologie HHT (Hierarchical Historical Territory). Elle englobe la description de territoires évolutifs, la représentation explicite de leurs changements et les revendications des acteurs sur ces territoires qu'elle représente sans avoir à connaître leur géométrie.

Mots-clés

Ontologie des territoires, représentation des changements, dynamiques d'acteurs, humanités numériques, territoires hiérarchiques

Abstract

Most territorial ontologies are focused on nowadays territories, rely on spatial representation and do not intend to encompass the impact of actors over said space. In order to represent historical territories, we proposed the HHT ontology (Hierarchical Historical Territory). It encompasses the description of evolving territories, explicit change representation, the claims of actors and allows to represent territories without having to know their geometry.

Keywords

territory ontology, change representation, actor dynamics, digital humanities, hierarchical territories

1 Introduction

Dans le contexte des humanités numériques, la représentation des territoires tels qu'ils étaient autrefois est un enjeu important, car il est nécessaire d'ancrer les faits dans une géographie contextualisée. La notion de *territoire* ne s'y réduit pas à une simple zone spatiale, qui pourrait être caractérisée par sa géométrie, mais englobe l'enchevêtrement d'une zone géographique et d'acteurs ayant une influence sur celle-ci. Les ontologies ont été utilisées dans les humanités numériques en raison de leur capacité à construire des modèles de représentation adaptés aux besoins des chercheurs en humanité et à favoriser la réutilisabilité et l'interopérabilité des connaissances qu'ils produisent [2]. Représenter correctement les territoires historiques implique de représenter à la fois les hiérarchies territoriales établies et les prétentions des acteurs à les modifier. En outre, la représentation géométrique des territoires historiques peut s'avérer difficile car elle est absente des données historiques disponibles. Cet article est un résumé étendu de celui publié à FOIS 2023 [4] présentant l'ontologie HHT (Historical Hierarchical Territories), développée dans le cadre du projet ObARDI [1] (Digital Humanities), qui vise à représenter à la fois les divisions territoriales et les dynamiques de pouvoir. La section 2 présente succintement l'ontologie HHT. La section 3 détaille le rôle des acteurs. Enfin, la section 4 rapporte brièvement quelques utilisations de cette ontologie qui valident les choix de conception.

2 L'ontologie HHT : territoires et temporalité

Les territoires historiques nécessitent une ontologie qui permette de représenter leurs multiples organisations hiérarchiquees superposées et leur évolution, sans connaître leur géométrie, et qui permette d'appréhender les différentes dynamiques de pouvoir qui les impactent. Les ontologies existantes (notamment TSN[1]) se concentrent généralement sur des territoires contemporains et ne permettent pas de prendre en considération l'ensemble de ces particularités. Pour ce faire, nous proposons l'ontologie HHT, divisée en trois modules : HHT (cœur de l'ontologie), HHT-Change pour décrire les changements et HHT-Claim pour représenter les revendications. Sur la figure 1, est représentée la hiérarchie territoriale au sein de HHT, qui constitue le cœur de l'ontologie. L'ontologie HHT se focalise sur les liens entre territoires, niveaux hiérarchiques et critères hiérarchiques, assurant ainsi une adaptabilité à divers contextes et autorisant la superposition de plusieurs hiérarchies. HHT propose également un mécanisme de raisonnement sur la géométrie sans représentation polygonale. Ce formalisme est décrit plus en détail dans [3]. HHT-Change propose une taxonomie de changements afin de matérialiser et qualifier les changements survenus entre deux versions de territoires. Elle distingue les changements individuels, n'affectant qu'un unique territoire, des changements composites,

1. https://obardi.hypotheses.org/

FIGURE 1 – Schema de l'ontologie HHT

aggrégeant de multiples changements individuels connexes pour leur attribuer un sens global.

3 Territoires, acteurs, revendications

En l'absence de spécificités sur la représentation des acteurs et institutions, l'Organization Ontology, recommandée par le W3C a été retenue pour les associer aux territoires dans HHT. Pour favoriser l'interopérabilité dans le contexte des humanités numériques, cette ontologie a été alignée avec CIDOC-CRM. De plus, afin de représenter l'interaction entre acteurs et territoires, des propriétés suivant une taxonomie minimale ont été ajoutées à HHT. L'interaction majeure entre territoires et acteurs se fait par le biais de revendications. Celles-ci traduisent la volonté des acteurs territoriaux d'opérer des changements. Cette représentation est l'objet d'HHT-Claim, dont la structure est donnée dans la figure 2. HHT-Claim propose deux façons complémentaires de décrire les revendications. La première s'appuie sur des sous-classes de Claim organisées en taxonomie et sur des propriétés génériques. Comme une revendication exprime une volonté de modifier une dimension du territoire, la propriété `hht:target` permet de définir l'objet (différent du territoire émetteur) qui est la finalité de la revendication. Un deuxième modèle de revendication permet de lier une revendication à un `hht:Change` qui décrira le changement que la revendication a l'intention de provoquer. Une distinction entre les changements effectifs et ceux faisant simplement l'objet de revendications est alors introduite.

4 Utilisation pratique

L'usage effectif de l'ontologie HHT met en exergue la problématique de la gestion des identités des territoires. Nous nous appuyons sur la notion de changement perturbateur utilisée dans [5], pour formuler un critère : *Une nouvelle entité est créée chaque fois que le nom change en même*

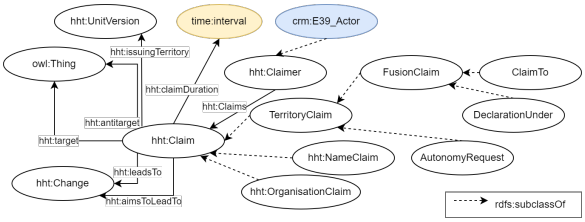


FIGURE 2 – Schéma du module ontologique HHT-Claim

temps qu'une autre propriété. Ce critère est indicatif, et les historiens peuvent aller à son encontre si cela leur semble pertinent. Par ailleurs, la construction incrémentale de la connaissance nécessite la mise en place d'un mécanisme de mise à jour des lignes de vies temporelles. En pratique, celle-ci s'effectue en refragmentant la ligne de vie des territoires pour la faire concorder avec l'état le plus récent des connaissances. Enfin, un algorithme permettant de détecter et qualifier les changements à partir de la ligne de vie d'un territoire a été développé [3].

5 Conclusion

L'ontologie HHT permet non seulement de représenter des territoires et des unités territoriales impliquées dans des hiérarchies sans avoir de représentation géométrique exacte, mais elle fournit également une représentation des acteurs attachés aux territoires et de leur volonté d'avoir un impact sur ceux-ci. En outre, elle comprend une représentation temporelle, qui permet de représenter le changement de manière implicite et explicite.

Références

[1] Camille Bernard, Marlène Villanova-Oliver, Jerome Gensel, and Hy Dao. Modeling changes in territorial partitions over time : Ontologies tsn and tsn-change. In *ACM Symposium on Applied Computing Pages*, page 866–875, 04 2018.

[2] George Bruseker, Nicola Carboni, and Anaïs Guillem. Cultural heritage data management : the role of formal ontology and cidoc crm. *Heritage and Archaeology in the Digital Age*, pages 93–131, 2017.

[3] William Charles, Nathalie Aussenac-Gilles, and Nathalie Hernandez. Hht : An approach for representing temporally-evolving historical territories. In *European Semantic Web Conference*, pages 419–435. Springer, 2023.

[4] William Charles and Nathalie Hernandez. Hht : An ontology to represent territorial dynamics for digital humanities. In *Formal Ontology in Information Systems*, pages 257–271. IOS Press, 2023.

[5] Pawel Garbacz, Bogumił Szady, and Agnieszka Ławrynowicz. Identity of historical localities in information systems. *Applied Ontology*, 16(1) :55–86, 2021.

Extraction d'informations à partir de rapports automobiles pour le peuplement d'ontologies

Hamid Ahaggach[1,3], Lylia Abrouk,[1,2], Eric Lebon [3]

[1] Laboratoire LIB , Université de Bourgogne, Dijon, France
prenom.nom@u-bourgogne.fr
[2] MISTEA, Université de Montpellier, INRAE & Institut Agro, France
[3] Syartec, Aix-en-Provence, France

Résumé

Dans cet article, nous présentons le travail publié dans la revue ***Applied Ontology*** *intitulé* ***Information extraction from automotive reports for ontology population*** [1] *(Février 2024). Ce travail met en lumière nos recherches dédiées à l'utilisation des ontologies et à l'extraction d'informations (EI), dans le but d'analyser et de modéliser les dommages subis par les carrosseries de voitures. Notre approche consiste à analyser les rapports non structurés en appliquant des techniques de traitement automatique du langage, telles que la reconnaissance d'entités nommées (REN) et l'extraction de relations (ER), afin d'identifier et d'extraire les informations pertinentes des rapports.*

Mots-clés

Extraction d'informations, Ontologie, Reconnaissance d'entités nommées, Extraction de relations.

