POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Extraction automatique d'une base de connaissances à partir de documents archéologiques et patrimoniaux en français

ERWAN MARCHAND
Département de génie informatique et génie logiciel

Mémoire présenté en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
Génie informatique

Avril 2020

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Ce mémoire intitulé :

Extraction automatique d'une base de connaissances à partir de documents archéologiques et patrimoniaux en français

présenté par **Erwan MARCHAND**
en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
a été dûment accepté par le jury d'examen constitué de :

Michel DESMARAIS, président
Michel GAGNON, membre et directeur de recherche
Amal ZOUAQ, membre et codirectrice de recherche
Bram ADAMS, membre

REMERCIEMENTS

J'aimerais remercier mon directeur de recherche, M. Michel GAGNON, ainsi que ma codirectrice de recherche, Mme Amal ZOUAQ, qui m'ont donné la chance d'effectuer cette maîtrise et m'ont apporté un soutien continu tout au long de mes travaux.

Je tiens également à remercier le Ministère de la Culture et des Communications du Québec qui a permis la réalisation de ce projet.

Finalement, je remercie ma famille, ma compagne et mes amis pour leurs soutiens continus, même dans les moments les plus difficiles.

RÉSUMÉ

Le Web sémantique est une extension du Web standard mettant l'accent sur les modèles de données afin de permettre une meilleure réutilisation de celles-ci et de rendre leur traitement automatique par des machines plus aisé. Il repose sur des ontologies, qui dictent les types de données pouvant y être contenus ainsi que les relations possibles entre celles-ci. Afin de créer une ontologie complète, il est nécessaire de procéder en deux étapes, la création de l'ontologie de base, c'est-à-dire la définition des classes (ou types de données tels que *Personne*, *Ville*, etc.) et des relations les liant (tel que *est né à*, *réside à*, etc.), puis, le peuplement de cette ontologie, c'est à dire l'ajout de toutes les instances et leurs relations.

Avec pour objectif de sémantiser au maximum ses données, le *Ministère de la Culture et des Communications du Québec* (MCCQ) a fait appel à *Polytechnique Montréal* afin de créer une ontologie complète permettant la représentation de ses données patrimoniales. L'ontologie, en cours de développement, porte notamment sur les contenus archéologiques que le MCCQ possède sous différents formats ainsi que sur le contenu du *Patrimoine immobilier, mobilier et immatériel du Québec* (PIMIQ), répertoire entretenu par le MCCQ.

En parallèle au Web sémantique se développe, également, ces dernières années notamment, le traitement automatique de la langue naturelle (TALN). Les technologies liées au TALN permettent, entre autres, l'extraction automatique d'informations dans du contenu textuel telles que les entités et les relations entre celles-ci. Le domaine du TALN porte un intérêt majeur dans le cadre du peuplement automatique d'une ontologie, car, bien exploité, il permet d'extraire automatiquement les entités et les relations pouvant peupler une ontologie.

Ce mémoire présente les travaux effectués en utilisant des outils du TALN afin d'extraire des informations de textes patrimoniaux dans le but de peupler l'ontologie du MCCQ par l'extraction automatique d'entités et relations entre celles-ci. Dans un premier temps, les travaux portent sur l'extraction automatique d'informations au sein de rapports archéologiques manuscrits et non formatés dont nous récupérons le contenu à l'aide d'un outil de reconnaissance optique de caractères (ou OCR). Dans un second temps, l'intérêt est porté sur les fiches de patrimoines immobiliers du PIMIQ, mieux formatées et dans lesquelles l'identification de certaines relations récurrentes est possible.

ABSTRACT

The Semantic Web is an extension of the standard Web that focuses on data models to better reuse these data and make its automatic processing easier. It is based on models, called ontologies, that dictate the types of data that can exist and the possible relationships between them. In order to create a complete ontology, it is necessary to proceed in two stages. First, the creation of the basic ontology by defining classes (or data types such as *Person*, *City*, etc.) and relations linking them (such as *is born in*, *lives in*, etc.). Secondl, by populating the ontology with instances and links between them.

With the goal of semanticizing data, the *Ministère de la Culture et des Communications du Québec* (MCCQ) asked *Polytechnique Montréal* to create a complete ontology to represent its heritage data. The ontology, currently being developed, includes the archaeological content that the MCCQ possesses in different formats as well as the contents of *Patrimoine immobilier, mobilier et immatériel du Québec* (PIMIQ), a repertoire maintained by the MCCQ.

In parallel with the Semantic Web, the automatic processing of natural language (NLP) is also in development. NLP-related technologies allow the automatic extraction of information from textual content such as entities and the relations between them. The domain of NLP carries a major interest in the framework of the automatic population of an ontology because, well exploited, it enables the automatic extraction of entities and relations that can populate an ontology.

This thesis describes the work carried out in this context with the use of NLP techniques to extract entities and relations between them from heritage texts with the aim of populating the MCCQ ontology. Initially, the work focuses on the automatic extraction of information in hand-written and unformatted archaeological reports. In a second step, the focus is placed on the PIMIQ content, which is better formatted, making the identification of certain recurring relationships possible.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

LISTE DES SIGLES ET ABRÉVIATIONS

MCCQ	Ministère de la Culture et des Communications du Québec
NER	Named-Entity Recognition ou Reconnaissance d'entités nommées
NLP	Natural language processing (équivalent à TALN en français)
OIE	Open Information Extraction ou Extraction ouverte d'informations
OWL	Web Ontology Language
PIMIQ	Patrimoine immobilier, mobilier et immatériel du Québec
RDF	Resource Description Framework
SPARQL	SPARQL Protocol and RDF Query Language
TALN	Traitement automatique de la langue naturelle

LISTE DES ANNEXES

CHAPITRE 1 INTRODUCTION

1.1 Contexte des travaux

Le Web sémantique permet une représentation de données aisément utilisables automatiquement [5]. Il permet de faire un lien entre des données brutes compréhensibles par des humains, mais difficilement par des ordinateurs, et des représentations facilement exploitables automatiquement.

Dans le but de profiter des avantages du Web sémantique et de mettre en valeur une quantité importante de données brutes présentes uniquement en format textuel, le MCCQ a décidé de faire appel à Polytechnique Montréal afin de développer une bases de connaissances à partir de ses données archéologiques et patrimoniales.

1.2 Problématique

Une tâche majeure de la génération d'une base de connaissances est son peuplement. Une fois les classes et relations mises en place, il est nécessaire d'ajouter des instances à ces classes et de les relier entre elles avec les relations. Dans certains cas, cette tâche peut être effectuée manuellement si l'ontologie est de petite envergure. Dans d'autres cas, cette tâche peut être automatisée relativement facilement, telle que dans le cas de Wikipedia, où les instances sont des éléments directement extraits des infobox (ou infoboîtes).

Le corpus de documents archéologiques fourni par le MCCQ comporte 13 058 rapports archéologiques dont la plupart contiennent plusieurs dizaines de pages de texte, tandis que le corpus PIMIQ comporte 39 625 descriptions de sites immobiliers. Les rapports archéologiques contiennent, principalement, des descriptions des différents éléments qui sont trouvés sur les sites archéologiques, tel que les artefacts, ainsi que leurs emplacements. Des plans et photos, que nous ne traitons pas dans le cadre de nos recherches, se trouvent également dans la plupart des rapports archéologiques. Les descriptions de sites immobiliers du PIMIQ contiennent des informations textuelles notamment sur l'histoire des différents patrimoines, à savoir, leurs constructions et les différents évènements qui ont pu s'y produire. On y trouve également des descriptions des valeurs qui sont attribuées à ces patrimoines ainsi que des descriptions des éléments caractéristiques de ceux-ci. Dû à la quantité importante de rapports et à la taille du corpus du PIMIQ, l'extraction manuelle des informations intéressantes pour

peupler une base de connaissances n'est donc pas envisageable. La plupart des informations étant uniquement au format textuel et non présentes dans des éléments structurés, l'extraction ne peut être faite de manière directe, comme dans le cas de DBpedia. C'est la raison pour laquelle, afin de permettre l'extraction automatique d'entités et de relations, il est nécessaire de tirer profit des technologies liées au traitement automatique de la langue naturelle.

Les outils de traitement automatique de la langue naturelle, dont une revue est effectuée dans le chapitre 2 de ce mémoire, permettent une extraction automatique des entités et relations présentes au sein d'un texte. Ces outils performent bien sur des textes en anglais, mais ont des performances bien moindres sur des textes en français.

1.3 Objectifs

L'objectif principal des travaux de recherche effectués est de cibler et d'appliquer les méthodes permettant l'extraction d'entités et de relations au sein de textes en français. Ces entités peuvent, par la suite, être ajoutées à une base de connaissances.

Plus spécifiquement, le premier objectif des travaux est de permettre l'extraction automatique de relations de composition entre artefacts et matériaux au sein de rapports archéologiques en français. Cette relation a été choisie car elle est très présente dans ces rapports et l'ontologie créée pour le MCCQ repose notamment sur les artefacts, les matériaux et les liens entre ceux-ci.

Le second objectif spécifique des travaux est de permettre une extraction automatique d'entités et de relations que l'on retrouve dans les documents patrimoniaux du PIMIQ en français, sans que ces entités et relations soient spécifiées d'avance.

Le troisième objectif spécifique des travaux est, suite à l'évaluation des extractions qui ont été permises par l'utilisation d'outils, de proposer une méthode d'amélioration de ces extractions, mais également de proposer une méthode de spécialisation aux éléments du domaine étudié. Plus précisément, on cherche à reconnaître et extraire les entités qui font références à des patrimoines immobiliers, ce type d'entité n'étant pas reconnu initialement par les outils actuels de reconnaissance d'entités nommées.

Afin de répondre à ces objectifs, nous tentons de répondre aux questions de recherche sui-

vantes :

1. Dans quelle mesure est-il possible d'exploiter l'analyse syntaxique d'un texte afin d'extraire les relations spécifiques qui y sont présentes dans le contexte de documents archéologiques en français ?
2. Est-il possible de recourir à la traduction en anglais suivie de l'utilisation d'outils d'extraction ouverte d'informations pour extraire les relations présentes dans les textes en Français ?
3. Est-il possible d'améliorer les extractions effectuées à l'aide de modèles d'apprentissage supervisés ?

1.4 Contributions

Les travaux effectués dans le cadre de ce mémoire contiennent plusieurs contributions au domaine du traitement automatique de la langue naturelle sur des textes français dans le domaine patrimonial ainsi qu'au domaine du Web sémantique. Tout d'abord, nous présentons une méthode d'extraction de certaines relations (relation de composition entre artefacts et matériaux dans notre cas) basée uniquement sur l'utilisation de règles syntaxiques par l'identification de patrons. Ensuite, nous présentons une méthode qui consiste à traduire les textes en anglais et utiliser les outils matures qui ont étés développés pour cette langue. Par la suite, nous présentons une méthode permettant, à l'aide d'un modèle d'apprentissage supervisé, d'améliorer le taux d'extraction obtenu par les outils déjà existants, mais également permettant l'extraction de nouveaux types d'entités jusqu'alors inconnus des outils existants. Finalement, nous proposons une méthode de création d'ontologie, d'identification des classes et relations à créer et de peuplement.

De plus, dans le cadre de l'atelier *DOING : Intelligent Data – From Data to Knowledge* de 2020, nous avons soumis une publication intitulée *Extraction of a Knowledge Graph from French Cultural Heritage Documents* [6] où nous décrivons de manière succincte les travaux effectués dans le cadre de ce mémoire.

1.5 Structure du mémoire

Le mémoire est divisé de la manière suivante : Dans un premier temps, une revue de littérature des outils de traitement automatique de la langue naturelle est effectuée dans le chapitre 2.

Dans le chapitre 3, nous présentons une première approche d'extraction de relations en nous concentrant sur une relation précise à extraire au sein de documents archéologiques. Nous y présentons une première méthode basée uniquement sur l'utilisation de patrons syntaxiques. Ensuite, dans le chapitre 4, nous présentons une autre méthode d'extraction d'entités et de relations par l'utilisation d'outils en anglais sur des textes traduits du français issus de fiches patrimoniales du PIMIQ. Finalement, dans le chapitre 5, nous effectuons une synthèse des travaux effectués et discutons des limitations des approches proposées ainsi que des améliorations futures.

CHAPITRE 2 REVUE DE LITTÉRATURE

2.1 Extraction d'information ouverte

La notion de traitement automatique de la langue naturelle remonte jusqu'au milieu du XX^{e} siècle. En 1950, Alan M. Turing publie un article [7] posant la question "*Can machines think ?*" ("*Les machines peuvent-elles penser ?*"), démarrant ainsi les questionnements et recherches liés à l'intelligence artificielle. La même année, Turing développe un test permettant de prouver, selon lui, si une machine est capable de penser ou non. Ce test se déroule de la manière suivante : un humain communique à l'écrit à la fois avec un autre humain et une machine, si ce premier n'est pas capable de discerner lequel est lequel, la machine passe le test. Bien que sa réelle utilité soit contestée de nos jours [8], le test de Turing pose tout de même les bases de la notion d'intelligence artificielle, une machine capable de simuler une forme d'intelligence. En 1954, en plein contexte de guerre froide, naît la première forme d'intelligence artificielle et notamment de traitement automatique de la langue naturelle, un traducteur automatique du russe vers l'anglais [9].

Depuis, le traitement automatique de la langue naturelle a beaucoup évolué. Il s'est fractionné en de nombreux domaines d'applications, tels que le résumé automatique de textes [10], la traduction automatique, la compréhension de textes ou encore la conception d'agents conversationnels (ou *chatbots*). Le domaine auquel nous nous intéressons dans le cadre de nos recherches concerne l'extraction ouverte d'informations (OIE), notre objectif étant de permettre l'extraction d'entités et de relations à partir de textes en français. La recherche évoluant rapidement dans ce domaine, de nombreux instituts et universités ont développé leurs outils. On compte, parmi les outils les plus récents, Ollie [11], ClausIE [12], Stanford NLP [4], NESTIE [13], MinIE [14] ou encore MinScIE [15], par ordre chronologique de parution. Ce sont les plus populaires et performants selon certaines études [16] [17].

L'intérêt de l'utilisation de tels outils est de permettre l'extraction des relations présentes dans un texte. Ainsi, l'extraction ouverte d'informations sur la phrase *Victor Hugo est un écrivain français* devrait nous donner les relations ⟨*Victor Hugo, est, écrivain*⟩ et ⟨*Victor Hugo, est, français*⟩. Bien que cette tâche puisse paraître simple à réaliser pour un humain, elle reste complexe à automatiser sur une machine. Le processus d'extraction d'informations dans un domaine donné s'effectue en plusieurs étapes tel que décrit dans [3]. Tout d'abord, il doit y avoir identification et extraction des entités nommées, c'est à dire, les éléments du

monde réel pouvant être identifiés par un nom tel que les personnes, lieux etc. Dans un second temps, il doit y avoir identification et extraction des relations.

La première étape de l'OIE consiste en une extraction d'entités présentes dans un texte et en leur classification par type. Ainsi, dans notre exemple *Victor Hugo est un écrivain français*, un extracteur devrait idéalement donner les couples d'entités et types (*Victor Hugo*, *PERSONNE*), (*écrivain*, *PROFESSION*) et (*français*, *NATIONALITÉ*). Parmi les outils existants permettant l'extraction d'entités nommées (ou NER), on compte Stanford NLP [4] et spaCy [18] comme les plus performants [19]. L'outil d'extraction d'entités nommées de Stanford NLP ne possède pas de modèle en français contrairement à l'outil spaCy. L'outil spaCy performe, cependant, notablement moins bien que l'outil de Stanford NLP [19] et permet d'identifier un plus faible nombre de types distincts d'entités.

Le seconde étape de l'OIE consiste en une extraction des relations présentes dans un texte. Certaines de ces relations peuvent être identifiables aisément par un verbe, tel que la relation ⟨*Victor Hugo, né à, Besançon*⟩ issu de la phrase *Victor Hugo est né à Besançon*, tandis que d'autres peuvent être implicites tel que la relation ⟨*Louise Labé, est, poète*⟩ issu de la phrase *La poète Louise Labé est née à Lyon*. Évaluer les sorties d'un extracteur de relations est, cependant, une tâche difficile, la définition exacte d'une relation étant floue [17].

Ollie [11] est un premier outil permettant l'extraction de relations au sein de textes. L'avantage de cet outil, en comparaison de ses prédécesseurs, est qu'il permet, selon ses concepteurs, d'extraire des relations qui ne sont pas dictées par des verbes. De plus, Ollie tient compte des informations contextuelles présentes dans la phrase en les incluant dans les relations extraites. Le fonctionnement de Ollie est basé sur l'apprentissage de patrons syntaxiques et lexicaux en se basant sur les extractions effectuées par l'outil REVERB [20].

ClausIE [12] est un autre outil d'extraction de relations basé sur l'utilisation de propositions. Les concepteurs de cet outil décrivent les propositions comme étant des parties de phrases contenant des informations cohérentes (contenant au moins un sujet et un objet). Toutes les propositions sont constituées, principalement, d'un sujet et d'un verbe. D'autres catégories, telles que des adverbes, adjectifs ou compléments d'objet, peuvent venir compléter ces propositions. Celles-ci sont par la suite utilisées comme des patrons afin de permettre la détection de relations. Chaque proposition contenant nécessairement un verbe, ClausIE ne permet pas la détection de relations qui ne sont pas dictées pas des verbes tel que dans notre exemple

précédent ⟨*Louise Labé, est, poète*⟩.