Abstract

In this article, we present the paper published in the journal ***Applied Ontology*** *titled* ***Information extraction from automotive reports for ontology population*** *[1] (February 2024). It highlights our research work dedicated to the use of ontologies and information extraction, aimed at analyzing and modeling the damages incurred by car bodies. Our approach analyzes unstructured reports using natural language processing techniques, such as REN and RE, to identify and extract relevant information from the reports.*

Keywords

Information extraction, Ontology, Named entity recognition, Relationship extraction.

1 Introduction

Dans le secteur automobile, la gestion du transport des voitures est une tâche complexe. À l'arrivée de chaque voiture, un contrôle de qualité est effectué pour identifier les dommages subis pendant le transport, incluant la prise de photographies et la rédaction de rapports. Cependant, ces rapports de dommages sont non structurés et ne suivent pas de standards uniformisés pour la description des dommages. Ceci entraîne un processus de saisie manuelle des données chronophage et susceptible d'erreurs. Pour cela, nous avons développé *OCD* (Ontology for Car Damage), une ontologie conçue pour la modélisation des dommages des voitures. Cette ontologie offre un cadre structuré permettant de décrire et de catégoriser les différents types de dommages de manière précise et uniforme. De plus, l'ontologie contribue à l'amélioration du système d'extraction d'informations que nous avons proposé. Ce système est conçu pour extraire les informations pertinentes des rapports automobiles non structurés afin de peupler notre ontologie. Notre approche facilite le processus de contrôle de qualité pour le transport des véhicules et offre une méthode standardisée pour documenter et catégoriser les dommages des voitures.

1. The accepted paper.

2 Travaux antérieurs

2.1 Ontologie

Dans le domaine de l'automobile, les ontologies ont été utilisées pour modéliser les accidents de la route en décrivant les circonstances, le lieu, les causes et les effets de l'accident. Cependant, l'accent mis sur la modélisation des dommages subis par le véhicule a été insuffisant. Le tableau 1 offre une comparaison complète des différentes ontologies automobiles selon plusieurs critères.

TABLE 1 – Comparaison des ontologies automobiles.

Critères/Travaux	[1]	[2]	[3]	[4]	[5]
Modélisation des dommages de voiture	✓	×	×	×	×
Modélisation des informations de voiture	✓	✓	✓	✓	✓
Modélisation des pièces	✓	×	✓	×	✓
Support multilingue	✓	×	×	×	×
Accès public	✓	×	×	✓	✓
Capacités d'inférence	✓	×	✓	×	×

2.2 Extraction d'information

Ces dernières années, plusieurs chercheurs se sont intéressés à l'extraction d'informations dans le domaine de l'automobile, en raison de son potentiel à améliorer divers pro-

cessus dans l'industrie. De nombreuses études se sont focalisées sur l'extraction d'informations telles que les marques et modèles de véhicules. Ces stratégies mises en œuvre demeurent conventionnelles, se basant principalement sur des systèmes de règles. Ces approches sont limitées, car elles ne parviennent pas à s'adapter aux variations du langage naturel et du contexte, ce qui affecte la précision et la fiabilité des données extraites. De plus, elles ne permettent pas d'extraire les relations entre les entités, élément essentiel pour une compréhension approfondie de la sémantique des informations.

3 Approche

Notre approche se compose de deux étapes principales : la construction de l'ontologie et l'extraction d'informations. Dans l'étape de construction de l'ontologie, nous définissons les concepts et les relations pour représenter le domaine de l'évaluation des dommages des voitures, L'ontologie est disponible sur *GitHub*[2] et *industryportal*[3]. Dans l'étape d'extraction d'informations (Figure 1), nous extrayons des informations (Reconnaissance d'Entités Nommées REN et Extraction de Relations ER) à partir du texte. Une fois les relations extraites, nous améliorons le résultat de cette extraction en utilisant le raisonnement de l'ontologie. Ceci permet de réduire les redondances en gérant les conflits et en minimisant les faux positifs et les faux négatifs dans les relations extraites. Nous peuplons l'ontologie en associant les entités et relations extraites aux concepts et propriétés de notre ontologie.

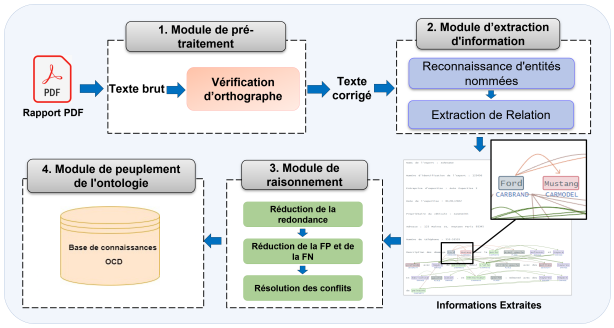


FIGURE 1 – Méthodologie générale

4 Résultats et Conclusion

Pour évaluer l'efficacité de notre approche, nous avons utilisé un ensemble de données comprenant des rapports automobiles décrivant les dommages aux voitures, fournis par l'entreprise *Syartec*[4]. L'ensemble de données a été prétraité et étiqueté à l'aide de l'outil *doccano* pour l'entraînement et le test de nos modèles REN et ER. Notre objectif est d'extraire des entités à partir de rapports de dommages automobiles. Plus précisément, nous nous sommes concentrés sur l'extraction de six types d'entités : $CarBrand$, $CarModel$, $Carparts$, $Damage$, $Severity$ et $Place$. Nous avons exclu les autres informations des rapports, car elles étaient déjà structurées et il n'était pas nécessaire de les extraire. Les résultats ont montré que le modèle $SpaCy$ est performant pour la plupart des entités, tandis que le modèle CRF est performant avec les entités spécifiques au domaine, et le modèle $BiLSTM - CRF$ est performant pour les entités composées. Une fois les entités extraites, nous avons utilisé des algorithmes d'apprentissage automatique pour identifier les relations entre elles. Nous avons cherché à extraire quatre types de relations : $hasDamage$, $hasCarParts$, $CarBrand$ et $Carparts$. Pour créer nos modèles d'extraction de relations, nous utilisons quatre algorithmes de classification utilisés pour l'extraction de relations : les machines à vecteurs de support, les k-plus proches voisins, les arbres de décision et les forêts aléatoires. En combinant le raisonnement de l'ontologie OCD, nous avons réussi à extraire des informations pertinentes de rapports automobiles complexes, particulièrement dans les scénarios où un seul événement de dommage est associé à plusieurs parties de la voiture. Nous avons créé l'ontologie OCD en utilisant *Protégé 5.5.0*[5] et l'avons peuplée avec les informations extraites en utilisant la bibliothèque $Owlready$.

Ces travaux ouvrent la voie à de nouvelles questions liées à la prédiction des coûts de réparation des véhicules. Nous avons initié une approche visant à répondre à cette problématique en proposant une approche hybride. En intégrant des règles $SWRL$ dans notre ontologie, nous pouvons identifier les composants réutilisables, réduisant ainsi la nécessité d'acheter de nouvelles pièces et, par conséquent, minimisant les coûts de réparation.

Références

[1] Ahaggach, H., Abrouk, L., & Lebon, E. (2024). Information extraction from automotive reports for ontology population. Applied Ontology, 1-30.

[2] Barrachina, J., Garrido, P., Fogue, M., Martinez, F. J., Cano, J. C., Calafate, C. T., & Manzoni, P. (2012, April). Caova : A car accident ontology for vanets. In 2012 IEEE wireless communications and networking conference (WCNC) (pp. 1864-1869). Ieee.

[3] Feld, M., & Müller, C. (2011, November). The automotive ontology : managing knowledge inside the vehicle and sharing it between cars. In Proceedings of the 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications (pp. 79-86).

[4] Hepp, M. (2010). Vehicle sales ontology. Retrieved from http ://www.heppnetz.de/ontologies/vso/ns

[5] Klotz, B., Troncy, R., Wilms, D., & Bonnet, C. (2018, October). VSSo : The Vehicle Signal and Attribute Ontology. In SSN@ ISWC (pp. 56-63).

2. github.com/OntologyCarDamage/OCD
3. industryportal.enit.fr/ontologies/OCD
4. www.syartec.com
5. `https://protege.stanford.edu`

Aligner les descriptions des plantes ayant des points de vue distincts

F. Amardeilh[1], S. Aubin[2], S. Bernard[3], S. Bravo[2], R. Bossy [4], C. Faron[5], F. Michel[5], C. Roussey[6]

[1] Elzeard, Bordeaux, France
[2] DIPSO, INRAE, France
[3] LISC, INRAE, Aubière, France
[4] MaIAGE, INRAE, Jouy-en-Josas, France
[5] Université Côte d'Azur, INRIA, CNRS, I3S, France
[6] MISTEA, INRAE, Montpellier, France

florence.amardeilh@elzeard.co, sophie.aubin@inrae.fr, stephan.bernard@inrae.fr,
robert.bossy@inare.fr, faron@i3s.unice.fr, fmichel@i3s.unice.fr, catherine.roussey@inrae.fr

1 Contexte

Nous présentons nos travaux sur l'alignement de deux graphes de connaissances complémentaires utiles dans le domaine agricole : le thésaurus des Usages des plantes Cultivées en France (FCU) et le REFérentiel TAXonomique national français TAXREF pour la faune, la flore et les champignons déjà publié dans [1]. FCU décrit l'utilisation des plantes en agriculture, ou plus exactement le rôle des plantes en agriculture : par exemple les '*tomates*' sont des cultures utilisées pour l'alimentation humaine. Il représente le point de vue des agriculteurs. TAXREF décrit les taxons biologiques et les noms scientifiques associés, ou plus exactement une description de cette plante suivant sa composition génétique : par exemple, une espèce de tomate peut être '*Solanum lycopersicum*'. TAXREF est maintenu par le Muséum national d'Histoire naturelle (MNHN) et représente le point de vue des taxonomistes et des agronomes. Les deux graphes de connaissances contiennent des noms vernaculaires de plantes. Les noms vernaculaires sont souvent ambigus et peu consensuels, ce qui rend l'activité d'alignement particulièrement difficile.

2 Travaux antérieurs

Nos travaux précédents [4] ont implémenté plusieurs méthodes d'alignement automatique basées sur la comparaison de noms vernaculaires. Ces méthodes automatiques ont réutilisé des sources de référence existantes comme la base de données européenne de l'EPPO [1] et le catalogue officiel français des espèces et variétés de plantes cultivées du GEVES [2]. Les résultats montrent qu'il est nécessaire de nettoyer les alignements produits automatiquement en raison de l'ambiguïté des noms vernaculaires. Par conséquent, un groupe d'experts agricoles a produit des alignements valides.

1. https ://gd.eppo.int/
2. Catalogue officiel des esp 'eces et variétés de plantes cultivéesées en France accessible sur https ://www.geves.fr/catalogue/

3 Matériels

3.1 TAXREF et TAXREF-LD

TAXREF [3] est le référentiel taxonomique français de la faune, de la flore et des champignons. TAXREF est disponible sous la forme d'un graphe de connaissances respectant les principes des données liées, nommé TAXREF-LD [5]. TAXREF-LD est disponible sur AgroPortal [3]. Ce travail a été développé en utilisant la version 15.2 de TAXREF-LD qui contient 287 229 classes et plus de 1 000 000 instances.

3.2 Thésaurus FCU

Le thésaurus des Usages des plantes Cultivées en France normalise les noms de cultures en français. De plus, il organise ces noms de cultures en catégories selon leurs usages sur le territoire français. Les usages représentent également les secteurs agricoles. Le thésaurus est publié sur le Web selon les principes des données liées. Le thésaurus est disponible sur AgroPortal [4]. Ce travail a été développé en utilisant la version 3.3 de FCU qui contient 707 instances de `skos:Concept`.

3.3 Propriétés d'alignement

Dans TAXREF-LD, un taxon est défini par une classe `owl:Class` et les noms de taxon sont définis par des instances de `skos:Concept`. Une classe regroupe l'ensemble des spécimens caractérisant actuellement le taxon considéré. Ces spécimens sont généralement stockés par les muséums d'histoire naturelle ou par les centres de ressources génétiques. Les noms scientifiques découlent des avancées de la taxonomie : la science de classer et nommer les êtres vivants. Les méthodes de classification, de création des taxons, ont évoluées avec le temps. Ainsi, les noms scientifiques d'espèces et leurs synonymes sont une trace de ces évolutions successives des taxons. Dans FCU, un rôle de plante cultivée est défini

3. https ://agroportal.lirmm.fr/ontologies/TAXREF-LD
4. https ://agroportal.lirmm.fr/ontologies/CROPUSAGE

par une instance de `skos:Concept`. Un concept FCU a pour label un nom vernaculaire enrichi de son rôle : par exemple le concept "carotte potagère" représente les carottes consommées pour l'alimentation humaine. Nous avons défini 12 propriétés d'objet pour lier une instance de `skos:Concept` représentant un nom scientifique à une instance de `skos:Concept` de FCU représentant un rôle de plante en agriculture. Ces propriétés permettent de répondre à la question de compétence : "quel nom scientifique (principalement de rang espèce) peut jouer ce rôle en agriculture ?". Sachant qu'il est possible qu'un nom scientifique d'espèce ne réponde pas entièrement au rôle, et inversement. Ces propriétés d'objets peuvent être vu comme une spécialisation de la propriété `skos:closeMatch` : un nom scientifique identifiant une espèce n'est pas équivalent à un rôle agricole mais il est proche. Par exemple, toutes les espèces de carottes cultivées ont potentiellement deux rôles en agriculture : carottes potagères (alimentation humaine) et carottes fourragères (alimentation animale). Ce qui n'est pas le cas pour la chicorée qui a des espèces différentes remplissant l'un des deux usages agricoles : chicorées potagères et chicorées industrielles. Pour compléter, nous avons aussi défini 10 propriétés d'annotations pour lier une `owl:Class` représentant un taxon à une instance de `skos:Concept` de FCU représentant un rôle de plante en agriculture. Ainsi, les triplets utilisant les propriétés d'objet sont documentés (annotés) par les triplets utilisant les propriétés d'annotation.
Il n'existe pas de règles d'alignement entre les espèces (noms scientifiques) et les rôles agricoles, ce qui explique le besoin d'aligner manuellement ces différentes entités. L'ensemble de ces nouvelles propriétés sont disponible dans une ontologie French Crop Usage Ontology disponible sur AgroPortal [5].

3.4 Schéma d'alignement SSSOM

"Simple Standard for Sharing Ontology Mappings" (SSSOM) est un standard récent, développé par la communauté biomédicale autour de OBO Foundry et décrit dans [2]. Pour ce travail, nous avons utilisé la version 0.11.O de SSSOM sortie en mars 2023.

4 Méthode d'alignement manuelle

Suite aux résultats non satisfaisants des méthodes automatiques, nous avons créé un nouvel ensemble d'alignements en demandant à des experts de proposer des alignements corrects et réputés entre des rôles de plantes en agriculture, les taxons et leurs noms scientifiques. Ainsi, tous les alignements proposés doivent avoir une valeur de confiance élevée. En cas d'ambiguïté, l'alignement ne doit pas être créé. Dans un premier temps, nous avons fourni aux experts des consignes pour les aider dans leurs décisions. Deuxièmement, quatres outils de recherche ont été proposés pour rechercher des termes dans les deux graphes de connaissances : AgroPortal, deux SPARQL end points et un outil de recherche par facettes. Troisièmement, trois règles de conservation ont été écrites pour contextualiser les alignements créés et indiquer leur provenance. Nous nous sommes concentrés sur des cultures spécifiques : vigne, carotte, chicorée et tomate en fonction de la disponibilité des experts.

5 Résultats

Les alignements réalisés par les experts en suivant la méthode précédente sont publiés en données ouvertes sur le référentiel français Recherche Data Gouv [6]. Ainsi, ils pourraient être utilisés comme base de référence pour valider toute méthode d'alignement automatique.