Stanford NLP [4] est une suite d'outils de traitement automatique de la langue naturelle. L'outil d'extraction de relations de cette suite surpasse ses prédécesseurs sur certaines tâches, selon l'évaluation effectuée par ses créateurs [21]. Cet outil remplace l'utilisation de nombreux patrons par seulement quelques patrons et ajoute un classificateur automatique afin de permettre la détection plus souple de propositions au sein de phrases plus longues.

NESTIE [13] est un outil d'extraction de relations proposé par l'université du Michigan. Contrairement à ses prédécesseurs, NESTIE offre la possibilité d'extraire des relations imbriquées de la forme *(arg1,rel1,(arg2,rel2,arg3))*, *arg* représentant des arguments et *rel* des relations. Précédemment, les relations extraites étaient uniquement binaires au format *(arg1,rel,arg2)*, *arg2* étant parfois optionnel si la relation est extraite uniquement d'une proposition "sujet verbe". Le fonctionnement de NESTIE repose également sur l'apprentissage de patrons syntaxiques à l'aide de méthodes de bootstrapping.

MinIE [14] et MinScIE [15] sont des outils d'extraction de relations proposés par l'université de Mannheim. Une première particularité de MinIE est de permettre la représentation de données de polarité, de modalité, d'attribution ou de quantité sous forme d'annotations sémantiques. La polarité permet d'identifier si le triplet est sous forme positive ou négative. Les phrases *Superman does live in Metropolis* et *Superman does not live in Metropolis* seront, ainsi, toutes deux représentées par le triplet *(Superman ; does live in ; Metropolis)*, mais dans le second cas, la polarité sera négative. La modalité permet d'indiquer la certitude du triplet selon la phrase d'origine. L'attribution permet d'indiquer l'origine du triplet si celui-ci peut être attribué à une personne. Le triplet *(Superman ; does live in ; Metropolis)* issus de la phrase *according to John, Superman does not live in Metropolis* pourra ainsi être attribué à *John*. La quantité permet de modifier tous les nombres et opérateurs de quantité trouvés dans le texte et n'apportant pas d'informations pour l'extracteur par des annotateurs. Par exemple, les phrases *9 cats*, *all cats* et *almost about 100 cats* seront toutes retranscrites par *Q cats*, avec *Q* comme annotateur de quantité. La seconde particularité de MinIE est de permettre la suppression de parties trop spécifiques. MinScIE est une extension de l'outil MinIE permettant d'améliorer ses performances en rendant possible la prise en compte de citations dans des textes du domaine scientifique. MinScIE permet, en effet, de détecter le présence d'une référence à un article au sein d'un texte ainsi que le contexte de cette référence.

2.2 Analyse syntaxique de phrases

L'analyse syntaxique de phrases est une des méthodes couramment utilisées dans des contextes de traitement automatique de la langue naturelle. Elle consiste à analyser une phrase en générant sa représentation sous forme d'arbre syntaxique. Dans cet arbre syntaxique, les mots contenus dans la phrase analysée sont identifiés par leurs catégories (verbe, déterminant, adjectif etc.) et sont liés entre eux par leurs relations grammaticales (sujet, complément, déterminant etc.) Un exemple d'arbre syntaxique pour la phrase *Victor Hugo est un écrivain français* est présenté dans la Figure 2.1.

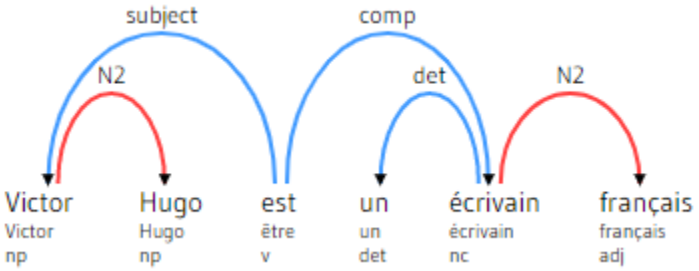

Figure 2.1 Exemple d'arbre syntaxique pour la phrase *Victor Hugo est un écrivain français*. Source : Généré à l'aide de l'outil FrMG Wiki [1]

Dans notre exemple, on peut aisément constater que le mot *Hugo* appartient au même groupe nominal que le mot *Victor*, cette relation étant spécifiée par l'annotation *N2*. On peut également y constater que *français* appartient au même groupe nominal que le mot *écrivain* dont le déterminant est le mot *un*. Finalement, on peut constater que la phrase contient le verbe *est* dont le sujet est le groupe nominal du mot *Victor* et dont le complément est le groupe nominal du mot *écrivain*. En considérant qu'un verbe représente une relation entre deux entités ou groupes nominaux, il nous est donc possible d'extraire de cette phrase le triplet ⟨*Victor Hugo, est, écrivain français*⟩ où *est* est la relation liant l'entité *Victor Hugo* à l'entité *écrivain français*. Une analyse syntaxique de phrase présente donc un avantage important dans l'extraction de relations au sein de phrases. Dans le cadre de nos travaux, nous utilisons le modèle français de l'outil SyntaxNet [22] afin d'effectuer l'extraction d'arbres syntaxiques de phrases. Dans la suite de ce mémoire, nous nous basons sur l'utilisation d'arbres syntaxiques afin d'extraire des patrons représentant des relations entre entités.

2.3 Données ouvertes et liées

Les données sur le Web sont présentes sous différents types de format. On peut trouver des données non structurées, tel que des textes, tout aussi bien que des données structurées, telles que les infoboîtes de Wikipédia. Lors d'une conférence au Gov 2.0 Expo de Washington DC en 2010, Tim Berners-Lee présente, pour la première fois, son modèle des cinq niveaux de types de données selon leurs degrés de structure. Michael Hausenblas a, par la suite, adapté et expliqué ce modèle [2]. Le modèle de Michael Hausenblas est présenté dans la Figure 2.2.

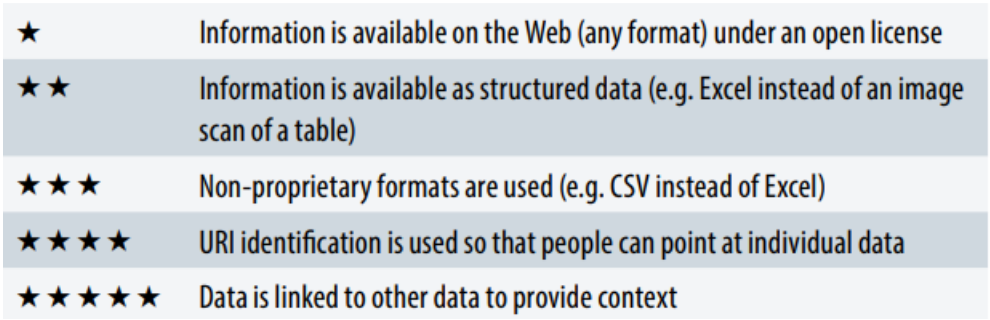

Figure 2.2 Modèle cinq étoiles de données de Michael Hausenblas. Source : Extrait de [2]

Le premier niveau de données présenté par Hausenblas est un niveau de données présentes sur le Web sous licence ouverte. Ces données sont accessibles librement par tout utilisateur qui le désire.
Le deuxième niveau de données contient l'avantage du premier en y ajoutant le fait que ces données sont structurées, par exemple sous forme de tableau Excel. Il est ainsi plus aisé, pour un utilisateur, de réutiliser ces données.
Le troisième niveau de données contient les avantages du deuxième, mais les données y sont toutes sous format ouvert. Contrairement à des données au format propriétaire dont seule une entité a le contrôle, des données au format ouvert offrent la possibilité à toute personne le désirant d'y accéder et d'y effectuer tout type de modification sans restriction ou sans avoir le besoin de payer une entité pour permettre l'accès aux données.
Le quatrième niveau de données contient les avantages du troisième en y ajoutant des identifications par URI (ou identifiant uniforme de ressource) des entités. Il est ainsi possible de lier des données de différentes sources entre elles. Une même entité se trouvant dans deux différentes sources peut donc être connectée à l'aide d'une unique URI.
Le cinquième niveau de données contient les avantages du quatrième en y ajoutant des liens entre les entités. Ces données sont considérées comme des données ouvertes et liées (ou Linked Open Data).

Le format RDF, utilisé par le Web sémantique, les ontologies et les bases de données, est un format de données de niveau cinq étoiles. En effet, chaque entité y est représentée par une URI et les entités sont mises en lien les unes avec les autres à l'aide de prédicats.

Dans le cadre de nos travaux, les données fournies par le MCCQ que nous utilisons sont de niveau zero (soit en licence fermée) pour les données archéologiques et de niveau un pour les données patrimoniales.

2.4 Ontologies et bases de connaissances

Une *ontologie* est une représentation formalisée des termes et concepts d'un domaine de connaissances. Une ontologie, dans sa forme la plus simple, est composée de triplets ⟨*Sujet*, *Prédicat*, *Objet*⟩ afin de représenter les relations entre ses différentes entités. Elle est définie par des classes et relations auxquelles sont ajoutées des instances afin de la peupler. Lorsqu'on ajoute des instances à une ontologie, on obtient alors ce que nous appelons une base de connaissances. Les classes permettent de définir les différents types d'entités qui peuvent se trouver dans l'ontologie tandis que les relations définissent les liens pouvant exister entre ces différentes entités. Les instances représentent les éléments du monde réel et sont associées à des classes. Une ontologie permettant de représenter des liens entre personnes possédera donc une classe *Personne*. Les entités *Victor Hugo* et *Léopoldine Hugo* pourraient être des instances de cette classe. Le lien *a pour enfant* pourrait être une relation entre instances de la classe *Personne*. On pourrait donc retrouver, dans une base de connaissances, le triplet ⟨*Victor Hugo*, *a pour enfant*, *Léopoldine Hugo*⟩. Des bases de documents telles que Wikipedia sont utilisées pour peupler des bases de connaissances. Parmi les bases de connaissances basées sur des ontologies les plus populaires, on compte :

— DBpedia [23] : une des bases de connaissances les plus populaires dans le domaine du Web sémantique. Elle offre une version structurée par une ontologie au format *RDF* du contenu de Wikipédia, notamment, des infoboîtes de chaque page ;
— Wikidata [24] : une base de connaissances pouvant être aussi bien consultée par des humains que par des machines. Son contenu n'est pas extrait automatiquement, contrairement à DBpedia, mais est plutôt construit par sa base d'utilisateurs. Elle se veut complémentaire à DBpedia ;
— Dublin Core [25] : une taxonomie permettant de standardiser la représentation en

format RDF de ressources digitales telles que des vidéos, images, pages web etc. ;
— schema.org [26] : un travail collaboratif entre Bing, Google et Yahoo ayant pour objectif de créer un schéma de données structurées pouvant être par la suite réutilisé pour structurer de nouvelles données.

2.5 SPARQL

Un des avantages majeurs de l'utilisation d'une ontologie et d'une base de connaissances est la possibilité d'effectuer des requêtes sur cette base de connaissance afin de répondre à toute sorte de questions. Il est possible de questionner aisément une base de connaissances à l'aide de requêtes SPARQL [27] afin de répondre à des questions telles que "Quels sont les auteurs qui ont coécrit un livre avec Victor Hugo", tâche fastidieuse avec Wikipédia uniquement, mais aisée et rapide à l'aide de DBpedia et SPARQL. SPARQL est un langage similaire à RDF pour l'écriture de requêtes capables de récupérer, ajouter ou modifier des données au sein d'une base de connaissances en RDF. Le Web sémantique présente donc un avantage majeur dans l'exploitation et la réutilisation de données. Le Tableau 2.1 présente le résultat de la question de connaissance *Quels sont les auteurs qui ont coécrit un livre avec Victor Hugo ?* sur la base de connaissances de DBpedia. On peut y constater que, selon la base de connaissances de DBpedia, *Didier Decoin* est la seule personne à avoir coécrit un livre avec *Victor Hugo*. La description de l'utilité de chaque ligne de la requête est :

— *SELECT DISTINCT ?nom_author* : l'opérateur *SELECT* permet d'indiquer les entités que l'on désire obtenir dans la liste des résultats de notre requête, l'opérateur *DISTINCT* permet de fitrer toutes les entitées qui sont dupliquées. Cette ligne nous permet donc de préciser que la sortie de la requête est composée de toutes les entités *?nom_author* distinctes ;
— *WHERE* : *WHERE* est un opérateur de filtrage permettant d'indiquer des conditions sur les données à retourner ;
— *?victor rdfs:label "Victor Hugo"@fr* : extrait toutes les entités *?victor* contenant l'étiquette française *Victor Hugo* ;
— *?livre dbo :author ?victor* : extrait l'ensemble des entités *?livre* dont on peut trouver un lien avec la relation *dbo :author* avec une des entités *?victor* extraite précédemment. Ici, on extrait donc l'ensemble des livres dont Victor Hugo est l'auteur ;
— *?livre dbo:author ?author.* : extrait, pour chaque entité *?livre*, la liste des entités *?author* en lien par la relation *dbo :author*. On extrait donc ici, pour chaque livre précédemment extrait, la liste de ses auteurs ;

— *FILTER (?author != ?victor)* : filtre toutes les entités *?author* qui ne sont pas distinctes de l'entité *?victor*. Ici, on élimine Victor Hugo de l'ensemble de la liste des nouveaux auteurs trouvés ;
— *?author rdfs:label ?nom_author* : extrait l'ensemble des étiquettes pour chaque entité *?author*. Ici, on récupère les noms, dans l'ensemble des langues, de chaque auteur trouvé ;
— *FILTER (lang(?nom_author) = 'fr')* : filtre l'ensemble des étiquettes qui ne sont pas en français. Chaque entité pouvant posséder un grand nombre d'étiquettes dans différentes langues, on cherche à garder uniquement celles en français.

Tableau 2.1 Question de compétences sur la base de connaissances de DBpedia : Quels sont les auteurs qui ont coécrit un livre avec Victor Hugo ?

Question de connaissance	Quels sont les auteurs qui ont coécrit un livre avec Victor Hugo ?

Requête SPARQL	SELECT DISTINCT ?nom_author
	WHERE
	{
	?victor rdfs :label "Victor Hugo"@fr.
	?livre dbo :author ?victor.
	?livre dbo :author ?author.
	FILTER (?author != ?victor)
	?author rdfs :label ?nom_author.
	FILTER (lang(?nom_author) = 'fr')
	}

Sortie	"Didier Decoin"@fr

Nous allons utiliser l'ensemble des techniques présentées dans notre revue de littérature pour, d'une part, extraire des relations de composition entre artefacts et matériaux au sein de rapports archéologiques, et, d'autre part, effectuer une extraction ouvert d'entités et de relations au sein de documents patrimoniaux.

CHAPITRE 3 EXTRACTION DE RELATIONS DE COMPOSITION AU SEIN DE RAPPORTS ARCHÉOLOGIQUES

3.1 Introduction

Dans le cadre de travaux réalisés en partenariat avec le MCCQ, une ontologie a été développée afin de représenter du contenu archéologique. Dans le but de permettre son peuplement, le MCCQ a mis à notre disposition des rapports archéologiques contenant une partie de l'information pouvant être utilisée. Dans le cadre de ce projet, les travaux effectués se concentrent sur l'extraction d'artefacts, matériaux et relations de compositions entre ceux-ci. Ce chapitre décrit la méthode mise en place et les résultats obtenus.

3.2 Présentation de l'ontologie utilisée

L'ontologie développée avec le MCCQ a pour principal objectif de permettre la représentation sémantique de contenus archéologiques, à savoir, des sites d'excavation, des artefacts, des matériaux, des rapports archéologiques et bien d'autres informations. Ici, nous utilisons une partie de cette ontologie, schématisée dans la Figure 3.1. Utiliser une telle ontologie permet, par la suite, de pouvoir la questionner à l'aide de requêtes SPARQL tel que présenté dans la section 3.7. La Figure 3.5, présente plus loin dans ce mémoire, illustre un exemple d'instances d'une relation de composition avec l'artefact et le matériau qui y sont liés.

La partie de l'ontologie utilisée est composée des classes suivantes :

- Artefact : Un objet que l'on trouve sur un site archéologique (par exemple : marteau) ;
- Matériau : Un matériau composant au moins un artéfact (par exemple : bois) ;
- Relation de composition : Une relation de composition entre un artefact et un matériau, une explication plus complète de cette classe est effectuée dans la suite de cette section ;
- Document : Un rapport de site archéologique selon son identifiant (par exemple : *s00213a1973v00_doc001* qui est l'identifiant d'un rapport archéologique) ;
- Site d'excavation : Un site archéologique (par exemple : Bassin de la rivière Chaudière (site Désy)).

et des relations :

- objet de relation de composition : Permet d'indiquer l'artefact concerné par une relation de composition ;

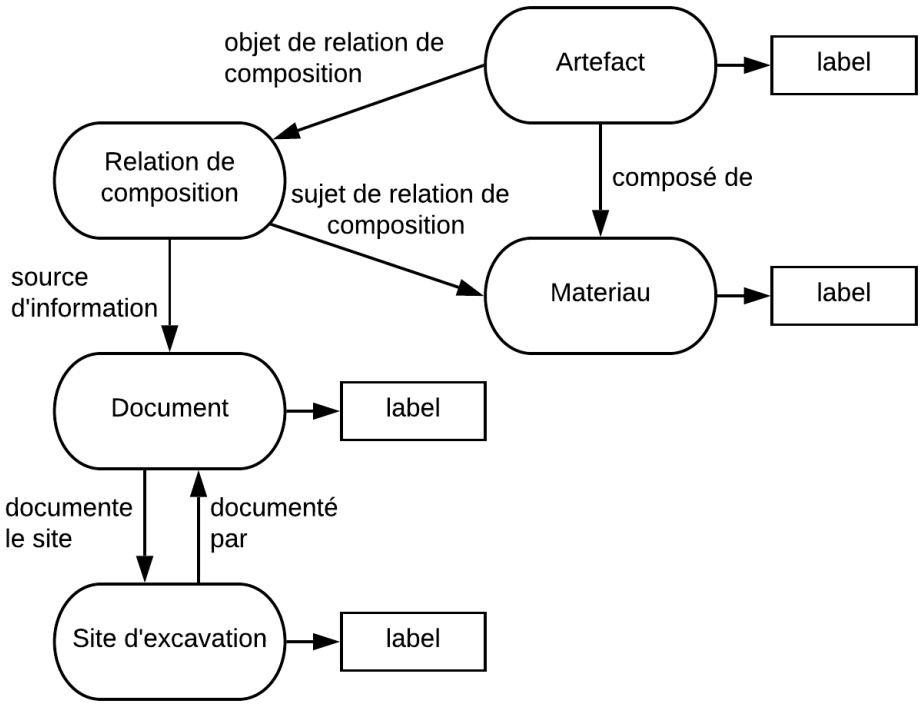

Figure 3.1 Schéma représentant la partie de l'ontologie utilisée

— sujet de relation de composition : Permet d'indiquer le matériau concerné par une relation de composition ;
— composé de : Permet de faire le lien direct entre un artefact et le matériau qui le compose ;
— source d'information : Permet d'indiquer le rapport archéologique dont est extrait une relation de composition ;
— documente le site : Permet d'indiquer le site qui est décrit dans un rapport archéologique ;
— documenté par : Relation inverse de la relation précédente, indique un rapport décrivant le site archéologique.