Références

[1] Florence Amardeilh, Sophie Aubin, Stephan Bernard, Catherine Faron, Sonia Bravo, Robert Bossy, Franck Michel, Juliette Raphel, and Catherine Roussey. Combining different points of view on plant descriptions : mapping agricultural plant roles and biological taxa. *Frontiers in Artificial Intelligence*, 6 :118803, 2023.

[2] Matentzoglu et al. A Simple Standard for Sharing Ontological Mappings (SSSOM). *Database*, 2022, 05 2022. baac035.

[3] Olivier Gargominy, Sandrine Tercerie, C Régnier, T Ramage, P Dupont, P Daszkiewicz, and L Poncet. TAXREF v15, référentiel taxonomique pour la France : méthodologie, mise en œuvre et diffusion. Technical report, 2021.

[4] Franck Michel, Florence Amardeilh, Robert Bossy, Catherine Faron, Catherine Roussey, and Camille Noûs. Alignement entre sources : cas d'usage des plantes cultivées. In *Journées francophones d'Ingénierie des Connaissances*, Saint-Étienne, France, June 2022.

[5] Franck Michel, Olivier Gargominy, Sandrine Tercerie, and Catherine Faron-Zucker. A Model to Represent Nomenclatural and Taxonomic Information as Linked Data. Application to the French Taxonomic Register, TAXREF. In *Proceedings of the ISWC2017 workshop on Semantics for Biodiversity (S4BioDiv)*, volume 1933, Vienna, Austria, 2017. CEUR Workshop Proceedings. http ://ceur-ws.org/Vol-1933/paper-3.pdf.

Remerciements

Nous tenons à remercier les experts agronomes : Thierry Lacombe (orcid :0000-0001-9968-8228), Olivier Yobregat (orcid :0000-0002-7516-8727) et Juliette Raphel (orcid :0000-0002-5872-5034). Ces travaux ont été financé par projet ANR Data to Knowledge in Agronomy and Biodiversity (ANR-18-CE23-0017) et par le Plan de Relance et le Programme d'Investissements d'Avenir «i-Nov» du gouvernement français.

5. https ://agroportal.lirmm.fr/ontologies/FCUO

6. https ://doi.org/10.57745/LVRFWJ

PyGraft: un outil Python pour la génération de schémas et graphes de connaissance synthétiques

Nicolas Hubert[1,2], Pierre Monnin[3], Mathieu d'Aquin[2], Davy Monticolo[1], Armelle Brun[2]

[1] Université de Lorraine, ERPI, Nancy, France
[2] Université de Lorraine, CNRS, LORIA, Nancy, France
[3] Université Côte d'Azur, Inria, CNRS, I3S, Sophia-Antipolis, France

prenom.nom@univ-lorraine.fr

Résumé

Des graphes de connaissances (GCs) se sont imposés comme des benchmarks standards pour certaines tâches. Cependant, l'utilisation d'un nombre limité de jeux de données ne permet pas d'évaluer la généralité d'une approche. Nous proposons PyGraft, un outil Python qui génère des schémas et GCs agnostiques et hautement personnalisables. PyGraft rend possible la génération d'un éventail plus diversifié de ressources pour permettre une évaluation plus holistique. PyGraft est disponible en libre accès : `https://github.com/nicolas-hbt/pygraft`.

Mots-clés

Graphe de connaissance, schéma, Web sémantique, générateur synthétique

Abstract

A few knowledge graphs (KGs) have become standard benchmarks for some tasks. However, relying on a limited collection of datasets is insufficient to assess the generality of an approach. We introduce PyGraft, a Python-based tool that generates agnostic and highly customizable schemas and KGs. The aim of PyGraft is to encourage the generation of a more diverse array of resources to allow for a more holistic evaluation. PyGraft is available at : `https://github.com/nicolas-hbt/pygraft`.

Keywords

Knowledge graph, schema, semantic Web, synthetic data generator

1 Introduction

Cet article a été accepté à ESWC 2024 [4].

Les graphes de connaissances (GCs) sont de plus en plus utilisés comme structure de représentation des données en forme de graphe. Plus spécifiquement, un GC est une collection de triplets (s, p, o) où s (sujet) et o (objet) sont deux entités du graphe, et p est un prédicat qui qualifie la nature de la relation les liant [3].

Les GCs sont utilisés dans un large éventail de tâches, pour beaucoup desquelles une collection limitée de GCs s'est établie comme jeux de données standards pour évaluer la performance des modèles. Cependant, dans de nombreuses tâches telles que la classification de noeuds, les jeux de données standards exhibent des caractéristiques similaires, ce qui limite la possibilité d'une évaluation holistique d'un modèle ou d'une approche [6].

Il convient également de souligner le nombre limité de GCs publiquement disponibles dans certains domaines d'application à fort enjeu tels que l'éducation ou la médecine. Dans de tels cas, il devient difficile d'évaluer la qualité d'une approche en l'absence de jeux de données publiquement accessibles. Il faut alors se tourner vers des GCs privés – ce qui n'est pas toujours possible – ou des GCs généralistes – ce qui limite la pertinence et la portée de l'évaluation.

Les problèmes susmentionnés ont conduit à plusieurs tentatives de construction de générateurs synthétiques de schémas et de GCs [5, 7], même si dans la plupart des cas ces deux aspects (génération de schémas *vs.* génération de GCs) ont été considérés séparément [2, 5].

Dans ce travail, nous présentons PyGraft. Contrairement aux approches existantes, PyGraft génère à la fois des schémas et GCs synthétiques et agnostiques à tout domaine.

2 Approche

Les différentes spécifications et étapes dans la génération de schémas et de GCs sont représentées en Figure 1. Il convient de noter que PyGraft permet la génération d'un schéma, d'un GC, ou des deux à la fois. Nous nous focalisons sur le dernier cas pour illustrer notre propos.

L'utilisateur spécifie les paramètres souhaités pour la génération du schéma et du GC (partie gauche de la Figure 1). Un grand nombre de constructions `OWL` et `RDFS` sont permises. Dans un premier temps, PyGraft instancie puis appelle le `Class Generator`, lequel génère un ensemble de classes et de constructions associées, telles que `rdfs:subClassOf` et `rdfs:disjointWith`. Ensuite, ces informations de classes sont passées au `Relation Generator` qui, sur la base des contraintes établies par le `Class Generator` et les spécifications présentes dans les paramètres de schéma, va générer un ensemble de relations et de propriétés associées, telles que `owl:ReflexiveProperty` ou

`owl:SymmetricProperty`. Le schéma est alors établi. La consistance sémantique de ce dernier est validée via un raisonneur sémantique – dans notre cas HermiT [1]. Enfin, sur la base du schéma généré et validé ainsi que des paramètres utilisateur, le GC est généré sur la base d'autres paramètres tels que le nombre d'entités (`num_entities`), de triplets (`num_triples`), la proportion d'entités typées (`prop_untyped`), le nombre moyen de classes par entités (`avg_multityping`), etc. Enfin, la consistance sémantique du GC est elle aussi vérifiée (partie droite de la Figure 1).
De plus amples informations sur les constructions `OWL` et `RDFS` et les divers paramètres acceptés (p.e., profondeur de la hiérarchie des classes, domaines et co-domaines des relations, etc.) sont répertoriées dans la documentation officielle [1].

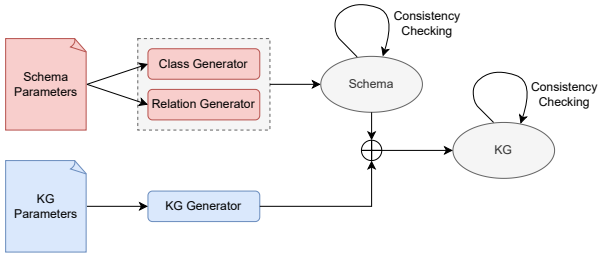


FIGURE 1 – Vue d'ensemble de PyGraft

3 Evaluation

Nous effectuons une évaluation exhaustive des capacités de PyGraft, tant sur le plan de la validité sémantique des ressources produites que de leur temps de génération. En particulier, nous établissons 27 combinaisons différentes de paramètres pour la génération de schémas et de GC : 9 spécifications de schémas et 3 spécifications de GCs avec différents niveaux de complexité et de taille.
La consistance sémantique des 27 combinaisons de ressources générées (schéma et GC) a été systématiquement validée dès la première tentative de génération. Ce résultat valide la capacité de PyGraft à générer des ressources dont la logique interne est vérifiée. Une analyse plus approfondie du temps de génération par combinaison de schéma et GC est présentée dans l'article original [4]. En l'essence, on observe que PyGraft permet de générer des GCs consistants très rapidement, même pour des tailles de GCs semblables aux benchmarks traditionnellement utilisés.