Cette ontologie permet d'accueillir une base de connaissances contenant les éléments extraits des rapports archéologiques.

Utiliser une classe "Relation de composition" permet d'ajouter des informations sur cette relation (tel que le document duquel celle-ci est extraite), ce qui ne serait pas possible avec

uniquement un lien direct entre la classe "Artefact" et la classe "Materiau". Ce lien existe toutefois afin de faciliter l'utilisation, par la suite, de l'ontologie dans certains cas (par exemple, si l'on se questionne sur les matériaux composant un artefact donné, il n'est pas nécessaire de prendre en compte l'instance de "Relation de composition" dans la requête.) Cette méthode d'utilisation d'une classe plutôt que d'une relation est également appliquée par Wikidata tel que discuté dans l'article [24] pour la traduction en RDF de déclarations complexes. Cette méthode est appelée la réification. Une description plus détaillée de cette méthode est présente dans [28]. Un des défauts majeur du langage RDF est son manque de méthode simple permettant d'ajouter du contexte à une relation.

La représentation de l'exemple *Victor Hugo a écrit Les Misérables en 1862 et Notre-Dame de Paris en 1831* en triplets RDF serait ⟨*Victor Hugo, a écrit, Les Misérables*⟩ et ⟨*Victor Hugo, a écrit, Notre-Dame de Paris*⟩. Sans réification de la relation *a écrit*, il est difficile d'ajouter les dates d'écritures de chaque livre aux relations. Il nous serait possible d'ajouter les triplets ⟨*Victor Hugo, a écrit un livre en, 1862*⟩ et ⟨*Victor Hugo, a écrit un livre en, 1831*⟩, mais cette représentation ne permet pas de discerner quel est le livre qui a été écrit à chaque date. La réification de cette relation dans notre exemple donnerait la liste des triplets : ⟨*écriture 1, auteur, Victor Hugo*⟩, ⟨*écriture 1, livre, Les Misérables*⟩, ⟨*écriture 1, date, 1862*⟩, ⟨*écriture 2, auteur, Victor Hugo*⟩, ⟨*écriture 2, livre, Notre-Dame de Paris*⟩ et ⟨*écriture 2, date, 1831*⟩. Dans cet exemple, la relation *a écrit* est réifiée par une classe *Écriture*, dont nous générons deux instances, *écriture 1* et *écriture 2*. Le nombre de triplets nécessaires à la représentation de la phrase en RDF suite à la réification de sa relation a fortement augmenté, il nous est cependant possible, maintenant, d'ajouter du contexte tel que la date à la relation d'écriture. La réification de relations en classe génère donc une représentation plus chargée en nombre de triplets RDF. Elle permet cependant, tel que mis en évidence dans l'exemple précédent, un ajout d'informations qui serait difficile sans réification.

3.3 Présentation des données disponibles

Les données fournies par le MCCQ dans le cadre de ce projet sont les suivantes :

- 13 058 rapports archéologiques ;
- Une liste de 3 000 artefacts extraite de Parc Canada (L_{APC}) ;
- Une liste de 200 matériaux produite par le personnel du MCCQ (L_{MAN}) ;
- Une liste de 14 041 compositions possibles d'artefacts extraite du contenu de PIMIQ servant de référence de départ dans le cadre de l'extraction de relations au sein de

textes, jouant ainsi un rôle d'un *"Oracle"* (L_{Oracle}). L'hypothèse de départ est qu'un artefact a peut être composé d'un matériau m seulement si la paire *(a,m)* se trouve au sein de cette liste.

À l'aide de ces listes sont générées deux listes :

- Une liste d'artefacts L_A, contenant ceux extraits de la liste L_{APC} et ceux présents dans le contenu de l'Oracle ;
- Une liste de matériaux L_M, contenant ceux de la liste L_{MAN} et ceux présents dans le contenu de l'Oracle.

On a donc les listes suivantes :

$$\begin{cases} L_A = L_{APC} \cup \{a | (a, m) \in Oracle\} \\ L_M = L_{MAN} \cup \{m | (a, m) \in Oracle\} \end{cases} \tag{3.1}$$

Les rapports archéologiques sont composés de textes, tableaux, photos et schémas. Dans le cadre de ce projet, l'extraction de contenu se focalise sur les textes et les tableaux. Le contenu d'un tableau et celui d'un texte étant différemment présentés, les méthodes d'extraction utilisées sont différentes. La section 3.4 présente la méthode d'extraction utilisée pour le contenu de tableaux tandis que la section 3.5 présente la méthode d'extraction utilisée pour le contenu de textes.

3.4 Extraction de relations de composition au sein de tableaux

La première phase de l'extraction de relations de composition au sein de rapports se concentre sur le contenu présent sous forme de tableaux. Le Tableau 3.1 présente une portion d'un tableau extrait d'un rapport. Tel que l'on peut le constater dans celui-ci, chaque ligne représente au moins une relation de composition entre un artefact (ou objet) et un matériau. L'objectif ici est donc de distinguer, dans un rapport, la présence d'un tel tableau, identifier s'il y a présence d'une colonne représentant les artefacts et une colonne représentant les matériaux. Il est ensuite nécessaire d'extraire de ces colonnes chaque objet et le matériau qui lui est associé.

Les rapports se trouvant uniquement au format PDF, il est tout d'abord nécessaire d'en extraire leur contenu au format texte afin de pouvoir les traiter plus simplement. À cette fin, l'outil *Omnipage* de *Kofax*[1] est utilisé. Celui-ci permet une reconnaissance automatique

1. https ://www.kofax.com/Products/omnipage

Tableau 3.1 Exemple de tableau extrait d'un rapport contenant des relations de compositions

Lot	Code	Matériau	Objet	No. frag.	No. obj.	Code fonction
2B1	3.1.1.13	Fer tréfilé	Crochet	1	1	4.7.2.99
2B2	2.1	Verre Incolore	Gobelet	1	1	4.1.3.3
2B2	2.1	Verre Incolore	Objet de service	1	1	4.1.3.4
2B2	2.1	Verre Incolore	Bouteille	2	1	4.10
2B2	2.1	Verre Incolore	Indéterminé	5	5	4.99
2B2	3.1.2	Métaux et alliages cuivreux	Monnaie	1	1	5.2
2B2	5.1.1	Os	Ossements	15	1	6.1.1
2B2	1.1.1.3	TCG sans glaçure	Brique	2	2	4.7.1.2
2B2	2.2.1.1	Verre teinté rég. vert	Vitre	3	3	4.7.1.1
2B2	2.2.1.1	Verre teinté rég. vert	Bouteille	2	2	4.10
2B2	2.3.1.1	V col transp vert foncé	Bouteille à vin	5	2	4.2.1.2

des caractères et ainsi l'extraction du texte présent dans un fichier PDF. Cet outil reconnait également la présence de tableaux et les affiche en texte sous le format suivant "contenu case 1 ;contenu case 2 ;contenu case 3 ;etc." chaque ligne du tableau étant une nouvelle ligne dans le fichier textuel.

Le processus d'extraction des relations de composition au sein des tableaux utilise ces fichiers textuels et fonctionne de la manière suivante :

- On parcourt l'ensemble des lignes extraites d'un rapport ;
- Tant que les lignes ne correspondent pas à un tableau (du texte pur par exemple), on continue ;
- Une fois qu'une ligne correspondant à un tableau est rencontrée, on parcourt chaque case afin de trouver si les mots "Objet"/"Artefact" et "Matériau" (une étude préliminaire ayant identifié ces appellations comme étant les titres de colonnes utilisés pour identifier les artefacts et matériaux) apparaissent comme titre de colonne et, si c'est le cas, on conserve leurs positions (numéros de colonne) ;
- On continue de parcourir chaque ligne et tant que celles-ci continuent de correspondre aux lignes d'un tableau (sous le format "contenu case 1 ;contenu case 2 ;contenu case 3 ;etc."), on extrait les éléments aux deux cases des positions de colonne précédemment récupérées et on génère des couples Artefact-Matériau avec ces éléments ;

— Une fois qu'une ligne ne correspond plus à du contenu de tableau, on considère que nous sommes sortis du tableau et revenons à la première étape

Une fois l'extraction des relations de composition au sein de tableaux terminée, les artefacts et matériaux extraits sont ajoutés aux deux listes précédemment définies à la section 3.3. On obtient ainsi les listes enrichies suivantes :

$$\begin{cases} L'_A = L_A \cup \{a | (a, m) \in Tableaux\} \\ L'_M = L_M \cup \{m | (a, m) \in Tableaux\} \end{cases} \tag{3.2}$$

Chaque couple est également ajouté à la base de connaissances comme décrite dans la section 3.7. Cette approche nous a permis d'extraire 277 468 couples.

3.5 Extraction des relations de composition au sein de textes

L'extraction des relations de composition entre artefacts et matériaux au sein du contenu textuel des rapports archéologiques s'effectue en deux étapes. Tout d'abord, nous identifions les phrases contenant à la fois au moins un artefact (un sous-ensemble de la phrase se trouve dans L'_A) et un matériau (un sous-ensemble de la phrase se trouve dans L'_M), puis nous analysons automatiquement la phrase à l'aide de sa structure (ou arbre) syntaxique afin de déterminer s'il y a bien une relation de composition entre les deux entités. Cette section décrit la méthode appliquée.

3.5.1 Méthodes existantes en extraction de relations de composition

De nombreux outils de traitement de la langue naturelle de l'état de l'art, tels que CoreNLP de Stanford [4] et MinIE [14], reposent sur la reconnaissance automatique d'entités et de verbes les liants. Malheureusement, les relations de composition entre artefacts et matériaux s'expriment généralement uniquement à l'aide de prépositions tel que dans l'exemple : *Seul un petit tube de fer et trois clous indéterminés furent retrouvés dans ce lot.* entre l'artefact *tube* et le matériau *fer*. On peut constater, tel que montré dans la Figure 3.2, que dans de tels cas, ces approches échouent à reconnaître la relation de composition présente dans la phrase, les deux triplets extraits étant ⟨*fer et trois clous indéterminés furent retrouvés, dans, ce lot*⟩ et ⟨*de fer et trois clous indéterminés furent retrouvés, dans, ce lot*⟩. Idéalement, le triplet que nous aurions dû obtenir afin de trouver la relation de composition présente dans la phrase est ⟨*tube, de, fer*⟩.

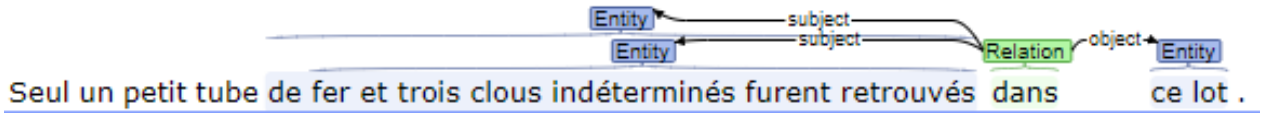

Figure 3.2 Exemple d'extraction automatique de relation en français avec CoreNLP

D'autres approches telles que PKDE4J [3] reposent sur l'utilisation de dictionnaires d'entités et d'arbres syntaxiques afin de déceler la présence de relation entre entités. L'approche proposée dans cette section est similaire. Grâce aux données fournies par le MCCQ, il est possible de générer des dictionnaires pour les artefacts et matériaux. À l'aide de l'outil SyntaxNet de Google, nous pouvous également générer l'arbre syntaxique de chaque phrase contenant au moins une paire *artefact - matériau*. Il nous est donc possible, tel que présenté dans la suite de cette section, d'analyser le chemin syntaxique séparant l'artefact du matériau et trouver des patrons pouvant représenter des relations de composition.

3.5.2 Détection des artefacts et matériaux

La détection des artefacts et matériaux dans le texte utilise les listes L'_A et L'_M précédemment créées. Le texte de chaque rapport est parcouru phrase par phrase et on recherche dans chacune la présence de matériaux ou artefacts sans considérer la casse (un matériau en minuscules dans la liste L'_M sera décelé dans le texte même s'il s'y trouve avec une majuscule). Pour chaque artefact et matériau trouvé au sein d'une phrase, on génère un triplet ⟨PHRASE,ARTEFACT,MATERIAU⟩. Suite à cette étape, nous avons un total de 2 280 171 triplets représentant potentiellement une relation de composition.

3.5.3 Identification des patrons lexico-syntaxiques pertinents représentant des relations de composition

Tel que décrit dans la section 3.5.1, la méthode utilisée ici repose sur l'utilisation de la structure syntaxique des phrases. L'objectif est de déterminer les patrons lexico-syntaxiques représentant une relation de composition entre un artefact et un matériau. À cette fin, la liste L_{Oracle}, présentée dans la section 3.3, qui contient des paires indiquant une relation de composition entre artefacts et matériaux, est utilisée. Les triplets ⟨PHRASE,ARTEFACT,MATERIAU⟩ validés par cet oracle, c'est-à-dire, les triplets avec (ARTEFACT,MATERIAU) $\in L_{Oracle}$, sont retenus. Un total de 40 751 triplets est ainsi retenu. L'intérêt de l'utilisation de l'oracle est de permettre d'avoir un échantillon de triplets ayant une forte probabilité de représenter une relation de composition. Pour chaque triplet, l'arbre syntaxique de la phrase le conte-

nant est généré à l'aide de l'outil SyntaxNet de Google. Le chemin lexico-syntaxique séparant l'artefact du matériau est ensuite identifié.

Un exemple d'un triplet extrait de rapport et validé par l'oracle est le suivant :
⟨"Le premier est cet insigne en cuivre représentant les armoiries papales.","insigne","cuivre"⟩

L'arbre syntaxique lié à cette phrase est schématisé sur la Figure 3.3.

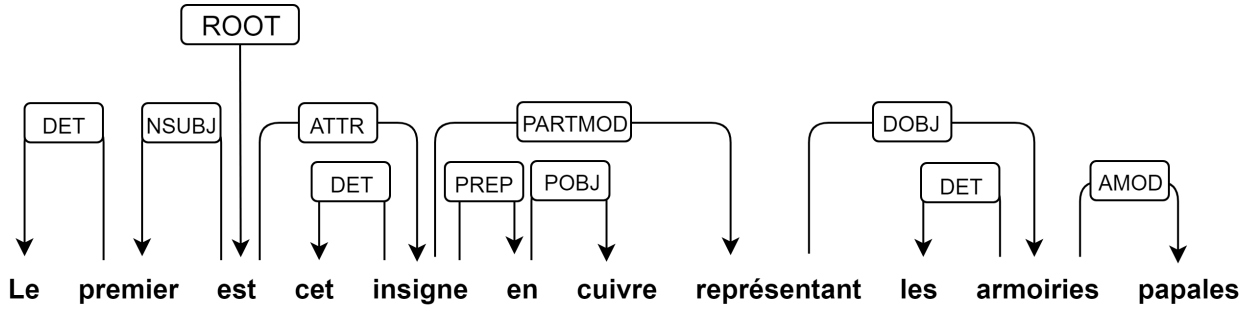

Figure 3.3 Schéma représentant l'arbre syntaxique de la phrase "Le premier est cet insigne en cuivre représentant les armoiries papales"

Le chemin lexico-syntaxique séparant "*insigne*" et "*cuivre*" est la préposition "*en*". On peut donc extraire de ce triplet le patron lexico-syntaxique "[Artefact] en [Matériau]".

Le nombre d'occurrences de chaque patron lexico-syntaxique trouvé parmi l'ensemble des triplets validés par l'oracle est ensuite comptabilisé. Sur un total de 40 751 couples valides, 23 345 ne possèdent aucun chemin syntaxique direct entre l'artefact et le matériau au sein de la phrase, 3 672 sont des occurrences du patron "[Artefact] de [Matériau]", 3 199 occurrences du patron "[Artefact] en [Matériau]" tandis que le reste se partage en de très nombreux patrons ayant de faibles occurrences et ne représentant que rarement des relations de composition, les plus fréquents d'entre eux étant présentés dans le Tableau 3.2, ces patrons ne sont pas conservés pour la détection de nouvelles relations (ces analyses ont été effectuées uniquement sur notre corpus, les patrons pour cette même relations pourraient différer sur d'autres corpus de textes). On considère un chemin syntaxique entre un artefact et un matériau comme direct lorsque le chemin syntaxique entre ces deux entités ne nécessite pas de remonter dans l'arbre syntaxique afin de parcourir le chemin entre le noeud de l'artefact et le noeud du matériau. Le tableau présent dans l'annexe A présente 50 cas dans lesquels le triplet ⟨PHRASE,ARTEFACT,MATERIAU⟩ est valide selon l'oracle mais le chemin syntaxique entre l'artefact et le matériau n'est pas direct. Dans ces cas là, nous avons remarqué qu'il n'y avait pas de relation de composition entre les entités. Nous considérons donc que l'absence de chemin syntaxique direct entre un artefact et un matériau au sein d'une phrase signifie, dans le cas de nos textes, l'absence de relation de composition entre ces éléments. Cette conclusion

nous a amené à déduire que les 23 345 cas où aucun chemin syntaxique n'est trouvé entre l'artefact et le matériau ne contiennent pas de relation de composition entre ceux-ci.

Tableau 3.2 Exemples de patrons extraits à faibles occurrences

Patron	Nombre d'occurrences
"[Artefact] suspendu. [Matériau]"	286
"[Artefact] a [Matériau]"	172
"[Artefact] en terre [Matériau]"	159
"[Artefact] materiel charbon de [Matériau]"	159
"[Artefact] en verre [Matériau]"	107
"[Artefact] du [Matériau]"	94
"[Artefact] de echantillon materiel charbon de [Matériau]"	93
"[Artefact] verre [Matériau]"	72
"[Artefact] fragment de [Matériau]"	46
"[Artefact] fragments de [Matériau]"	36

Nous en avons donc conclu qu'une relation de composition entre un artefact et un matériau au sein d'une phrase est majoritairement identifiable à l'aide d'un patron lexico-syntaxique "[Artefact] de [Matériau]" ou "[Artefact] en [Matériau]".

3.5.4 Détection des relations de composition

Afin de détecter de nouvelles relations de composition au sein des triplets candidats trouvés dans les textes mais non validés par l'oracle, nous utilisons les deux patrons syntaxiques précédemment trouvés. Pour chaque triplet, la structure syntaxique de la phrase est ajoutée, toujours à l'aide de l'outil SyntaxNet de Google. Le chemin syntaxique entre l'artefact et le matériau est identifié puis comparé aux deux patrons. Si celui-ci correspond à l'un de ces deux patrons, nous considérons qu'il y a une relation de composition entre l'artefact et le matériau au sein de cette phrase. Ainsi, nous avons pu trouver 124 354 relations de composition parmi les 2 280 171 triplets candidats. Ce faible taux provient du grand nombre de triplets étant générés pour une même phrase, mais ne représentant pas une relation de composition, tel que nous pourrons le voir dans l'exemple de la section 3.5.5. Toutefois, l'approche par patron permet de déceler un bien plus grand nombre de relations que l'utilisation de l'*Oracle* seule (124 354 contre 40 751). Chacun de ces triplets est ensuite ajouté à la base de connaissances comme décrite dans la section 3.7.

3.5.5 Exemple d'extraction

Soit la phrase extraite d'un rapport archéologique : "*La fonction entreposage est représentée par des fragments de bouteille de boisson gazeuse en verre américain, un fragment de couvercle ainsi qu'un fragment de pot en verre teinté régulier vert.*"

La liste des artefacts L'_A générée dans la section 3.3 contient les artefacts :

- "fragments de bouteille"
- "fragment de couvercle"
- "fragment de pot"
- "bouteille"
- "couvercle"
- "pot"

Dans chaque cas, on conserve uniquement le terme le plus long se trouvant dans la liste, on obtient donc uniquement :

- "fragments de bouteille"
- "fragment de couvercle"
- "fragment de pot"

La liste des matériaux L'_M contient :

- "verre américain"
- "verre teinté régulier"
- "verre"

On conserve ici le terme le plus long se trouvant dans la liste, on obtient donc uniquement :

- "verre américain"
- "verre teinté régulier"

La liste des triplets générés pour cette phrase est donc :

- T1 : ⟨PHR,"fragments de bouteille","verre américain"⟩
- T2 : ⟨PHR,"fragments de bouteille","verre teinté régulier"⟩
- T3 : ⟨PHR,"fragment de couvercle","verre américain"⟩
- T4 : ⟨PHR,"fragment de couvercle","verre teinté régulier"⟩
- T5 : ⟨PHR,"fragment de pot","verre américain"⟩
- T6 : ⟨PHR,"fragment de pot","verre teinté régulier"⟩

Afin d'alléger la lecture, nous avons abrégé la phrase par "PHR"

L'arbre syntaxique lié à cette phrase est schématisé sur la Figure 3.4

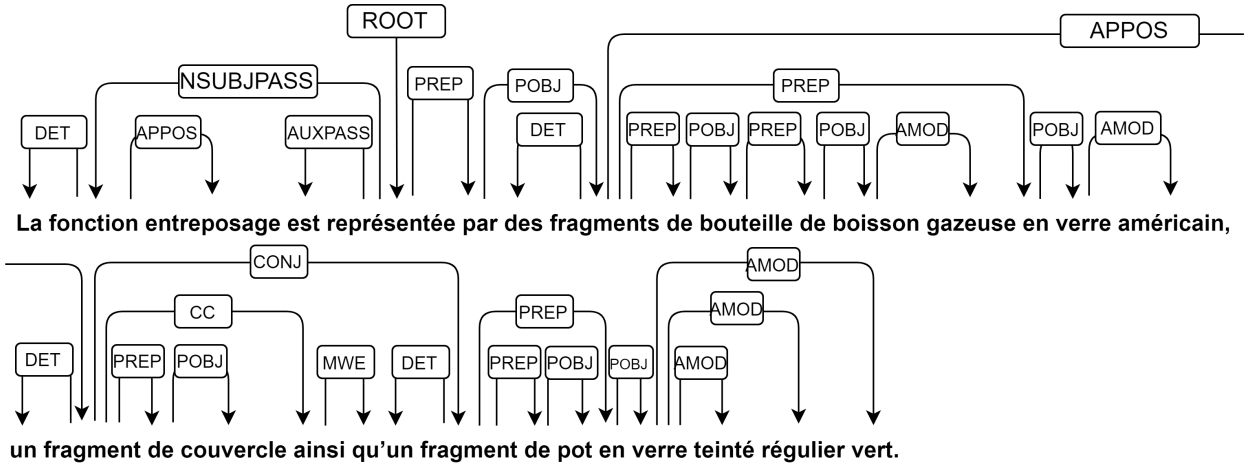

Figure 3.4 Schéma représentant l'arbre syntaxique de la phrase "La fonction entreposage est représentée par des fragments de bouteille de boisson gazeuse en verre américain, un fragment de couvercle ainsi qu'un fragment de pot en verre teinté régulier vert."

L'ajout des chemins syntaxiques entre artefacts et matériaux aux triplets nous donne :

- T1 - "fragments de bouteille" "en" "verre américain"
- T2 - "fragments de bouteille" "fragment fragment en" "verre teinté régulier"
- T3 - Aucun
- T4 - "fragment de couvercle" "fragment en" "verre teinté régulier"
- T5 - Aucun
- T6 - "fragment de pot" "en" "verre teinté régulier"

Ces chemins sont générés en parcourant l'arbre syntaxique depuis le noeud du premier mot trouvé de l'artefact vers le noeud du premier mot trouvé du matériau. Par exemple, pour le chemin menant de l'artefact "fragments de bouteille" au matériau "verre teinté régulier", on constate en parcourant l'arbre que le premier mot trouvé composant cet artefact est "fragments". Afin de rejoindre le noeud du premier mot trouvé composant le matériau, qui est le noeud du mot "verre", on constate qu'il faut passer par les noeuds des mots "fragment", "fragment" (il s'agit du même mot mais à deux emplacements différents dans la phrase et donc deux noeuds différents dans l'arbre syntaxique) puis "en". On obtient donc le chemin composé des mots "fragment fragment en".

Selon nos patrons, nous pouvons déterminer que seuls les triplets T1 et T6 sont valides.

La phrase *La fonction entreposage est représentée par des fragments de bouteille de boisson gazeuse en verre américain, un fragment de couvercle ainsi qu'un fragment de pot en verre teinté régulier vert.* contient donc, selon notre approche, des relations de composition entre les couples artefact-matériau ("fragments de bouteille","verre américain") et ("fragment de pot","verre teinté régulier").

3.6 Évaluation des résultats

Afin d'évaluer la méthode proposée ici, nous avons sélectionné aléatoirement 200 triplets générés par notre méthode : 100 validés par les patrons et 100 non validés. Suite à l'analyse manuelle des triplets, le taux de précision obtenu est de 74% alors que le taux de rappel est de 97%. Le score F1 est d'environ 84%.

L'erreur du taux de précision provient, dans 23 des 26 cas de faux positifs, d'une mauvaise définition de certains artefacts. On trouve par exemple "charbon" dans la liste des artefacts et "bois" dans la liste des matériaux. Chaque apparition de "charbon de bois" dans des textes génère donc un triplet validé par notre méthode. Les trois autres cas de faux positifs proviennent de détection de mots homonymes à des artefacts tel que "pointe" issu de la relation "pointe en terre" où "terre" a bien été identifié comme un matériau. La méthode d'évaluation ne tient pas compte de l'absence potentielle de certains artefacts ou matériaux au sein des listes ni de la possibilité d'avoir une relation de composition s'étendant sur deux phrases dans le texte. Elle permet uniquement d'évaluer l'exactitude du choix des patrons. Pour une meilleure évaluation, il serait donc nécessaire de chercher manuellement dans un rapport archéologique les relations de compositions et de comparer celles-ci aux couples trouvés à l'aide de la méthode proposée ici.

3.7 Peuplement et utilisation de la base de connaissances peuplée

Une fois les couples représentant les relations de composition entre artefacts et matériaux extraits, il nous est possible d'ajouter ceux-ci à la base de connaissances utilisant l'ontologie présentée dans la section 3.2. La Figure 3.5 représente les liens entre différentes instances pour représenter un exemple de relation de composition entre un artefact et un matériau ainsi que l'origine de cette relation. Cette relation représente donc une "pointe" composée de "quartzite" et l'information sur cette relation de composition a été trouvée dans le document "s00213a1973v00_doc001.csv" présentant le site archéologique "Rivière aux Iroquois".

L'utilité de se servir d'une base de connaissances pour stocker l'information est de pouvoir questionner simplement cette information par la suite. Ainsi, si l'on veut connaître "10 sites

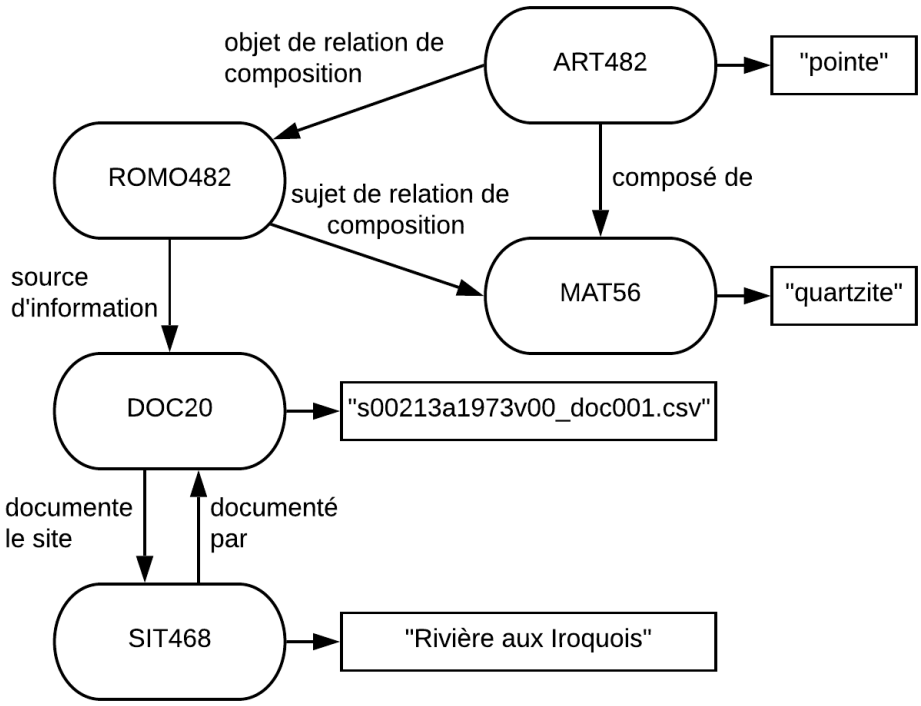

Figure 3.5 Schéma représentant une relation de composition telle que présente dans l'ontologie

dans lesquels on retrouve des pointes faites de chert Onondaga", on peut traduire cette question en SPARQL à l'aide de la requête de la Figure 3.6. Dans cette requête, nous indiquons les entités que nous désirons obtenir en sortie, à savoir ?sitename pour la liste des étiquettes des sites, à l'aide de l'opérateur *select*. Dans le corps de la requête (situé après l'opérateur *where* et entre les accolades), nous extrayons, dans un premier temps, l'ensemble des matériaux présents dans la base de connaissance et conservons uniquement ceux dont l'étiquette contient le mot *onondaga*. Nous extrayons, ensuite, l'ensemble des artefacts contenus dans la base de connaissance ayant un lien de composition avec un des matériaux précédemment extraits. Parmi ces artefacts, nous conservons uniquement ceux dont l'étiquette contient le mot *pointe*. Finalement, nous identifions le document source d'où a été extraite l'information de la relation de composition entre l'artefact et le matériau, identifions le site auquel le document est lié pour enfin récupérer l'étiquette de ce site. Nous ajoutons également l'opérateur *LIMIT* avec une valeur de dix afin de limiter à dix le nombre de résultats sortis.

Cette requête nous renvoie le résultat suivant :

— "Bassin de la rivière Chaudière (site Désy)"

```
select ?sitename where {
    ?mat rdf:type ont:Material;
        rdfs:label ?matlab.
    FILTER( regex(str(?matlab), "onondaga" ))
    ?art ont:object_Made_of ?mat;
        rdfs:label ?artlab;
        ont:object_Made_of_Subject ?romo.
    FILTER(regex(str(?artlab), "pointe" ))
    ?romo ont:source_of_Information ?doc.
    ?doc ont:document_site ?site.
    ?site rdfs:label ?sitename.
}
LIMIT 10
```

Figure 3.6 Exemple de requête SPARQL sur l'ontologie peuplée

— "Lac Aylmer 2"
— "Lac Aylmer 3"
— "Lac aux Araignées 10"
— "Nebessis"
— "Décharge du lac des Joncs 5"
— "Île Morrison"
— "Lac Aylmer 2"
— "Batiscan"
— "Hamel"

3.8 Conclusion

Dans ce chapitre, nous avons pu discuter de la méthode nous ayant permis d'extraire 124 354 relations de composition entre artefacts et matériaux au sein de documents archéologiques. La méthode tire profit des patrons syntaxiques afin de maximiser le nombre d'extractions par rapport à l'utilisation de patrons lexicaux. En effet, pour des exemples tels que celui vu plus haut dans ce mémoire *La fonction entreposage est représentée par des fragments de bouteille de boisson gazeuse en verre américain, un fragment de couvercle ainsi qu'un fragment de pot en verre teinté régulier vert.*, le patron lexical entre l'artefact *fragments de bouteille* et le matériau *verre amérindien* est *de boisson gazeuse en* tandis que le patron syntaxique est *en*. Le

patron syntaxique nous permet ainsi de faire coïncider le patron extrait à la liste des patrons identifiés pour les relations de composition en ne tenant pas compte de l'information supplémentaire *de boisson gazeuse*. La méthode présentée ne permet, cependant, pas de détecter de nouvelles entités, en dehors de celles issues des tableaux. Il est donc nécessaire, pour chaque type d'entité, de posséder une liste des noms d'entités que nous pouvons trouver dans les textes. La méthode proposée nécessite également un *oracle* afin de permettre l'identification des patrons syntaxiques. Dans notre approche, nous avons également considéré une seule relation, identifiée à l'avance. L'intérêt des méthodes d'extractions de relations est cependant de permettre l'extraction de bien plus de relations, sans avoir à les identifier à l'avance. Dans la section suivante de ce mémoire, nous présentons donc une nouvelle méthode adoptée, cette fois-ci sur des documents patrimoniaux du PIMIQ, afin de permettre une extraction plus ample des entités et relations et sans nécessiter de liste d'entités à l'avance ni d'*oracle*.

CHAPITRE 4 EXTRACTION DE RELATIONS AU SEIN DE DOCUMENTS PATRIMONIAUX

4.1 Introduction

Le site Web du Patrimoine immobilier, mobilier et immatériel du Québec (ou PIMIQ) est un répertoire en ligne contenant des descriptions détaillées de différents types de patrimoines au Québec. La Figure 4.1 présente un exemple de fiche tel qu'on le trouve sur le site du PIMIQ. On trouve notamment, sur ces fiches, des descriptions textuelles. Celles-ci sont riches en informations, mais ces informations ne se trouvent sur aucune base de connaissances ou ontologie et restent, de ce fait, inexploitées. L'objectif des travaux réalisés ici est de développer un outil capable d'extraire automatiquement des informations de ces textes et de les ajouter à une base de connaissances afin de permettre, par la suite, de les questionner ou de les réutiliser aisément. À cette fin, les recherches effectuées au sein de notre travail se concentrent sur les fiches du patrimoine immobilier dont nous cherchons à extraire le plus grand nombre d'informations possible.

4.2 Présentation des données disponibles

Les données utilisées dans cette partie des travaux comportent un ensemble de 17 138 extractions de fiches du PIMIQ concernant des patrimoines immobiliers. Ces patrimoines immobiliers se distinguent en deux catégories :

— Des adresses précises, par exemple : *1, rue de l'Église Sud* ;
— Des lieux typés, par exemple : *Bibliothèque de Chibougamau* .