4 Conclusion et travaux futurs

PyGraft est un outil Python pour générer des schémas et GCs synthétiques intégrant un large évantail de constructions `OWL` et `RDFS`. PyGraft s'avère utile dans de nombreux scénarios : l'évaluation de nouvelles approches sur un panel plus large de jeux de données, la création de ressources agnostiques dans des domaines à fort enjeu, ou le développement de modèles neuro-symboliques tirant parti d'axiomes ontologiques.
Nos futurs directions de recherche concernent l'optimisation du processus de génération des GCs, afin d'en générer de plus grands et en un temps d'exécution réduit. Nous souhaitons aussi enrichir l'évantail des constructions `OWL` et `RDFS` modélisées. Enfin, nous chercherons à rendre possible la génération de littéraux, qui ne sont actuellement pas pris en charge.

1. `https://pygraft.readthedocs.io/en/latest`

Références

[1] Birte Glimm, Ian Horrocks, Boris Motik, Giorgos Stoilos, and Zhe Wang. Hermit : An OWL 2 reasoner. *J. Autom. Reason.*, 53(3) :245–269, 2014.

[2] Yuanbo Guo, Zhengxiang Pan, and Jeff Heflin. LUBM : A benchmark for OWL knowledge base systems. *J. Web Semant.*, 3(2-3) :158–182, 2005.

[3] Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard de Melo, Claudio Gutierrez, Sabrina Kirrane, José Emilio Labra Gayo, Roberto Navigli, Sebastian Neumaier, Axel-Cyrille Ngonga Ngomo, Axel Polleres, Sabbir M. Rashid, Anisa Rula, Lukas Schmelzeisen, Juan Sequeda, Steffen Staab, and Antoine Zimmermann. *Knowledge Graphs*. Synthesis Lectures on Data, Semantics, and Knowledge. Morgan & Claypool Publishers, 2021.

[4] Nicolas Hubert, Pierre Monnin, Mathieu d'Aquin, Armelle Brun, and Davy Monticolo. Pygraft : Configurable generation of synthetic schemas and knowledge graphs at your fingertips. In *The Semantic Web - 21st International Conference, ESWC 2024, Proceedings*.

[5] André Melo and Heiko Paulheim. Synthesizing knowledge graphs for link and type prediction benchmarking. In *The Semantic Web - 14th International Conference, ESWC 2017, Portorož, Slovenia, May 28 - June 1, 2017, Proceedings, Part I*, volume 10249 of *Lecture Notes in Computer Science*, pages 136–151, 2017.

[6] John Palowitch, Anton Tsitsulin, Brandon Mayer, and Bryan Perozzi. Graphworld : Fake graphs bring real insights for gnns. In *KDD '22 : The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 14 - 18, 2022*, pages 3691–3701. ACM, 2022.

[7] Jan Portisch and Heiko Paulheim. The DLCC node classification benchmark for analyzing knowledge graph embeddings. In *The Semantic Web - ISWC 2022 - 21st International Semantic Web Conference, Virtual Event, October 23-27, 2022, Proceedings*, volume 13489 of *Lecture Notes in Computer Science*, pages 592–609. Springer, 2022.

C3PO : Une ontologie pour la planification de cultures et les processus de production agricole

B. Darnala[1,2], F. Amardeilh[2], C. Roussey[3], K. Todorov[1], C. Jonquet[1,3]

[1] LIRMM, University of Montpellier, CNRS, Montpellier, France
[2] Elzeard, Villenave-d'Ornon, France
[3] MISTEA, University of Montpellier, INRAE, Institut Agro, Montpellier, France

21 mai 2024

1 Introduction

Le maraîchage est un métier complexe qui repose sur plusieurs facteurs comme les cycles de vie des cultures, la météo, ou les besoins commerciaux. Les maraîchers cherchent en partie à diversifier leurs types de production. Des études [1, 5] ont montré que la diversification à la fois spatiale et temporelle améliore la défense naturelle des plantes, prévient les risques relatifs au changement climatique et économique, et améliore la stabilité et la résilience des agroécosystèmes. Pour mener à bien cette diversification et faire les bons choix de planification, les maraîchers planifient leurs cultures en prenant en compte à la fois leurs connaissances agronomiques, mais aussi l'expérience de leurs précédentes cultures. Les maraîchers s'appuient aussi sur l'utilisation d'itinéraire technique de culture (ITK), c'est-à-dire les tâches à réaliser pour mener à bien une culture. Nous avons construit une ontologie appelée *Crop Planning and Production Process Ontology* [2]. Cette ontologie représente la planification maraîchère avec la disposition spatiale et temporelle des cultures, les ITKs et la connaissance sur les plantes nécessaires à l'agriculteur pour pouvoir planifier. Cette ontologie a servi de base à la construction d'un graphe de connaissance utilisé par plusieurs applications.

2 État de l'art

Plusieurs ontologies et graphes de connaissance existent pour formaliser les connaissances sur les plantes et sur l'agriculture. Certaines ressources décrivent une partie de la connaissance agronomique comme les graphes de connaissance TAXREF-LD [6] et NCBI taxonomy [4] pour la taxonomie botanique, French Crop Usage (FCU) [10] pour la représentation des familles d'usages agricoles et la Crop Ontology [1] pour la représentation des caractéristiques observables des cultures. D'autres ressources représentent les cultures sur les parcelles comme Agronomy Ontology [3] pour enregistrer les expériences et les observations et le modèle DEMETER [7] pour les cultures et les informations obtenus grâce à des capteurs. Chaque ressource a un point de vue particulier mais aucune ne permet de représenter les spécificités de la planification maraîchère. Cependant, plusieurs ressources ont été liées au graphe de connaissance issu de l'ontologie C3PO.

3 Méthodologie de construction d'une ontologie et d'un graphe de connaissance

L'ontologie et le graphe de connaissance ont été produits dans le cadre du développement de plusieurs applications métiers. Pour être en accord avec les processus de développement de ces applications, nous avons implémentés deux méthodologies agiles. La méthodologie LOT [9] décrivant les processus de construction d'une ontologie : la spécification des besoins, la conceptualisation, l'implémentation, la publication, la construction du graphe de connaissance et la maintenance. La phase de conceptualisation a été enrichie en suivant la méthodologie SAMOD [8] apportant un cadre pour la documentation des cas d'usages et des définitions dans un processus agile.

4 C3PO : Crop Planning and Production Process Ontology

C3PO est une ontologie composée de plusieurs sous-ontologies appelées modules : 6 modules de domaine pour la connaissance métier et 3 modules de support pour favoriser l'interopérabilité et la réutilisation.

Les modules de support sont les modules Time, Vocabulary et Parameter. Time étend la Time ontology pour représenter des intervalles relatifs de temps qui ne sont pas rattachés à une année spécifiquement, nécessaires pour représenter la connaissance sur les ITKs. Un exemple est un intervalle du 2 mai au 8 juin. Vocabulary est un thésaurus SKOS composés d'instances appartenant à des listes fermées comme les unités ou les types de climats. Parameter représente les paramètres numériques comme le poids ou le volume.

Les modules de domaines sont Plant, Plot, Crop Management, Supply, Admin, Sale. Le module Plant est une taxonomie des plantes basés sur le point de vue d'un maraîcher avec des informations agronomiques sur les plantes. Le module Plot représente la ferme et le découpage parcellaire. Le module Crop Management qui représente les ITKs à trois échelles : une échelle théorique pour le partage de connaissance, une échelle planifiée pour préparer ses productions et une échelle réalisée pour analyser et adapter ses futures

cultures. Le module Admin représente les personnes et les organisations. Le module Supply représente les intrants et les équipements agricoles. Enfin, le module Sale représente la vente et distribution des productions.

Un graphe de connaissance a été construit sur la base de cette ontologie et a été instancié à l'aide de plusieurs sources de données hétérogènes. La connaissance des plantes et des ITKs provient de données recueillies auprès d'experts du domaine. Les graphes de connaissance TAXREF-LD et FCU ont été liés manuellement aux instances du graphe de connaissance de C3PO. Une base de données des produits phytosanitaires a été transformée et ajoutée pour fournir la connaissance sur les intrants. Enfin, les données utilisateurs issues de plusieurs applications (Elzeard, Pépinière, Serre des Savoirs) ont été recueillies dans le graphe de connaissance.