Cette distinction est nécessaire, car, tel que nous le verrons plus tard, il est possible d'extraire le type d'un lieu typé, par exemple : *bibliothèque* dans le cas précédent, mais cela n'est pas possible avec une adresse. Parmi les extractions effectuées, certaines ne contiennent pas suffisamment de texte pour permettre une extraction de relations. Il s'agit de fiches contenant uniquement un titre, une image et parfois quelques données déjà structurées dont le traitement ne représente pas d'intérêt pour nos recherches d'extraction à partir de textes. Le Tableau 4.1 présente les proportions de fiches extraites de PIMIQ contenant suffisamment ou pas assez de texte pour permettre des extractions de relations. Chacune des fiches extraites est séparée en quatre sections, *Description*, *Éléments caractéristiques*, *Informations*

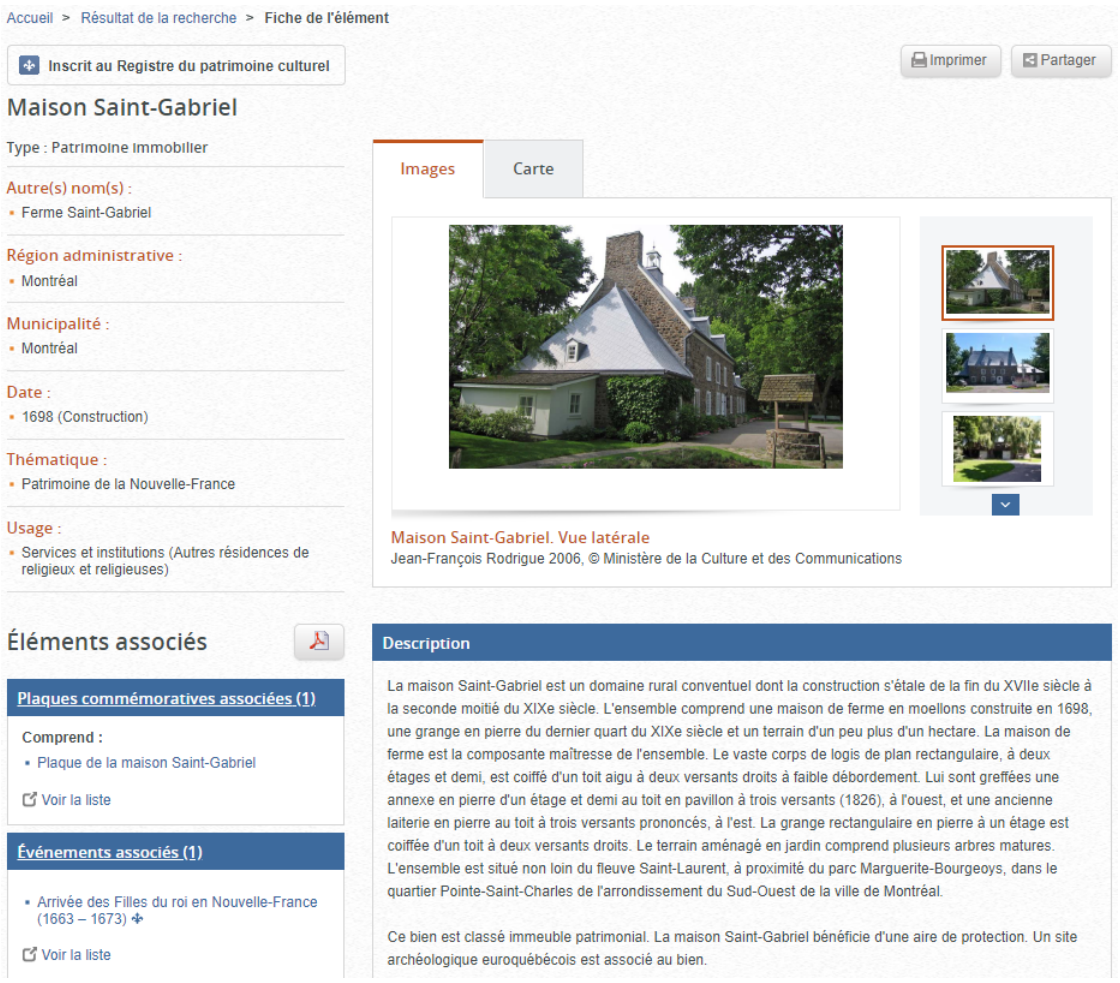

Figure 4.1 Portion de la fiche *Maison Saint-Gabriel* du PIMIQ

historiques et *Valeur patrimoniale*. Chaque section est traitée séparément.

4.3 Présentation générale de la méthode proposée

La méthode proposée ici afin d'extraire des entités et des relations pouvant peupler une base de connaissances consiste à tirer profit d'outils d'extraction ouverte d'informations (ou OIE pour *open information extraction*) déjà existants. Dans cette partie des travaux, nous ne ciblons pas une ou des relations spécifiques à extraire, l'extraction est ouverte et l'identification des relations pertinentes est effectuée par la suite.

Dans un premier temps, nous nous concentrons sur les entités et relations qu'il est possible d'extraire uniquement par l'utilisation d'outils d'extraction ouverte d'informations. La Figure 4.2 présente le pipeline adopté ici. Il est inspiré du pipeline de la Figure 4.3 issue de l'article [3]

Tableau 4.1 Proportion des fiches ne permettant aucune extraction de relation

Type de fiche	Contenant suffisamment de texte pour présenter des possibilités d'extractions	Ne contenant pas suffisamment de texte pour présenter des possibilités d'extractions	Total
Adresses	3 269	8 368	11 637
Lieux typés	4 730	771	5 501
Total	7 999	9 139	17 138

qui présente *PKDE4J*, un système d'extraction automatique d'entités et relations sur un corpus de texte non structuré. Notre pipeline est séparé en deux modules principaux, le module de prétraitement, qui prend en entrée les textes bruts, et le module de traitement, qui prend en entrée la sortie du module précédent et renvoie une liste de triplets qui sont, par la suite, ajoutés à une base de connaissances, comme discuté plus loin dans ce mémoire.

Dans un second temps, nous proposons une méthode permettant l'identification de nouvelles entités afin de permettre l'augmentation du nombre de relations convenablement extraites à l'aide d'un modèle d'apprentissage supervisé.

Finalement, nous utilisons l'ensemble des triplets ⟨*sujet, prédicat, objet*⟩ extraits afin de peupler une base de connaissances créée manuellement, basée sur les relations les plus fréquentes. La Figure 4.4 présente le pipeline final contenant, contrairement au pipeline Figure 4.2, notre modèle d'apprentissage supervisé. Nous y avons également intégré un module de peuplement permettant d'ajouter à la base de connaissances les entités et les relations extraites.

4.4 Description détaillée des différents modules

Dans cette section, nous décrivons de manière détaillée chacun des modules du pipeline décrit à la Figure 4.2. Un exemple d'extraction d'une section de fiche patrimoniale est également présenté afin de démontrer leur fonctionnement.

L'exemple décrit dans ce chapitre est une portion de la section *informations historiques* de la fiche patrimoniale *Pont de Des Rivières* :

Le pont présente une structure de type Howe. En 1840, l'inventeur américain William Howe (1803-1852) obtient un brevet pour une amélioration de la ferme Long, caractérisée par ses éléments de compression disposés en croix de Saint-André. (...) Vers la fin du XIXe siècle,

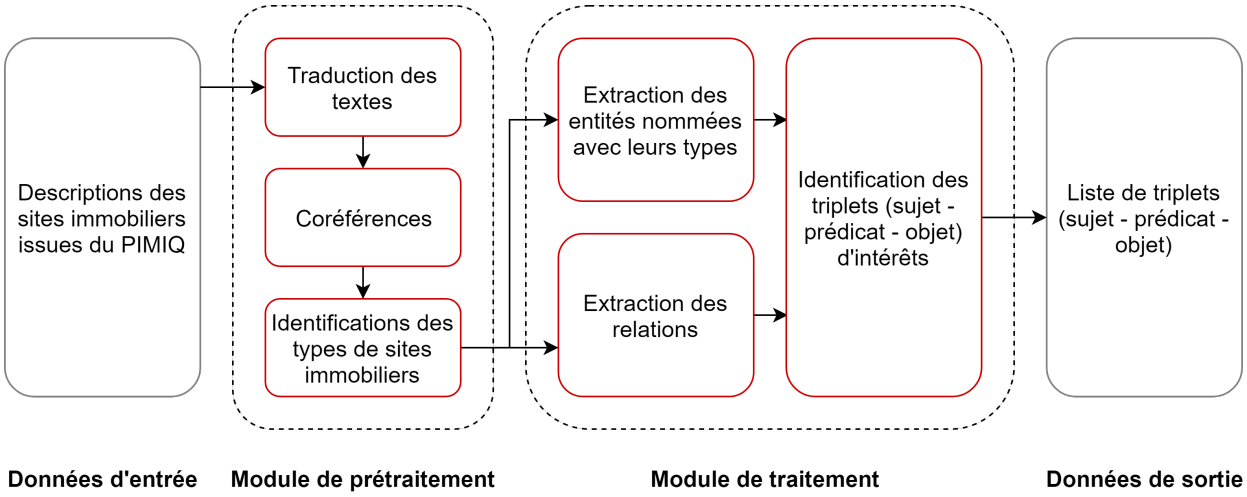

Figure 4.2 Pipeline de la méthode proposée pour l'extraction de triplets au sein de documents patrimoniaux

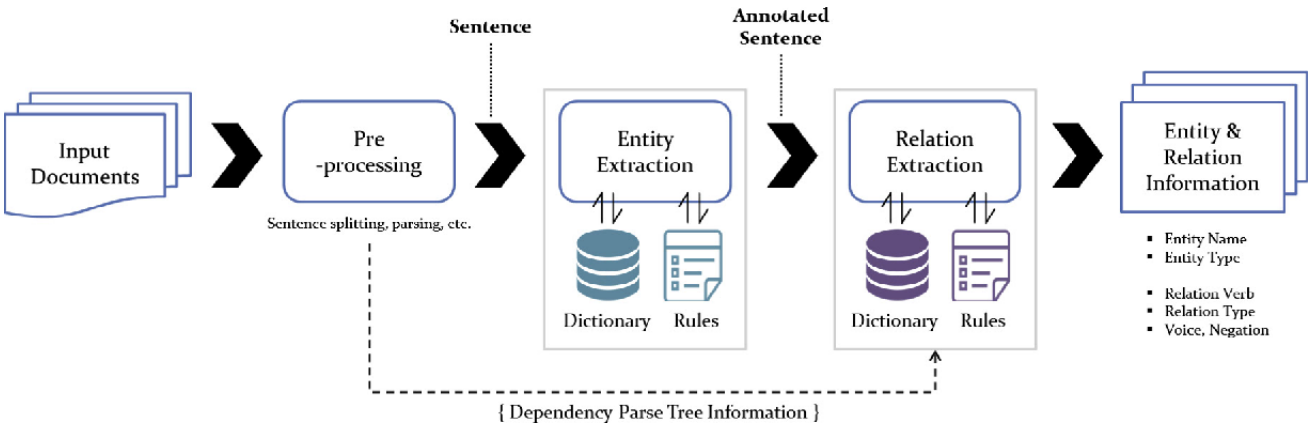

Figure 4.3 Architecture du système PKDE4J. Extrait de [3]

le hameau de Malmaison est peu à peu délaissé. Son territoire est inclus dans la municipalité de Notre-Dame-de-Stanbridge, érigée en 1889.

4.4.1 Module de prétraitement

Le module de prétraitement a pour objectif de mettre en forme le texte afin de maximiser l'efficacité du module de traitement. Celui-ci est composé de trois sous-modules. Tout d'abord, un module de traduction prend en entrée le texte brut en français pour donner en sortie l'équivalent anglais. Ensuite, un module de résolution de coréférences utilise ce texte anglais comme entrée puis renvoie en sortie un nouveau texte dont chaque référence (par exemple les anaphores) à une entité est remplacée par l'entité elle-même. Finalement, un

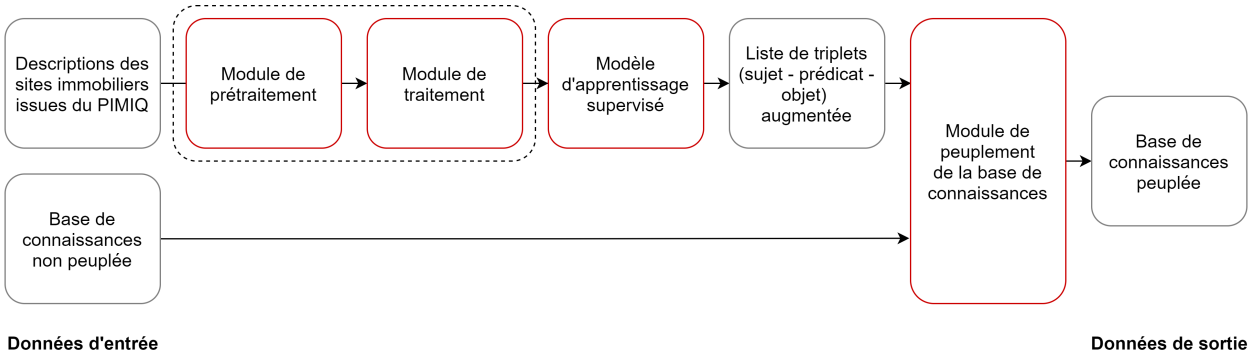

Figure 4.4 Pipeline global avec module d'amélioration de l'identification des entités

module d'identification des types de sites immobiliers est ajouté pour compléter le module de résolution de coréférence, puis sa sortie est envoyée au module de traitement. La suite de ce chapitre décrit le fonctionnement et l'intérêt de chacun de ces sous-modules.

Traduction des textes

La première étape du prétraitement des textes consiste en une traduction du français à l'anglais. Cette traduction est nécessaire, car plusieurs outils de *Stanford NLP*, que nous désirons utiliser par la suite (à savoir, l'outil de coréférences et le NER), ne supportent pas la langue française, mais uniquement la langue anglaise, tel que présenté sur le tableau de la Figure 4.5. Afin de traduire les textes du français à l'anglais, nous utilisons l'API de traduction de *Google*.

Annotator	Arabic	Chinese	English	French	German
Tokenize	✓	✓	✓	✓	✓
Sent. split	✓	✓	✓	✓	✓
Truecase			✓		
POS	✓	✓	✓	✓	✓
Lemma			✓		
Gender			✓		
NER		✓	✓		✓
RegexNER	✓	✓	✓	✓	✓
Parse	✓	✓	✓	✓	✓
Dep. Parse		✓	✓		
Sentiment			✓		
Coref.			✓		

Figure 4.5 Schéma indiquant les langues supportées pour chaque outil de Stanford CoreNLP. Extrait de [4]

La sortie, suite au passage par le module de traduction, de notre exemple est :

The bridge has a Howe type structure. In 1840, the American inventor William Howe (1803-1852) obtained a patent for an improvement of the Long farm, characterized by its compression elements arranged in a cross of St. Andrew. (...) Towards the end of the 19th century, the hamlet of Malmaison was gradually abandoned. Its territory is included in the municipality of Notre-Dame-de-Stanbridge, erected in 1889.

Ajout des coréférences

Le deuxième sous-module du module de prétraitement consiste en un module de résolution des coréférences présentes dans un texte. L'objectif de ce module est d'éliminer les pronoms personnels tels que *He*, *She*, et *It* ainsi que les pronoms possessifs tels que *His*, *Her*, et *Its* et de les remplacer par les entités auxquelles ils réfèrent. L'intérêt de ce module est de faciliter, par la suite, la détection d'entités et de relations. Ainsi, si le texte *Victor Hugo is an author. He wrote Les Misérables.* est directement envoyé au module de traitement, la sortie de l'extraction de relations serait, théoriquement, la liste des deux relations ⟨*Victor Hugo*, *is*, *author*⟩ et ⟨*He*, *wrote*, *Les Misérables*⟩. Dans ce cas là, la seconde relation extraite est futile, *He* ne pouvant pas être lié à une personne. Le passage de ce texte par le module de coréférences donnerait le texte *Victor Hugo is an author. Victor Hugo wrote Les Misérables.*, détectant que *He* est une référence à *Victor Hugo*. La sortie du module d'extraction des relations serait donc maintenant, théoriquement, la liste des deux relations ⟨*Victor Hugo*, *is*, *author*⟩ et ⟨*Victor Hugo*, *wrote*, *Les Misérables*⟩, permettant de faire le lien entre *Victor Hugo* et *Les Misérables*.

L'outil utilisé afin de déceler les coréférences au sein d'un texte est celui de *Stanford NLP*. Celui-ci permet d'identifier les références d'une entité que nous remplaçons ensuite dans le texte par celle-ci.

La sortie, suite au passage par le module de coréférences, de notre exemple est :

The bridge has a Howe type structure. In 1840, the American inventor William Howe (1803-1852) obtained a patent for an improvement of the Long farm, characterized by its compression elements arranged in a cross of St. Andrew. (...) Towards the end of the 19th century, the hamlet of Malmaison was gradually abandoned. the hamlet of Malmaison's territory is

included in the municipality of Notre-Dame-de-Stanbridge, erected in 1889.

Le module de coréférence permet, ici, d'identifier le lien entre *the hamlet of Malmaison* et *Its territory* permettant ainsi de remplacer *Its* par *the hamlet of Malmaison* afin de faciliter la suite du traitement. On peut remarquer que dans ce cas-ci, l'outil d'ajout des coréférences a manqué l'identification de la coréférence du premier pronom possessif *its*.

Identification des types de sites immobiliers

L'outil de coréférence de *Stanford NLP* permet d'optimiser la détection des entités. Cependant, cet outil reste incomplet. Dans notre exemple, on pourrait facilement s'apercevoir que *The bridge* réfère à *Pont de Des Rivières* ou plus précisément *Bridge of Des Rivières*. L'outil de coréférence de *Stanford NLP* ne permet pas de faire ce lien, car *Bridge of Des Rivières* n'apparaît jamais littéralement dans le texte, mais seulement dans le titre de la fiche. Le module d'identification des mentions de sites immobiliers permet de remédier à ce problème.

Suite à une étude des textes, nous avons remarqué que les sites immobiliers sont souvent référencés dans les textes à l'aide de leur type (par exemple : *The presbytery* pour *Presbytery of Saint-Adelphe* ; *The house* pour *House Saint-Gabriel* ou *The bridge* pour *Bridge of Des Rivières*). Identifier le type d'un site immobilier s'avère donc être nécessaire afin d'identifier ses mentions dans le texte. Pour cela, nous utilisons l'ontologie *schema.org*[1], travail collaboratif de *Google*, *Microsoft*, *Yahoo* et *Yandex* ayant pour objectif de créer un vocabulaire commun pour les données structurées du Web [26], et qui comporte de nombreuses classes associées à des types de bâtiments. On peut, en effet, y trouver des classes telles que *Church* ou *School*. Ces classes sont regroupées sous les superclasses *Place* et *Organization*.

L'approche adoptée ici afin d'identifier les différents types possibles de bâtiments est d'extraire de schema.org, à l'aide d'une requête SPARQL, toutes les sous-classes, récursivement, des classes *Place* et *Organization*. Cette extraction nous donne une liste de types possibles de bâtiments L_{types}.

Une fois la liste des différents types possibles de bâtiments générée, nous pouvons l'utiliser afin de tenter d'identifier le type de chacun des 5 501 sites typés. Pour cela, nous comparons la chaîne de caractères de chacun des noms de site à celle de chacun des types de L_{types} et, si

1. https ://schema.org/

la chaîne de caractères d'un des types est incluse dans celle du nom d'un site, nous pouvons supposer que ce site est de ce type. Nous avons donc :

$$\langle site, is, type \rangle \iff type \subset site, type \in L_{types} \tag{4.1}$$

À l'aide de cette méthode, nous avons pu extraire 2 853 types de site parmi les 5 501 fiches. Suite à une évaluation manuelle effectuée en sélectionnant 100 de ces sites aléatoirement, nous obtenons une précision de 89%. L'analyse manuelle de l'erreur a montré que celle-ci provient généralement de la mauvaise identification d'une sous-chaîne de caractères du site. Par exemple : le site *Chapel of the third cemetery of Sainte-Anne-de-Beaupré* devrait être de type *Chapel*. Cependant, notre système identifie son type, à tort, comme *cemetery*, dont la chaîne de caractères se trouve dans le titre du site. De plus, *Chapel* n'est pas un type existant dans la taxonomie de *schema.org*.

Au total, nous avons trouvé 48 différents types de patrimoines immobiliers à l'aide des classes de la taxonomie de *schema.org*. Une liste de ces types est présente dans l'annexe C. Nous avons également pu lier nos fiches à la taxonomie de *schema.org* en considérant les 2 853 fiches dont nous avons pu identifier le type comme des instances des classes de *schema.org*.

Suite au passage par le module d'identification des types de sites immobiliers, la sortie de notre exemple est le triplet :

⟨*Pont de Des Rivières* , *type* , *Bridge*⟩

Le module a donc bien identifié que *Pont de Des Rivières* est un site de type *Bridge* (ou *Pont* en français.)

4.4.2 Module de traitement

Le module de traitement a pour objectif de prendre en entrée les textes prétraités, d'en extraire les entités et les relations entre celles-ci et de les filtrer afin de conserver uniquement celles pouvant être utilisées dans le peuplement d'une base de connaissances. Ce module a donc pour entrée la sortie du module de prétraitement et renvoie en sortie une liste d'entités ainsi qu'une liste de triplets ⟨*Sujet,Prédicat,Objet*⟩ entre celles-ci.

Extraction des relations

Dans le but d'extraire les relations présentes au sein des textes, nous avons considéré et testé trois différents outils d'extraction ouverte de relations. Les outils considérés sont *StanfordOIE* [21] (issu de la suite *Stanford NLP [4]*), *OllIE* [11] et *MinIE* [14]. Afin d'évaluer l'outil le plus performant pour l'application sur notre corpus, nous avons considéré six textes issus des sections *Informations historiques* (pour leur contenu riche en entités et relations intéressantes) de fiches patrimoniales du *PIMIQ* sélectionnées aléatoirement et avons évalué, pour chaque outil, le nombre de triplets distincts, exacts et d'intérêts identifiés. On considère un triplet comme exact lorsque le sujet et l'objet du triplet représentent bien des entités (par exemple *municipality* est une entité correcte tandis que *municipality erected in 1889* ne l'est pas) et le prédicat représente bien une relation entre ces entités. On considère un triplet comme étant d'intérêt lorsque celui-ci peut être utilisé sans traitement supplémentaire dans un cadre de développement ontologique (par exemple, le triplet ⟨*William Howe , is , inventor*⟩ peut aisément être ajouté à une base de connaissances, contrairement au triplet ⟨*American inventor William Howe , obtained , patent for improvement of Long farm*⟩.) Le Tableau 4.2 présente les résultats de cette analyse.

Tableau 4.2 Nombre de triplets distincts, exacts et d'intérêts identifiés selon chaque outil pour différentes fiches patrimoniales

Nom de la fiche patrimoniale	StanfordOIE	OllIE	MinIE
Domaine Beauséjour	**14**	5	9
90, Rang 1 Neigette Est	**5**	3	4
Ancien magasin général de la mine Johnson	**11**	7	5
605, avenue Royale	**8**	6	1
Ancien couvent Notre-Dame-de-la-Merci	**18**	10	12
17-25, rue des Remparts	**14**	8	6

Bien que l'outil *StanfordOIE* [21] ne soit pas le plus performant selon certaines études [16] [17], c'est avec celui-ci que nous obtenons le plus de triplets utilisables. De plus, cet outil provient de la suite *Stanford NLP* [4], qui présente d'autres outils utilisés dans le cadre de ce mémoire, à savoir, le résolveur de coréférences discuté dans la section 4.4.1 ainsi que l'extracteur d'entités nommées discuté dans la section suivante. La suite des travaux effectués repose donc sur l'utilisation de cet outil. Le nombre de relations, exactes ou non, extraites par cet ou-

til, est cependant très important, la précision de celui-ci étant faible (il extrait un grand nombre de triplets dans la plupart des textes mais seul un petit pourcentage de ceux-ci sont distincts, exacts et d'intérêts). Les étapes de traitement des relations extraites permettent de conserver uniquement, parmi ce grand nombre de relations, certaines exactes et utilisables.

À l'aide de l'outil *StanfordOIE* [21], nous avons pu extraire un total de 256 409 triplets ⟨*Sujet,Prédicat,Objet*⟩ distincts, soit une moyenne d'environ 32 triplets par fiche patrimoniale. Ces triplets ne sont pas, pour la plupart, directement ajoutables à une base de connaissances car inexacts ou non pertinents. L'objectif des modules suivants est de permettre un filtrage de ces triplets afin de conserver uniquement des triplets corrects.

Suite au passage par le module d'extraction des relations, la sortie de notre exemple est la liste de relations :

(bridge , has , structure)
(bridge , has , structure like Howe)
(American inventor William Howe , obtained , patent for improvement of farm)
(American inventor William Howe , obtained , patent for improvement of farm characterized by its compression elements arranged)
(American inventor William Howe , obtained , patent for improvement of Long farm characterized)
(American inventor William Howe , obtained , patent for improvement of farm characterized)
(William Howe , is , inventor)
(American inventor William Howe , obtained , patent for improvement of farm characterized by its compression elements arranged in cross of St. Andrew)
(American inventor William Howe , obtained , patent for improvement of Long farm)
(American inventor William Howe , obtained , patent for improvement characterized by its compression elements)
(American inventor William Howe , obtained patent In , 1840)
(American inventor William Howe , obtained , patent)
(American inventor William Howe , obtained , patent for improvement)
(American inventor William Howe , obtained , patent for improvement of farm characterized by its compression elements arranged in cross)
(American inventor William Howe , obtained , patent for improvement characterized by its compression elements arranged in cross)
(American inventor William Howe , obtained , patent for improvement of Long farm charac-

terized by its compression elements)
(William Howe , is , American)
(American inventor William Howe , obtained , patent for improvement of Long farm characterized by its compression elements arranged in cross of St. Andrew)
(American inventor William Howe , obtained , patent for improvement of farm characterized by its compression elements arranged in cross of St. Andrew)
(hamlet , is , abandoned)
(hamlet , is abandoned In , nineteenth century)
(hamlet , is abandoned In , late nineteenth century)
(hamlet , is abandoned In , late century)
(hamlet , is gradually abandoned In , century)
(hamlet , is gradually abandoned In , nineteenth century)
(hamlet , is abandoned In , century)
(hamlet , is gradually abandoned In , late century)
(hamlet , is , gradually abandoned)
(hamlet , is gradually abandoned In , late nineteenth century)
(hamlet of Malmaison territory , is included in , municipality)
(hamlet of Malmaison territory , is included in , municipality erected)
(hamlet of Malmaison territory , is included in , municipality of Notre-Dame-de-Stanbridge)
(hamlet of Malmaison territory , is included in , municipality erected in 1889)
(hamlet of Malmaison territory , is , included)
(hamlet of Malmaison territory , is included in , municipality of Notre-Dame-de-Stanbridge erected)
(hamlet of Malmaison territory , is included in , municipality of Notre-Dame-de-Stanbridge erected in 1889)

Dans cet exemple, on note que 36 relations ont pu être extraites. Celles-ci concernent essentiellement quatre entités, à savoir, *bridge*, *William Howe* avec l'ajout de son titre et sa nationalité, *hamlet* et *hamlet of Malmaison territory*. Parmi les relations extraites, seules 6 sont intéressantes :

(William Howe , is , inventor)
(American inventor William Howe , obtained , patent)
(American inventor William Howe , obtained patent In , 1840)
(William Howe , is , American)
(hamlet , is abandoned In , nineteenth century)

(hamlet of Malmaison territory , is included in , municipality of Notre-Dame-de-Stanbridge)

La majorité des autres relations extraites contiennent des relations trop complexes.

Extraction des types d'entités nommées

Le module d'extraction des types d'entités nommées a pour objectif d'identifier le type du plus grand nombre possible d'entités au sein des textes. Cette tâche est nécessaire car elle permet d'identifier la classe à laquelle doit être liée une instance (représentant une entité extraite.) L'outil utilisé pour l'extraction des types d'entités est l'outil *StanfordNER* de la suite *StanfordNLP* [4]. Nous avons pu identifier le type d'un total de 22 116 entités distinctes parmi 20 types (*Person*, *Country*, *Organization*, *Set*, *Number*, *Title*, *Misc*, *Date*, *Location*, *City*, *Duration*, *Cause_of_death*, *Ideology*, *Ordinal*, *Religion*, *State_or_province*, *Nationality*, *Time*, *Criminal_charge* et *Money*). À cette liste, nous ajoutons également le type *Patimmo* pour représenter les patrimoines immobiliers dont nous avons pu extraire une liste à l'aide du module d'identification des types de sites immobiliers. Les entités identifiées comme de type *Patimmo* sont l'ensemble des noms de site patrimoniaux (par exemple : *Bridge of Des Rivières*) ainsi que l'ensemble des différents types extraits de *schema.org* (par exemple : *Bridge*) de la liste L_{types}.

Suite au passage par le module d'extraction des types d'entités nommées, la sortie pour notre exemple, est la liste d'entités typées :

(Howe , PERSON)
(1840 , DATE)
(American , NATIONALITY)
(inventor , TITLE)
(William Howe , PERSON)
(1803-1852 , DURATION)
(the late nineteenth century , DATE)
(Malmaison , LOCATION)
(Notre-Dame-de-Stanbridge , LOCATION)
(1889 , DATE)
(Bridge , PATIMMO)
(Bridge of Des Rivières , PATIMMO)

Dans cet exemple, les deux dernières entités ne proviennent pas du texte mais du module

d'identification des types de sites immobiliers.

Extraction des triplets d'intérêt

Le module d'extraction des triplets d'intérêt a pour rôle d'identifier les triplets pouvant être ajoutés dans une base de connaissances par la suite. Il prend en entrée les triplets ⟨*Sujet, Prédicat, Objet*⟩ sortant du module d'extraction de relations ainsi que la liste des entités typées sortant du module d'extraction des types d'entités nommées et associe à chaque sujet et objet de chaque triplet son type, si l'entité se trouve dans la liste des entités typées, le type *Undef* sinon. Le module compare ensuite le triplet ⟨*Type du sujet, Prédicat, Type de l'objet*⟩ composé du prédicat de la relation ainsi que du type du sujet et de l'objet à une liste $L_{accepted}$ de triplets acceptés et ajoute le triplet ⟨*Sujet (Type du sujet), Prédicat, Objet (Type de l'objet)*⟩ à la liste de sortie s'il y appartient. Cette liste $L_{accepted}$ de triplets acceptés par ce module ainsi que leur signification est représentée dans le Tableau 4.3. Cette liste a été générée manuellement par l'étude des fréquences de chaque triplet ⟨*Type du sujet, Prédicat, Type de l'objet*⟩. Nous avons intégré dans cette liste une partie des triplets les plus fréquents et révélateurs. Par exemple, nous considérons les triplets ⟨*PERSON, is, NATIONALITY*⟩, ⟨*PERSON, built, PATIMMO*⟩ ou encore ⟨*PATIMMO, was built in, DATE*⟩ comme révélateurs car la signification de l'information qu'ils transportent est aisément identifiable. Dans le cadre de ces recherches, nous nous sommes limités à dix relations parmi les plus fréquentes, des travaux futurs portés sur ce projet pourraient aisément identifier de nouvelles relations à ajouter.

Suite au passage par le module d'extraction des triplets d'intérêts, la sortie de notre exemple est la liste de triplets :

⟨*William Howe (Person)* , *is* , *American (Nationality)*⟩
⟨*William Howe (Person)* , *is* , *inventor (Title)*⟩

Suite au passage par le module d'extraction des triplets d'intérêts de l'ensemble de nos triplets, 6 542 sont acceptés.

Tableau 4.3 Liste des triplets acceptés selon le prédicat et le type du sujet et de l'objet

Type du sujet	Prédicat(s)	Type de l'objet	Signification de la relation
PERSON	is/is of/is by/is to	NATIONALITY	Nationalité de la personne
PERSON	is/is of/is by/is to	TITLE	Profession de la personne
PERSON	built	PATIMMO	Personne ayant bâtis le patrimoine immobilier
PATIMMO	was built by	PERSON	Personne ayant bâtis le patrimoine immobilier
PATIMMO	was built in	DATE	Date de construction du patrimoine immobilier
PATIMMO	was built around	DATE	Date approximative de construction du patrimoine immobilier
PATIMMO	was built for	PERSON	Personne pour qui a été bâtis le patrimoine immobilier
PATIMMO	was sold by	PERSON	Personne ayant vendu le patrimoine immobilier
PERSON	sold	PATIMMO	Personne ayant vendu le patrimoine immobilier
PATIMMO	was sold to	PERSON	Personne à qui a été vendu le patrimoine immobilier

4.5 Identification de nouvelles entités

4.5.1 Motivations

La méthode présentée précédemment permet d'extraire des entités et des triplets (à l'aide d'outils d'extraction ouverte d'entités et de relations et par traitement et filtrage des sorties de ces outils) pouvant être, par la suite, utilisés lors du peuplement de bases de connaissances. Les entités extraites peuvent être de différents types tels que *Patimmo* pour représenter un patrimoine immobilier et *Person* pour représenter une personne. Parmi les relations extraites entre ces deux types d'entités, on retrouve des prédicats tels que *was built by*, *built*, *was built for*, *was sold to* et bien d'autres. On peut cependant remarquer, en analysant l'ensemble des triplets liés à ces relations qu'un nombre important d'entités n'a pas pu être identifié convenablement (tel que présenté dans le Tableau 4.4). En effet, si l'on considère, par exemple, les triplets comprenant la relation *was built by* et dont le sujet est de type *PATIMMO*, on peut constater que, dans 39 cas sur les 119, le type de l'objet n'a pas pu être identifié automatiquement par l'outil de reconnaissance des entités nommées. Une identification manuelle a cependant confirmé qu'il s'agissait tout de même de personnes. L'objectif des travaux présen-

tés dans cette section est donc de permettre l'extraction d'un plus grand nombre de triplets dont le sujet et l'onjet sont typés par l'amélioration de l'outil de reconnaissance automatique des entités.

Tableau 4.4 Statistiques pour l'identification du type *Person* pour certaines relations extraites avec StanfordOIE

Type du sujet	Prédicat	Type de l'objet	Position considérée	Nombre total d'occurrences	Nombre d'instances où l'entité est effectivement une personne
PATIMMO	was built by	PERSON	objet	130	127 (98%)
PATIMMO	was built by	UNDEF	objet	119	**39** (33%)
PERSON	built	PATIMMO	sujet	83	76 (92%)
UNDEF	built	PATIMMO	sujet	52	**27** (52%)
PATIMMO	was built for	PERSON	objet	57	57 (100%)
PATIMMO	was built for	UNDEF	objet	35	**13** (37%)
PATIMMO	was sold to	PERSON	objet	28	26 (93%)
PATIMMO	was sold to	UNDEF	objet	47	**18** (38%)

4.5.2 Approche proposée

L'approche proposée ici consiste à utiliser l'ensemble des entités dont le type a pu être identifié par un outil d'extraction des entités nommées afin d'entraîner un modèle d'apprentissage supervisé dont l'objectif est de permettre l'identification du type de nouvelles entités ou d'entités dont le type n'a pas pu être identifié par les outils d'extraction des entités nommées. Dans cette section, nous présentons le modèle utilisé en détail avec le choix des paramètres et hyper-paramètres ainsi que la méthode utilisée pour l'apprentissage du modèle.