L'ontologie C3PO a pour URL `http://www.elzeard.co/ontologies/c3po` et est actuellement publiée sur GitLab et AgroPortal [1] [2]. Une sous-partie du graphe de connaissance contenant les données sur les plantes et les ITKs est accessible sur GitLab et à travers un endpoint SPARQL [3] [4]. L'ontologie et cette sous-partie du graphe de connaissance sont distribuées sous la licence Creative Commons Attribution 4.0 International license (CC-BY 4.0).

5 Conclusion

La planification des cultures maraîchères est complexe et demande d'appréhender plusieurs types d'informations pour pouvoir prendre des décisions. Pour répondre à cette problématique, nous proposons l'ontologie C3PO et ses modules pour représenter les multiples domaines relatifs à la planification maraîchère. L'ontologie a servi de base à la construction d'un graphe de connaissance composé d'informations importantes pour aider les agriculteurs dans leurs prises décisions et le graphe recueille les données de plusieurs applications métiers (Elzeard, Pépinière, Serre des Savoirs).

Remerciements

Nous reconnaissons le soutien de l'Office National de la Biodiversité avec la subvention MesclunDurab et de BPI avec le financement I-NOV. Ce travail a également été partiellement réalisé avec le soutien du projet Data to Knowledge in Agronomy and Biodiversity (D2KAB - www.d2kab.org) qui a reçu un financement de l'Agence Nationale de la Recherche (ANR-18-CE23-0017) et du projet "Partages de Connaissances" (PACON) du programme transversal MetaBio financé par l'INRAE. Nous remercions Kevin Morel (INRAE), Juliette Raphel (Elzeard) et Guillaume Turlier (Elzeard) pour leur aide en tant qu'experts du domaine et tous les contributeurs des projets MesclunDurab et D2KAB pour leurs commentaires constructifs.

1. `https://gitlab.com/serre-des-savoirs/c3po`
2. `https://agroportal.lirmm.fr/ontologies/C3PO`
3. `https://gitlab.com/serre-des-savoirs/c3po-kb`
4. `https://graph.elzeard.co/sparql`

Références

[1] E. Arnaud, L. Cooper, R. Shrestha, N. Menda, R.T. Nelson, L. Matteis, M. Skofic, R. Bastow, P. Jaiswal, L. Mueller, and G. McLaren. Towards a reference plant trait ontology for modeling knowledge of plant traits and phenotypes. *proceedings of the 4th Conference on Knowledge Engineering and Ontology Development, 4-7 October 2012, Spain*, 2012.

[2] Baptiste Darnala, Florence Amardeilh, Catherine Roussey, Konstantin Todorov, and Clément Jonquet. C3po : a crop planning and production process ontology and knowledge graph. *Frontiers in Artificial Intelligence*, 6, 2023.

[3] M. Devare, C. Aubert, M-A. Laporte, L. Valette, E. Arnaud, and P.L. Buttigieg. Data-driven agricultural research for development : A need for data harmonization via semantics. *Proceedings of the Joint International Conference on Biological Ontology and BioCreative, Corvallis, Oregon, United States, August 1-4, 2016.*, 2016.

[4] Scott Federhen. The ncbi taxonomy database. *Nucleic acids research*, 40(D1) :D136–D143, 2012.

[5] F. Isbell, P.R. Adler, N. Eisenhauer, D. Fornara, K. Kimmel, C. Kremen, D.K. Letourneau, M. Liebman, H.W. Polley, S. Quijas, and M. Scherer-Lorenzen. Benefits of increasing plant diversity in sustainable agroecosystems. *J. Ecol.*, 105 :871–879, 2017.

[6] F. Michel, O. Gargominy, S. Tercerie, and C. Faron-Zucker. A model to represent nomenclatural and taxonomic information as linked data. application to the french taxonomic register, taxref. *Proceedings of ISWC 2017 Workshop on Semantics for Biodiversity (S4Biodiv 2017), Oct 2017, Vienna, Austria*, pages 1–12, 2017.

[7] Raul Palma, Ioanna Roussaki, Till Döhmen, Rob Atkinson, Soumya Brahma, Christoph Lange, George Routis, Marcin Plociennik, and Szymon Mueller. Agricultural information model. In Dionysis D. Bochtis, Claus Grøn Sørensen, Spyros Fountas, Vasileios Moysiadis, and Panos M. Pardalos, editors, *Information and Communication Technologies for Agriculture—Theme III : Decision*, pages 3–36. Springer International Publishing, 2022.

[8] Silvio Peroni. SAMOD : an agile methodology for the development of ontologies. page 1579911 Bytes, 2016. Artwork Size : 1579911 Bytes Publisher : figshare.

[9] María Poveda-Villalón, Alba Fernández-Izquierdo, Mariano Fernández-López, and Raúl García-Castro. LOT : An industrial oriented ontology engineering framework. 111 :104755, 2022.

[10] Catherine Roussey. Frenchcropusage : Thésaurus sur les cultures françaises. le thésaurus décrivant les cultures françaises par leur utilisation au format skos. *FAIRsharing.org*, 2018.

La méthodologie ACIMOV pour l'intégration agile et continue des modules ontologiques

F.-Z. Hannou[1], V. Charpenay[2], M. Lefrançois[2], C. Roussey[3], A. Zimmermann[2], F. Gandon[4]

[1] EDF R&D, Palaiseau, France
[2] Mines Saint-Etienne, Univ Clermont Auvergne, INP Clermont Auvergne, CNRS, UMR 6158 LIMOS, Saint-Étienne, France
[3] MISTEA, INRAE Centre Occitanie, Montpellier, France
[4] Inria, Univ. Cote d'Azur, I3S, CNRS, France

fatma-zohra.hannou@edf.fr, victor.charpenay@emse.fr, maxime.lefrancois@emse.fr, catherine.roussey@inrae.fr, antoine.zimmermann@emse.fr, fabien.gandon@inria.fr

Résumé

Ce travail décrit la méthodologie d'ingénierie d'ontologie Agile and Continuous Integration for Modular Ontologies and Vocabularies (ACIMOV) pour développer des ontologies et des vocabulaires. ACIMOV étend la méthodologie agile SAMOD pour mieux prendre en compte l'aspect modulaire des ontologies actuelles et le développement collaboratif entre différents types d'experts. ACIMOV adopte les principes de développement agile et DevOps. ACIMOV a été conçue pour être opérationnalisée à l'aide de plateformes de développement logiciel collaboratif Git et dotées de workflows d'intégration et de déploiement continus. Ces travaux ont été publiés dans un atelier international [4].

Mots-clés

Méthodologie d'ingénierie des ontologies, ontologie modulaire, Agile, Git, intégration et déploiement continu

Abstract

This work describes the Agile and Continuous Integration for Modular Ontologies and Vocabularies (ACIMOV) ontology engineering methodology for developing ontologies and vocabularies. ACIMOV extends the agile SAMOD methodology to better take into account the modular aspect of current ontologies and collaborative development between different types of experts. ACIMOV adopts the standard git-based approach for code version management, leveraging Agile and DevOps principles. It has been designed to be operationalized using collaborative software development platforms and tooled with continuous integration and continuous deployment workflows. This work has already been published in an international workshop [4].

Keywords

Ontology engineering methodology, modular ontology, Agile, Git, Continuous Integration and Continuous Deployment

1 Introduction

Ce travail décrit la méthodologie d'ingénierie d'ontologie *Agile and Continuous Integration for Modular Ontologies and Vocabularies* (ACIMOV) pour développer des ontologies et des vocabulaires, qui a été publiée à l'international [4]. ACIMOV étend la méthodologie agile SAMOD [5] pour (1) assurer l'alignement avec des ontologies de référence sélectionnées ; (2) planifier des développements modulaires intégrant la gestion des dépendances ; (3) définir les modules de l'ontologie qui peuvent être spécialisés pour des domaines spécifiques ; (4) permettre une collaboration active entre les ontologues et les experts du domaine ; (5) permettre aux développeurs d'applications de sélectionner des vues de l'ontologie pour leur domaine et des cas d'usages spécifiques. ACIMOV adopte l'approche standard de gestion des versions de code Git, et tire parti des principes Agiles et DevOps. Elle a été conçue pour être opérationnalisée à l'aide de plateformes de développement logiciel collaboratif telles que Github ou Gitlab, et est dotée de workflows d'intégration et de déploiement continus (workflows CI/CD) qui exécutent des contrôles syntaxiques et sémantiques sur le référentiel, valident les questions de compétence, spécialisent les modules, génèrent et publient les documentations de l'ontologie.