Modèle et paramètres

Le modèle d'apprentissage supervisé utilisé ici est un réseau de neurones généré à l'aide de l'outil *scikit-learn* [29]. Les paramètres utilisés en entrée sont séparés en 3 cartes :

— La carte typographique (4 entrées entières) : La carte typographique d'une entité contient une liste de ses caractéristiques typographiques, à savoir, le nombre de mots composants l'entité, le nombre de caractères, le nombre de majuscules ainsi que le nombre de chiffres ;

— La carte des types des sous-séquences (21 entrées booléennes) : L'outil *StanfordNER* permet d'identifier le type d'un certain nombre d'entités parmi 21 types distincts. Pour

chaque type, on recherche parmi les sous-séquences de mots si l'une d'elles correspond à l'un des types (pour l'exemple "*American inventor William Howe*", on aurait donc "1" pour les types "*Nationality*", "*Title*" et "*Person*" correspondant respectivement aux segments "*American*", "*inventor*" et "*William Howe*" qui ont été correctement identifiés par l'outil *StanfordNER*, et "0" pour les autres types) ;

— La carte relationnelle (<=2N entrées booléennes) : Cette carte permet de prendre en compte les relations avec lesquelles l'entité est utilisée dans tout le corpus de textes ainsi que la position de l'entité (sujet ou objet). Pour identifier ces relations, on utilise les sorties obtenues par l'outil *StanfordOIE*. Ces entrées dépendent du type d'entité que l'on désire identifier. Afin de générer les entrées pour un type d'entité, on extrait les entités déjà identifiées de ce type, on identifie ensuite, pour chacune, les relations dans lesquelles elles sont utilisées ainsi que la position de l'entité dans la relation (par exemple ⟨*Object - was built by*⟩). On conserve ensuite uniquement les N couples ⟨*position - relation*⟩ les plus fréquents. Afin de faciliter l'identification des entités négatives (dont le type ne correspond pas au type recherché), on ajoute également les N couples les plus fréquents des entités dont le type a pu être identifié par l'outil *StanfordNER* mais est différent du type considéré. Certains couples pouvant se recouper entre les types (par exemple, le couple ⟨*Subject - is*⟩ est très fréquent quel que soit le type d'entité considérée), on obtient un nombre inférieur ou égal à deux fois N le nombre de couples considérés. À notre connaissance, l'utilisation d'une telle carte relationnelle pour aider l'identification du type d'une entité est une idée originale de notre approche qui n'a pas encore été effectué dans d'autres travaux.

La sortie du modèle est un unique booléen indiquant si l'entité est du type considéré ou non.

La liste des dix relations les plus fréquentes pour une entité de type *PERSON* est :
⟨*Subject - is*⟩
⟨*Subject - has*⟩
⟨*Subject - is of*⟩
⟨*Subject - is by*⟩
⟨*Subject - is to*⟩
⟨*Subject - of*⟩
⟨*Subject - was*⟩
⟨*Subject - built*⟩
⟨*Object - belongs to*⟩
⟨*Subject - is 's*⟩

La liste des dix relations les plus fréquentes pour un autre type d'entité est :
⟨*Subject - is*⟩
⟨*Subject - was*⟩
⟨*Object - is in*⟩
⟨*Subject - has*⟩
⟨*Subject - is in*⟩
⟨*Object - was built in*⟩
⟨*Subject - was built in*⟩
⟨*Object - is*⟩
⟨*Subject - is of*⟩
⟨*Subject - of*⟩

On a donc un recoupement pour les relations :
⟨*Subject - is*⟩
⟨*Subject - has*⟩
⟨*Subject - is of*⟩
⟨*Subject - of*⟩
⟨*Subject - was*⟩

Ces relations sont conservées parmi les entrées du modèle afin de laisser la possibilité au modèle d'apprendre, par lui-même, l'influence de celles-ci sur l'identification du type d'une entité.

Le tableau Tableau 4.5 présente deux exemples d'entrées et sorties attendues du modèle pour un exemple positif et un exemple négatif dans l'identification d'une entité de type *Person*. Les 16 entrées non indiquées de la carte des types de sous-séquences ont toutes pour valeur "0" pour les deux exemples.

Tableau 4.5 Exemple des entrées et sorties attendues dans le réseau de neurones de deux éléments avec N = 10 pour la classification d'une entité de type *Person*

			"François Fortier"	"May 1979"
Entrées	Carte typographique (4 entiers)	Nombre de mots	2	2
		Nombre de caractères	16	8
		Nombre de majuscules	2	1
		Nombre de chiffres	0	4
	Carte des types de sous-séquences (21 booléens)	PATIMMO	0	0
		DATE	0	1
		MISC	0	0
		PERSON	1	0
		TITLE	0	0
		...		
	Carte relationnelle (15 booléens)	⟨Subject - is⟩	1	0
		⟨Subject - has⟩	0	0
		⟨Subject - is of⟩	0	0
		⟨Subject - is by⟩	1	0
		⟨Subject - is to⟩	0	0
		⟨Subject - of⟩	0	0
		⟨Subject - was⟩	0	0
		⟨Subject - built⟩	1	0
		⟨Object - belongs to⟩	0	0
		⟨Subject - is 's⟩	0	0
		⟨Object - is in⟩	0	0
		⟨Subject - is in⟩	0	0
		⟨Object - was built in⟩	0	1
		⟨Subject - was built in⟩	0	0
		⟨Object - is⟩	0	0
Sortie attendue (1 booléen)			1	0

Apprentissage et validation croisée

La méthode d'apprentissage proposée consiste à tirer profit des entités dont le type a déjà été convenablement identifié par l'outil *StanfordNER*. À l'aide de celles-ci, nous générons deux listes d'ensemble d'apprentissages, $L_{positifs}$ et $L_{négatifs}$. $L_{positifs}$ correspond à la liste des entités dont le type est le même que celui considéré (par exemple *Person* pour un modèle tentant d'identifier les entités de type *Person*) et $L_{négatifs}$ correspond à la liste des entités dont le type est distinct et ne se recoupe pas avec le type considéré. Pour chaque entité $E_i \in L_{positifs}$ et $E_i \in L_{négatifs}$, on génère la liste des paramètres E_i^x correspondant à la liste de ses valeurs pour chacun des éléments des trois cartes d'entrée décrites précédemment. On génère également la valeur de sortie de chaque entité de la manière suivante :

$$\begin{cases} E_i^y = 1 \iff E_i \in L_{positifs} \\ E_i^y = 0 \iff E_i \in L_{négatifs} \end{cases} \tag{4.2}$$

La liste de nos ensembles est ensuite générée par :

$$L_{ensembles} = L_{positifs} \cup L_{négatifs} \tag{4.3}$$

Cette liste est ensuite mélangée aléatoirement et séparée en un ensemble d'apprentissage et un ensemble de test.

$$\begin{cases} L_{apprentissage} = 85\% \times L_{ensembles} \\ L_{test} = 15\% \times L_{ensembles} = L_{ensembles} - L_{apprentissage} \end{cases} \tag{4.4}$$

La liste des ensembles d'apprentissages $L_{apprentissage}$ est utilisée, dans un premier temps, dans une validation croisée afin d'évaluer les paramètres et hyper-paramètres du modèle. Dans un second temps, cette liste est utilisée au complet afin d'entraîner le modèle avant son utilisation sur la liste des ensembles de tests L_{test}, puis sur de nouvelles données.

Les tableaux Tableau 4.6, Tableau 4.7 et Tableau 4.8 présentent les résultats de la validation croisée à 10 plis de différentes combinaisons de paramètres et d'hyper-paramètres afin de trouver la combinaison idéale obtenant le meilleur score F1 moyen.

Tout d'abord, le tableau Tableau 4.6 présente l'évaluation avec variation des entrées utilisées. Chaque colonne consiste en un réseau de neurones (RN) avec des entrées données. On y analyse l'impact de la présence ou de l'absence de chaque carte d'entrée ainsi que l'impact de la variation du nombre de relations utilisées pour la carte relationnelle. Dans cette analyse,

le nombre de couches cachées est fixé à deux, contenant, par ordre de profondeur, 50 et 30 neurones. La fonction d'activation est également fixée à l'utilisation de la fonction sigmoïde. On peut constater dans cette étude que l'utilisation de chaque carte a un impact positif sur le score F1 final obtenu. De plus, le score F1 final croît également avec l'augmentation du nombre de relations utilisées pour la fiche relationnelle tel que le montre le graphique de la figure Figure 4.6. Bien que le score F1 soit plus élevé avec 500 relations en utilisant la carte relationnelle seule, le score moyen lors de l'utilisation avec l'ensemble des cartes est plus faible dû à un sur-apprentissage. Pour la suite des travaux, nous avons donc fixé le nombre de relations utilisées dans la carte relationnelle à 100 et nous utilisons l'ensemble des cartes.

Dans l'étude des résultats du Tableau 4.6, on peut également constater que l'utilisation seule de la carte relationnelle comme entrée du modèle offre un rappel faible, quel que soit le nombre de relations considérées. Suite à une analyse des entités $E_i \in L_{positifs}$, nous avons remarqué que la valeur moyenne du nombre de relations utilisées par une entité de type *Person* est de 2,781, 46% de ces entités n'utilisant qu'une seule relation. Parmi ces entités, nous avons également remarqué que, dans 83% des cas, cette relation n'est pas déterminante du type de l'entité car elle n'est simplement pas présente dans la carte relationnelle ou bien elle est présente en fréquence similaire aussi bien pour des entités de type *Person* que pour des entités d'autres types. L'utilisation de la carte relationnelle seule ne nous permet donc pas, dans un grand nombre de cas, de déterminer si une entité est de type *Person* ou non.

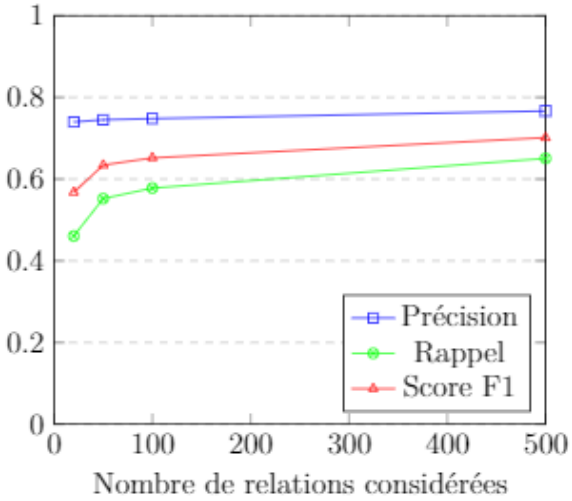

Figure 4.6 Précision, rappel et score F1 en fonction du nombre de relations considérées

Le tableau Tableau 4.7 présente l'évaluation de l'impact du nombre de couches cachées et de neurones par couche utilisée dans le modèle sur le score F1. Dans cette analyse, les entrées

Tableau 4.6 Évaluation de réseaux de neurones selon les cartes utilisées pour classification d'entités de type *Person*

		RN 1	RN 2	RN 3	RN 4	RN 5	RN 6	RN 7	RN 8	RN 9
Carte typographique		Oui	Non	Oui	Non	Non	Non	Non	Oui	Oui
Carte des types des sous-séquences		Non	Oui	Oui	Non	Non	Non	Non	Oui	Oui
Carte relationnelle		Non	Non	Non	Oui	Oui	Oui	Oui	Oui	Oui
Nombre de relations					20	50	100	500	100	500
Nombre total de paramètres		4	21	25	31	86	169	867	194	892
Précision	Pire	0,626	0,589	0,798	0,7	0,720	0,726	0,718	0,838	0,638
	Meilleure	0,725	0,931	0,844	0,774	0,770	0,775	0,787	0,884	0,887
	Moyenne	0,703	0,799	0,821	0,740	0,745	0,748	0,767	**0,859**	0,799
Rappel	Pire	0,801	0,634	0,841	0,417	0,526	0,544	0,408	0,854	0,062
	Meilleur	0,963	0,970	0,892	0,484	0,573	0,617	0,701	0,887	0,902
	Moyen	0,839	0,783	0,870	0,461	0,552	0,578	0,651	**0,874**	0,609
F1	Pire	0,749	0,732	0,828	0,529	0,608	0,628	0,631	0,853	0,113
	Meilleur	0,777	0,787	0,867	0,592	0,657	0,681	0,737	0,886	0,879
	Moyen	0,763	0,763	0,845	0,568	0,634	0,652	0,701	**0,866**	0,631

utilisent l'ensemble des cartes avec 100 relations pour la carte relationnelle, ainsi que la fonction sigmoïde comme fonction d'activation. Nous pouvons constater qu'au-delà d'une couche cachée à 50 neurones, l'ajout de neurones ou de couches n'a plus d'impact sur les résultats. Pour la suite des travaux, nous avons donc fixé le modèle à une couche cachée de 50 neurones.

Le tableau Tableau 4.8 présente l'évaluation de l'impact du choix de la fonction d'activation utilisée dans le modèle sur le score F1. Dans cette analyse, les entrées utilisent l'ensemble des cartes avec 100 relations pour la carte relationnelle, ainsi qu'une unique couche cachée de 50 neurones. Nous pouvons constater que l'utilisation de *Rectified Linear Units* impacte très négativement les résultats. Ne pas utiliser de fonction d'activation ne permet pas, non plus, d'obtenir les meilleurs résultats. L'utilisation de la fonction sigmoïde ou de la fonction tangente hyperbolique comme fonction d'activation résulte sur des scores élevés très similaires. Pour la suite des travaux, nous avons donc fixé la fonction d'activation à la fonction sigmoïde.

Tableau 4.7 Évaluation de réseaux de neurones selon le nombre de couches cachées et de neurones pour classification d'entités de type *Person*

		RN 1	RN 2	RN 3	RN 4	RN 5	RN 6
Nombre de couches cachées		1	1	1	2	2	2
Nombre de neurones dans la couche cachée 1		1	50	500	10	50	500
Nombre de neurones dans la couche cachée 2					6	30	300
Précision	Pire	0,634	0,846	0,846	0,829	0,838	0,660
	Meilleure	0,868	0,875	0,873	0,859	0,884	0,869
	Moyenne	0,831	**0,862**	0,861	0,845	0,859	0,832
Rappel	Pire	0,827	0,859	0,750	0,869	0,854	0,851
	Meilleur	0,885	0,899	0,905	0,898	0,887	0,903
	Moyen	0,852	0,875	0,867	**0,880**	0,874	**0,880**
F1	Pire	0,717	0,858	0,798	0,850	0,853	0,743
	Meilleur	0,872	0,875	0,886	0,876	0,886	0,880
	Moyen	0,840	**0,868**	0,863	0,862	0,866	0,854

Le modèle final utilise donc les trois cartes en entrées avec 100 relations pour la carte relationnelle, une seule couche cachée de 50 neurones ainsi que la fonction sigmoïde comme fonction d'activation.

Évaluation de l'ensemble de tests

Une fois les paramètres et hyper-paramètres du modèle fixés, nous pouvons évaluer le modèle sur l'ensemble de tests L_{test}. Nous avons obtenu une précision $P = 0,866$, un rappel $R = 0,870$ et un score F1 $F1 = 0.868$. Le Tableau 4.9 présente 20 exemples de prédictions d'ensembles de L_{test}. Sur les 20 exemples, le modèle a effectué deux erreurs, il a en effet déterminé, à tort, que l'entité *Yabu* n'était pas de type *Person* et que l'entité *Langlois Siding* l'était.

Tableau 4.8 Évaluation de réseaux de neurones selon la fonction d'activation utilisée pour classification d'entités de type *Person*

		RN 1	RN 2	RN 3	RN 4
Fonction d'activation		Identité	Sigmoïde	Tanh	ReLU
Précision	Pire	0,722	0,846	0,839	0,628
	Meilleure	0,795	0,875	0,871	0,712
	Moyenne	0,755	**0,862**	0,855	0,672
Rappel	Pire	0,775	0,859	0,863	0,182
	Meilleur	0,883	0,899	0,897	0,0
	Moyen	0,837	0,875	**0,881**	0,208
F1	Pire	0,773	0,858	0,857	0,287
	Meilleur	0,812	0,875	0,882	0,345
	Moyen	0,793	**0,868**	**0,868**	0,317

Tableau 4.9 Exemples de 20 prédictions de l'ensemble de tests du modèle d'apprentissage pour les entités de type *Person*

Entité	Valeur attendue	Valeur obtenue
land surveyor	0	0
house Delormey-Edey	0	0
Narcisse Ménard	1	1
The next day	0	0
1930	0	0
Jean-Baptiste Langevin	1	1
Hemlock Street	0	0
Yabu	1	0
former Methodist Church of Aylmer	0	0
June 22	0	0
Marie-Madeleine-Claire Brassard Deschenaux	1	1
1906 and 1919	0	0
Charles Fortin	1	1
Red River	0	0
Joseph Dalbé Viau	1	1
Saint-Michel de Courville	1	1
Langlois Siding	0	1
1884 to 1896	0	0
early days	0	0
Maurice Martineau	1	1

4.5.3 Identification de nouvelles personnes

Une fois le modèle d'identification des entités de type *Person* entraîné, il nous est possible de l'utiliser sur de nouvelles entités dont le type est indéterminé (*Undef*). Nous avons analysé les résultats de l'identification de nouvelles entités de deux manières différentes. Dans un premier temps, nous analysons de manière ouverte sur l'ensemble des entités de type *Undef*. Dans un second temps, nous nous concentrons sur des relations spécifiques.