2 La méthodologie ACIMOV

Dans le contexte de nos travaux sur le déploiement du Web sémantique des objets (projet CoSWot [1]) et des agents (projet HyperAgents [2]), nous avons eu besoin de développer une ontologie pour standardiser l'échange de données. Nous souhaitions réutiliser plusieurs ontologies de références du domaine : W3C Thing Description (TD) [1], OGC&W3C Semantic Sensor Network (SSN) [3], ETSI Smart Applications Reference ontology (SAREF) [2]. Ces trois ontologies adoptent une conception modulaire. Nous identi-

1. https://coswot.gitlab.io/
2. https://anr.fr/Projet-ANR-19-CE23-0030

fions les besoins suivants pour notre ontologie : (O1) L'ontologie doit s'aligner sur les ontologies de référence de l'IoT ; (O2) L'ontologie doit être modulaire et comprendre des modules qui couvrent les connaissances communes à tous les composants de la plate-forme IoT ; (O3) L'ontologie doit réutiliser certaines ontologies identifiées pour les domaines d'application concernés ; (O4) L'ontologie doit avoir une structure homogène et prévisible, de sorte que des concepts similaires pour différents domaines soient décrits de la même manière ; (O5) Différentes représentations alternatives doivent être possibles pour tenir compte de la nécessité de manipuler des graphes de connaissances de petite taille dans des environnements contraints. (O6) On doit pouvoir sélectionner un sous-ensemble de l'ontologie (une vue) qui couvre les besoins d'une application spécifique.

Nous avons choisi de prendre comme point de départ la méthode SAMOD [5] et de l'adapter pour couvrir les besoins suivants : (M1) Les principes agiles doivent être adoptés pour améliorer la collaboration entre les ontologues, les experts du domaine et les *product owners*, avec des cycles courts et des incréments de travail ; (M2) Des réunions régulières avec toutes les parties doivent être organisées pour aider à prioriser les exigences découlant des cas d'utilisation et à choisir la cible de la prochaine itération ; (M3) Des réunions régulières entre les ontologues doivent être organisées pour aider à prioriser les modules sur lesquels travailler et pour s'assurer que le travail sur différents modules peut être mené en parallèle ; (M4) Des plateformes de développement logiciel collaboratif avec versionnement du code et suivi des problèmes doivent être adoptées ; (M5) Les principes DevOps doivent être adoptés pour permettre l'intégration et le déploiement continus des artefacts de l'ontologie (par exemple, les modules de l'ontologie, la documentation, les exemples).

ACIMOV cible les ontologues sensibilisés aux concepts et outils récents pour la gestion des version de code.La figure 1 donne un aperçu des sept étapes de la méthodologie ACIMOV. Elle comporte un cycle long de développement Agile impliquant les ontologues, les experts de domaine et les *product owners*, et deux cycles internes plus courts dédiés aux activités de développement menées par les ontologues. Les étapes sont les suivantes : (1) Collecter les exigences et identifier les ontologies de référence ; (2) Organiser une réunion de bilan (un événement) ; (3) Sélectionner les modules pertinents dans les ontologies de référence ; (4) Gérer le planning de développement des modules ; (5) Organiser une réunion de développement des modules (un événement) ; (6) Développer et tester des modules ; (7) Intégrer les modules et publier l'ontologie. ACIMOV est documentée en ligne [3] et des projets témoins facilitent son adoption et son outillage [4]. Différentes pistes de développement sont encore à l'étude, dont l'intégration dans des actions GitHub, ou la génération de représentations alternatives ou de vues de l'ontologie. Nous comptons également évaluer la mise en oeuvre d'ACIMOV sur différents projets.

3. https://acimov.gitlab.io/
4. https://gitlab.com/acimov/

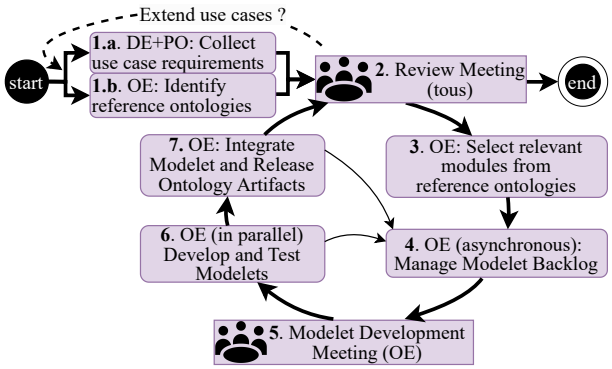


FIGURE 1 – Aperçu de la méthodologie ACIMOV. OE : Ontologue ; DE : Expert de domaine ; PO : *Product Owner*

Références

[1] V. Charpenay, M. Lefrançois, M. Poveda Villalón, and S. Käbisch. Thing Description (TD) Ontology, Editor draft, 10 May 2023. W3c working group draft, W3C, May 2023.

[2] 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. In *Energy Smart Appliances : Applications, Methodologies, and Challenges*. Wiley, 2023.

[3] Armin Haller, Krzysztof Janowicz, Simon JD Cox, Maxime Lefrançois, Kerry Taylor, Danh Le Phuoc, Joshua Lieberman, Raúl García-Castro, Rob Atkinson, and Claus Stadler. The SOSA/SSN ontology : a joint W3C and OGC standard specifying the semantics of sensors, observations, actuation, and sampling. *Semantic Web-Interoperability, Usability, Applicability an IOS Press Journal*, 56, 2019.

[4] Fatma-Zohra Hannou, Victor Charpenay, Maxime Lefrançois, Catherine Roussey, Antoine Zimmermann, and Fabien Gandon. The ACIMOV Methodology : Agile and Continuous Integration for Modular Ontologies and Vocabularies. In *2nd Workshop on Modular Knowledge associated with FOIS 2023*, 2023.

[5] Silvio Peroni. Samod : an agile methodology for the development of ontologies. In *Proceedings of the 13th OWL : Experiences and Directions Workshop and 5th OWL reasoner evaluation workshop (OWLED-ORE 2016)*, pages 1–14, 2016.

Remerciements

Ce travail a été en partie soutenu par des subventions de l'Agence française de la recherche (ANR) sur les projets CoSWot (ANR-19-CE23-0012) et HyperAgents (ANR-19-CE23-0030), et le projet européen ACCORD (Horizon Europe R&I convention de subvention du programme n° 101056973).

Utilisation de Modèles BERT pour Classer Automatiquement les Concepts de Domaine en Concepts de Haut Niveau DOLCE: Une Étude des Ontologies OAEI

Guilherme Sousa[1], Rinaldo Lima[2], Renata Vieira[3], Cassia Trojahn[1]

[1] IRIT: Institut de Recherche en Informatique de Toulouse, France
[2] Universidade Rural de Pernambuco, Recife, Brazil
[3] CIDEHUS, Universidade de Évora, Portugal

prenom.nom@irit.fr, rinaldo.jose@ufrpe.br, rinaldo.jose@ufrpe.br

1 Introduction

Les ontologies de haut niveau, avec leurs fondements philosophiques bien ancrés, servent d'outils indispensables dans l'ingénierie d'ontologies, facilitant des tâches telles que l'alignement d'ontologies. Cependant, toutes les ontologies ne sont pas ancrées dans des concepts de haut niveau, et certaines sont trop vastes pour une annotation manuelle. Les classifieurs automatiques de haut niveau proposent une solution en associant les ontologies de domaine aux ontologies de haut niveau. Des efforts récents, tels que [1], se sont concentrés sur la construction de jeux de données d'entraînement à partir des entités OntoWordNet, alignées sur les concepts DOLCE, pour évaluer les classifieurs destinés à prédire les concepts de haut niveau des entités. Cet article étend cette recherche en évaluant les performances des modèles de classification et l'impact de l'utilisation des commentaires d'entités seuls en tant que caractéristiques combinées à l'utilisation de modèles de langage plus grands.

Un autre aspect de cette recherche est de traiter les cas de multi-héritage, où différents concepts de haut niveau dans DOLCE peuvent découler de la hiérarchie des entités. Les améliorations dans ce cas impliquent une étape de désambiguïsation et de filtrage des chemins menant à plusieurs concepts. Le classifieur présentant les meilleurs résultats lors de l'entraînement est ensuite utilisé pour analyser la distribution des concepts de haut niveau dans les ontologies des jeux de données OAEI, ainsi que leurs alignements de référence.

2 Matériaux et méthodes

2.1 Jeux de données d'entraînement

Le jeu de données **Lopes22-5c** [1], le jeu de données original, utilisé pour l'entraînement des modèles de prédiction de concepts de haut niveau, comprend 116838 entités dérivées d'OntoWordNet, chacune liée à l'un des cinq concepts de haut niveau de DOLCE (Endurant, Perdurant, Qualité, Situation et Abstrait). Il est organisé en trois colonnes : Concept (concept de haut niveau DOLCE), Label (`rdfs:label`), et Commentaire (`rdfs:comment`). **Sousa23-5c**, une reconstruction de Lopes22-5c, aborde les préoccupations de multi-héritage en filtrant les entités ambiguës, tandis que **Sousa23-6c** aborde le déséquilibre de distribution des concepts en divisant Endurant en deux sous-groupes, créant ainsi un jeu de données plus équilibré avec six concepts. Pour gérer les cas de multi-héritage, deux scénarios sont considérés : l'un dans WordNet et l'autre dans la hiérarchie DOLCE. Des jeux de données de test basés sur les ontologies d'organisation de conférences de l'OAEI ont également été créés pour évaluer les modèles de classification, avec des concepts de haut niveau attribués en utilisant un alignement de référence fourni dans [3], ce qui donne les jeux de données **Conference-5c** et **Conference-6c**.

2.2 Modèles d'apprentissage

Le modèle de prédiction présenté dans [1] utilise à la fois des étiquettes et des commentaires, comprenant un système composé de deux parties. La première partie utilise un réseau de neurones à propagation (FNN), prenant la moyenne des plongements de mots des étiquettes en entrée, tandis que la deuxième partie utilise une architecture BiLSTM pour contextualiser les plongements appris pour chaque mot dans les commentaires du jeu de données. Bien qu'efficace, des architectures plus robustes comme BERT peuvent fournir de meilleurs résultats, comme indiqué dans [2], en raison de la capacité accrue de BERT à gérer le contexte pour de meilleures représentations de texte en langage naturel.

Des enjeux surviennent lorsque des entités partagent la même étiquette mais sont assignées à différents concepts de haut niveau dans le jeu de données Lopes22-5c, ce qui peut potentiellement affecter la capacité du modèle à discerner adéquatement les concepts. De plus, la rareté des commentaires dans les ontologies pose un obstacle supplémentaire, limitant potentiellement la généralisation du modèle lors des tests. Pour répondre à ces problématiques et améliorer la généralisation, une approche d'entrée unifiée incorporant à la fois des étiquettes et des commentaires a été

adoptée, en exploitant BERT avec une tête de classification pour prédire les concepts de haut niveau.

3 Évaluation expérimentale

En utilisant trois ensembles de données (Lopes22-5c, Sousa23-5c et Sousa23-6c), une validation croisée est utilisée après le sous-échantillonnage des instances de concepts majoritaires pour normaliser le jeu de données avant l'entraînement. Nous avons choisi Glove 6B pour les plongements de mots en raison de son équilibre entre performance et taille du modèle. Plusieurs modèles de base sont testés, notamment Bernoulli Naive Baye (BNB), Réseau de Neurones à Propagation Avant (FNN), Naive Bayes Gaussien (GNB), Arbre de Décision (DT), Forêt Aléatoire (RF), Régression Logistique (LR), et Machine à Vecteurs de Support (SVM). Les modèles proposés Model-Lopes et BERT sont entraînés avec des optimiseurs et des hyperparamètres spécifiques, évalués en utilisant la métrique micro-F1, et testés avec différentes combinaisons d'entrées. Notamment, l'utilisation des seuls commentaires tend à donner de meilleurs résultats dans tous les classifieurs, sauf pour le Naive Bayes Gaussien dans les ensembles de données Lopes22-5c et Sousa23-5c. Le modèle BERT surpasse constamment les autres, atteignant même des résultats significatifs lors de l'utilisation de l'entrée étiquette+commentaire. Les matrices de confusion pour BERT révèlent des erreurs de classification notables, notamment entre les Perdurant et Situation.

4 Application du Meilleur classifieur : Une Évaluation sur les Jeux de Données OAEI

Cette section se penche sur une analyse des concepts de haut niveau à travers divers axes OAEI [1], ainsi qu'un examen des caractéristiques des commentaires au sein des entités ontologiques. Tout d'abord, la distribution des concepts de haut niveau dans les ontologies est analysée en utilisant les estimations du modèle BERT. Les entités dans chaque axe d'ontologie, à l'exclusion des nœuds vides et des propriétés, sont analysées, avec des étiquettes collectées à partir de prédicats d'étiquetage, ou des identifiants de ressources. Les ontologies Complex, Food et BioDiv présentent des concentrations significatives d'Endurants, contrastant avec des concepts plus uniformément distribués dans d'autres axes. De même, la distribution par le modèle entraîné Sousa23-6c met en évidence des concentrations prononcées dans les ontologies d'Anatomie, d'Alimentation, de BioML et de KG. Des divergences entre les deux modèles sont observées, notamment dans les distributions d'entités de Qualité à travers les axes.

La cohérence des alignements entre les entités correspondantes est également évaluée, révélant des similitudes entre les deux modèles dans plusieurs axes et un nombre plus élevé de correspondances du même type dans l'axe de la Conférence, cependant, avec des divergences dans l'Anatomie, reflétant des différences de distribution sous-jacentes. BioDiv, en particulier, met en évidence des enjeux dans l'alignement en raison de classifications conflictuelles. Notamment, les modèles à 5 et 6 classes rencontrent des difficultés avec les correspondances dans MSE, attribuables à des informations d'entité éparses. Enfin, la discussion s'étend à la distribution terminologique, où la rareté des commentaires dans les axes d'ontologie pose des défis pour la généralisation du modèle.

1. Les descriptions détaillées de ces axes peuvent être trouvées sur `https://oaei.ontologymatching.org/2022/` (consulté le 01/07/23)

5 Conclusion et Travaux Futurs

Dans cette étude, la prédiction de concepts de haut niveau est explorée, générant des ensembles de données Sousa23-5c et Sousa23-6c avec 5 et 6 concepts, respectivement, dérivés d'OntoWordNet tout en abordant l'enjeu du multi-héritage lors de la génération d'ensemble de données. Les résultats de ce travail soulignent l'importance de `rdfs:comment` pour la compréhension automatisée des concepts par le système. De plus, le modèle offrant les meilleures performances est appliqué pour estimer les distributions de concepts dans les ontologies à partir des jeux de données OAEI. L'analyse des alignements de référence a relevé une forte proportion de correspondances partageant le même type.

Pour les travaux futurs, l'expérimentation par la mise en place de nouvelles architectures d'apprentissage profond est envisagée pour améliorer les résultats. De plus, l'exploitation de la structure ontologique en tant qu'information contextuelle dans les modèles de classification peut améliorer la prédiction de concepts de haut niveau, en particulier dans les cas d'ambiguïté d'étiquetage. De plus, la prévalence de correspondances partageant le même type dans certaines axes OAEI suggère la possibilité d'améliorer les performances du système de correspondance en alignant les entités avec des types de haut niveau similaires.

Références

[1] Alcides Gonçalves Lopes Junior, Joel Luis Carbonera, Daniela Schimdt, and Mara Abel. Predicting the top-level ontological concepts of domain entities using word embeddings, informal definitions, and deep learning. *Expert Syst. Appl.*, 203 :117291, 2022.

[2] Alcides Lopes, Joel Luis Carbonera, Daniela Schmidt, Luan Fonseca Garcia, Fabrício Henrique Rodrigues, and Mara Abel. Using terms and informal definitions to classify domain entities into top-level ontology concepts : An approach based on language models. *Knowl. Based Syst.*, 265 :110385, 2023.

[3] Daniela Schmidt, Cássia Trojahn, Renata Vieira, and Mouna Kamel. Validating top-level and domain ontology alignments using wordnet. In *Proceedings of the IX ONTOBRAS Brazilian Ontology Research Seminar, Curitiba, Brazil, October 3rd, 2016*, volume 1862 of *CEUR Workshop Proceedings*, pages 119–130. CEUR-WS.org, 2016.