Recherche ouverte d'identification de nouvelles entités de type *Person*

La recherche ouverte de nouvelles entités de type *Person* consiste à générer une liste L_{undef} contenant l'ensemble des entités E_{undef}, sujets et objets des relations extraites dont le type est *Undef*. Pour chaque entité, on génère son ensemble de paramètres E^x_{undef}. On applique ensuite le modèle sur chaque élément de L_{undef} afin de déterminer s'il s'agit ou non d'une entité de type *Person*. Le Tableau 4.10 présente les résultats de 15 entités sélectionnées aléatoirement. Parmi les 15 exemples utilisés ici, nous avons pu déterminer automatiquement que les entités *architect Joseph-Pierre Ouellet* et *Saint Francis Xavier* étaient de type *Person*. Le modèle a également identifié, à tort, *several generations of Simard* comme étant une entité de type *Person*. Au final, nous avons pu identifier un total de 4 773 nouvelles entités comme étant de type *Person*, *StanfordNER* en ayant initialement identifié 10 547, soit presque 50% de plus. Une évaluation sur 200 entités de L_{undef} nous a permis d'évaluer la précision des identifications à 0,833, le rappel à 0,714 et le score F1 à 0,769.

Recherche d'identification de nouvelles entités de type *Person* en lien avec une relation

La recherche d'identification de nouvelles entités de type *Person* en lien avec une relation consiste à ne pas évaluer toutes les entités extraites tel que précédemment, mais plutôt se concentrer sur les entités directement en lien avec une relation afin de pouvoir mieux évaluer l'apport du nouveau modèle. Ainsi, pour des relations dont nous savons que le sujet ou l'objet devrait être de type *Person* (par exemple : pour la relation "was built by" dont l'objet est majoritairement de type *Person*) nous pouvons procéder à une identification du type des entités non définies, liées à cette même relation, avec notre modèle. Par exemple, nous pouvons appliquer le modèle sur toutes les entités objets des triplets typés ⟨*sujet (Patimmo), "was built by". objet (Undef)*⟩.

Tableau 4.10 Exemples de 15 nouvelles identifications d'entités de type *Person*

Entité de type indéterminé selon *Stanford-NER*	Identifié comme étant de type *Person* par notre modèle
born in 1830 in Saint-Mathias	0
expansion of Mont-Joli	0
hills	0
profitable	0
three-storey stone storehouse	0
Steel-to-wood bridge	0
architect Joseph-Pierre Ouellet	1
preservation	0
precolonial vegetation	0
Saint Francis Xavier	1
several generations of Simard	1
several bedrooms	0
accentuated	0
same vocation	0
mixed-use building	0

Afin d'évaluer l'impact de l'ajout du modèle, nous calculons la précision, le rappel et le score F1 des entités, en lien avec une relation, convenablement identifiées comme étant de type *Person* avant et après application du modèle. Pour cela, nous comptabilisons manuellement, parmi les entités E_{person} et E_{undef} connectées à une relation (par exemple, objets dans la relation "was built by" dont le sujet est de type *Patimmo*), le nombre de vrais positifs, faux positifs, vrai négatifs et faux négatifs. Le Tableau 4.11 présente les résultats de ces évaluations pour quatre relations différentes.

Nous pouvons constater que, dans chaque cas, l'application du modèle permet d'améliorer grandement le rappel malgré une diminution de la précision. Le score F1 est cependant, dans chaque cas, amélioré. Utiliser le modèle d'apprentissage pour extraire de nouvelles entités de type *Person* nous permet donc bien d'obtenir de meilleurs résultats et d'apporter une amélioration à l'utilisation seule des outils d'extraction d'entités nommées.

4.5.4 Identification de nouveaux patrimoines immobiliers

Nous avons pu démontrer, dans la section précédente, que l'utilisation d'un modèle d'apprentissage supervisé pour l'identification d'entités de type *Person* permettait d'obtenir de

Tableau 4.11 Évaluation des extractions de triplets selon la relation considérée avec optimisation des identifications des entités de type *Person*

Relation considérée	Scores avant application du modèle			Scores après application du modèle		
	P	R	F1	P	R	F1
Patimmo "was built by" **Person**	0,977	0,765	0,858	0,904	0,970	**0,936**
Person "built" Patimmo	0,916	0,738	0,817	0,855	0,971	**0,909**
Patimmo "was built for" **Person**	1	0,814	0,898	0,896	0,986	**0,939**
Patimmo "was sold to" **Person**	0,929	0,591	0,722	0,809	0,864	**0,835**

meilleurs résultats que l'utilisation seule d'outils déjà existants de NER. Nous avons posé l'hypothèse que ce modèle, avec seulement de légères adaptations des paramètres, pouvait également être appliqué à d'autres types d'entités. Pour confirmer cela, nous avons sélectionné le type *Patimmo* car celui-ci n'était initialement pas reconnu par les outils d'extraction utilisés. Son identification était faite uniquement à l'aide d'un faible nombre d'entités extraites de *schema.org*. Cela avait pour effet un faible taux de rappel sur les entités de ce type. Ce type *Patimmo* est un nouveau type lié au domaine patrimoniale qui s'ajoute à la liste des différents types obtenus par les outils d'extraction ouverte d'informations, enrichissant ainsi les annotations pouvant être obtenues.

Génération du modèle d'identification des entités de type *Patrimoine immobilier*

La génération de ce modèle est très similaire à celle du modèle d'identification des entités de type *Person*. En effet, nous conservons les mêmes cartes comme paramètres, cependant, la carte relationnelle est adaptée aux relations liées aux entités de type *Patimmo*. Pour la génération des entités négatives (dont le type ne correspond pas au type *Patimmo*) pour l'apprentissage, nous ne considérons pas les entités de type *Organization* et *Place* sachant qu'il peut y avoir des recoupements entre les entités de ces types et les entités du type *Patimmo* (par exemple : l'entité *Site patrimonial Philippe-Aubert-de-Gaspé* peut aussi bien être de type *Patimmo* que de type *Place*.)

Comme pour le modèle précédent, nous avons analysé par validation croisée l'impact sur la précision, le rappel et le score F1 de l'utilisation de chacune des combinaisons de cartes. Ces résultats sont présentés dans le Tableau 4.12.

Tableau 4.12 Évaluation de réseaux de neurones selon les cartes utilisées pour classification d'entités de type *Patimmo*

		RN 1	RN 2	RN 3	RN 4	RN 5	RN 6	RN 7
Carte typographique		Oui	Non	Oui	Non	Oui	Non	Oui
Carte des types des sous-séquences		Non	Oui	Oui	Non	Non	Oui	Oui
Carte relationnelle		Non	Non	Non	Oui	Oui	Oui	Oui
Nombre de relations					100	100	100	100
Nombre total de paramètres		4	21	25	178	182	199	203
Précision	Pire	0,743	0,780	0,838	0,808	0,853	0,818	0,886
	Meilleure	0,824	0,853	0,884	0,871	0,920	0,871	0,919
	Moyenne	0,785	0,810	0,860	0,842	0,885	0,849	**0,903**
Rappel	Pire	0,459	0,579	0,766	0,548	0,685	0,719	0,840
	Meilleur	0,546	0,624	0,835	0,647	0,764	0,798	0,886
	Moyen	0,506	0,606	0,808	0,603	0,722	0,767	**0,859**
F1	Pire	0,588	0,676	0,807	0,653	0,774	0,784	0,864
	Meilleur	0,644	0,714	0,850	0,730	0,811	0,820	0,898
	Moyen	0,615	0,682	0,833	0,702	0,795	0,805	**0,881**

Tel que nous avons pu le constater dans le modèle précédent, nous obtenons les meilleurs résultats par utilisation simultanée des trois cartes comme paramètres. Nous avons également pu constater que, tel que dans le cas de l'analyse des résultats pour la détection d'entités de type *Person*, le résultat de l'utilisation seule de la carte relationnelle offre un rappel faible. Nous pouvons cependant constater que, contrairement au cas de la détection d'entités de type *Person*, dans le cas présent, l'utilisation de chaque carte seule offre un rappel faible, quelle que soit la carte. Cela peut être expliqué par le fait que, pour la carte typographique, une entité de type *Patimmo* est moins soumise à des contraintes typographiques qu'une entité de type *Person* (qui elle est, généralement, composée de deux mots, avec deux majuscules). Dans le cas de la carte des types des sous-séquences, nous avons également pu remarquer que, dans 45% des entités $E_i \in L_{positifs}$, aucune séquence n'avait de type ou bien le ou les types présents étaient des types présents en fréquence équivalente pour les entités $E_i \in L_{négatifs}$. L'utilisation, seule, d'une des trois carte ne permet donc pas de déterminer si une entité est

de type *Patimmo* ou non. La combinaison des trois cartes permet, quant à elle, d'obtenir de bons résultats.

Évaluation de l'ensemble de tests

Une fois les paramètres et hyper-paramètres du modèle fixés, nous pouvons évaluer le modèle sur l'ensemble de tests L_{test}. Nous avons obtenu une précision $P = 0,904$, un rappel $R = 0,863$ et un score F1 $F1 = 0,883$. Le Tableau 4.13 présente 20 exemples de prédictions d'ensembles de L_{test}.

Tableau 4.13 Exemples de 20 prédictions de l'ensemble de tests du modèle d'apprentissage pour les entités de type *Patimmo*

Entité	Valeur attendue	Valeur obtenue
Simon Soupiran	0	0
Jean Giroux	0	0
Quebec merchant involved	0	0
1828, avenue de la Rivière-Jaune	1	1
Suzanne Joubert	0	1
mayor of Saint-Jean	0	0
Pinard	0	0
Browne	0	0
André Cournoyer	0	0
1873	0	0
Duffy	0	0
Alfred Lalime	0	0
Carey Canadian	0	0
Leonidas Langevin	0	0
Église de Notre-Dame-de-Guadalupe	1	1
Joseph-Pierre Ouellet	0	0
Amelia Torrance	0	0
editor	0	0
Maison Bédard-Parent	1	1
Fortunat Denis	0	0

Recherche ouverte d'identification de nouvelles entités de type *Patrimoine immobilier*

La recherche ouverte d'identification de nouvelles entités de type *Patrimoine immobilier*, tout comme celle pour les entités de type *Person*, consiste à parcourir l'ensemble des entités à type indeterminé *Undef* E_{undef} parmi les sujets et objets des relations extraites par l'extracteur de relations, puis, à appliquer le modèle après génération des cartes d'entrée E^{x}_{undef} pour chaque entité. Le Tableau 4.14 présente les résultats de 15 entités sélectionnées aléatoirement. Parmi les 15 exemples utilisés ici, nous avons pu déterminer, automatiquement, que les entités "Royal Exchange Building", "parish of Saints-Angels-de-la-Chine" et "larger church" étaient de type *Patimmo* selon notre modèle tandis que les autres ne l'étaient pas. Au final, nous avons pu identifier un total de 8 471 nouvelles entités comme étant de type *Patimmo*, 9 816 étant initialement identifiées comme tel.

Tableau 4.14 Exemples de 15 nouvelles identifications d'entités de type *Patimmo*

Entité de type indeterminé selon *Stanford-NER*	Identifié comme étant de type *Patimmo* par notre modèle
Royal Exchange Building	1
first Baptist community	0
lawyer William Duncan Herridge	0
set of four townhouses offered	0
parish of Saints-Angels-de-la-Chine	1
monumental columns	0
new subdivision	0
estate to Bernard Leonard	0
French inspiration	0
Odd Fellows	0
larger church	1
attached detached house built	0
Teachers	0
representatives	0
plan of 1704	0

Recherche d'identification nouvelles entités de type *Patrimoine immobilier* en lien avec une relation

Tel que dans le cas de l'identification de nouvelles entités de type *Person* en lien avec une relation, nous avons effectué l'identification de nouvelles entités de type *Patimmo* en lien avec

une relation. Nous avons également effectué, sur un total de quatre relations fréquentes dont le sujet ou l'objet est, dans la majeure partie des cas, de type *Patimmo*, une évaluation, sur la précision, le rappel et le score F1, de l'impact de l'utilisation de notre modèle d'identification de nouveaux patrimoines immobiliers. Cette évaluation est présentée dans le Tableau 4.15. Tel que nous pouvons le constater, dans chaque cas, nous avons, à nouveau, une nette augmentation du rappel contre une légère (sauf pour la relation "is in") diminution de la précision résultant sur une augmentation importante du score F1. L'application du modèle sur l'identification des entités de type *Patimmo* est plus important que celui pour les entités de type *Person* car, tel que le suggèrent les taux de rappel de chaque relation, une grande partie des patrimoines immobiliers, sans l'application du modèle, n'est pas identifiés.

Tableau 4.15 Évaluation des extractions de triplets selon la relation considérée avec optimisation des identifications des entités de type *Patimmo*

Relation considérée	Scores avant application du modèle			Scores après application du modèle		
	P	R	F1	P	R	F1
Patimmo "was built by" Person	1	0,635	0,777	0,941	0,945	**0,943**
Patimmo "was built in" Date	1	0,327	0,493	0,940	0,939	**0,940**
Patimmo "is in" State_or_province	1	0,076	0,141	0,676	0,316	**0,431**
Patimmo "was built around" Date	1	0,609	0,757	0,933	0,972	**0,953**

4.5.5 Amélioration du niveau de granularité de l'extraction

Dans la suite des travaux, nous avons tenté d'appliquer une approche similaire à celle décrite précédemment à un niveau de granularité plus précis. Nous avons voulu évaluer s'il était possible d'utiliser le modèle d'apprentissage, en adaptant les cartes d'entrée, aux entités de type *Patimmo* afin d'identifier leurs types précis (*House*, *Résidence*, *Church* etc.) Pour ce nouveau modèle, nous nous sommes concentrés sur l'identification, plus particulièrement, des entités de type *Church*, un des types les plus récurrents. Les ensembles d'apprentissages contiennent les patrimoines immobiliers que nous avons pu identifier comme étant de ce type (à l'aide du module d'identification des types de sites immobiliers) comme ensembles positifs pour $L_{positifs}$ et les patrimoines immobiliers dont nous avons pu identifier le type comme étant distincts de *Church*, comme ensembles négatifs pour $L_{négatifs}$. Le Tableau 4.16 présente les

résultats de l'évaluation croisée de ce modèle selon les cartes utilisées. Tel que nous pouvons le constater, les résultats, quelle que soit la combinaison de cartes utilisée, sont médiocres. Ces résultats peuvent être expliqués par la mauvaise performance des cartes sur la différentiation d'entités à ce niveau de granularité. En effet, les intuitions sur l'utilisation de ces cartes pour différencier les entités de différents types ne sont plus valides à ce niveau de granularité. On peut, par exemple, trouver de nombreux cas où les cartes typographiques et des types des sous-séquences seront identiques entre plusieurs entités dont le résultat devrait être différent. Une liste non exhaustive de ces cas est la suivante :

— "*Église Wesley United*" et "*Mine American Chrome*" ;
— "*Église Templeton United*" et "*Calvaire de Saint-Prime*" ;
— "*Église Southwest United*" et "*Gare de Vallée-Jonction*" ;
— "*Église St-Andrew*" et "*Ferme Joseph-Roy*" ;
— "*Église Sainte-Thérèse-d'Avila*" et "*Maison Joseph-Willie-Robidoux*".

Ces deux cartes ne permettent donc pas de distinguer les entités à ce niveau de granularité. Nous avons également effectué une analyse sur les couples ⟨*position - relation*⟩ les plus récurrents utilisés avec des patrimoines immobiliers de type *Church* qui nous a montré que la plus grande partie de ces couples n'étaient pas propres à ce type (contrairement à des couples tels que ⟨*Object - belongs to*⟩ pour identifier des entités de type *Person*). La liste des dix couples les plus récurrents pour les patrimoines immobiliers de type *Church* est la suivante :

— ⟨*Subject - is*⟩
— ⟨*Subject - was*⟩
— ⟨*Subject - was built in*⟩
— ⟨*Subject - has*⟩
— ⟨*Subject - is in*⟩
— ⟨*Subject - was built on*⟩
— ⟨*Subject - was named*⟩
— ⟨*Subject - replaces*⟩
— ⟨*Object - build*⟩
— ⟨*Subject - was built according to*⟩

La carte relationnelle ne nous permet donc pas de distinguer les églises des autres types de patrimoines immobiliers. Nous avons donc pu conclure que le modèle ne peut pas être

utilisé sans une adaptation majeure des paramètres d'entrées, aux entités de ce niveau de granularité. Nous avons également tenté une approche similaire pour distinguer les professions des personnes entre toutes les entités de type *Person* et avons obtenu des résultats similaires.

Tableau 4.16 Évaluation de réseaux de neurones selon les cartes utilisées pour classification d'entités de type *Church*

		RN 1	RN 2	RN 3	RN 4	RN 5	RN 6	RN 7
Carte typographique		Oui	Non	Oui	Non	Oui	Non	Oui
Carte des types des sous-séquences		Non	Oui	Oui	Non	Non	Oui	Oui
Carte relationnelle		Non	Non	Non	Oui	Oui	Oui	Oui
Nombre de relations					100	100	100	100
Nombre total de paramètres		4	21	25	163	167	184	188
Précision	Pire	0	0,467	0,375	0,308	0,333	0,267	0
	Meilleure	1	0,750	1	0,636	0,714	0,694	0,929
	Moyenne	0,3	0,631	0.680	0,502	0,513	0,493	0,551
Rappel	Pire	0	0,148	0,120	0,080	0,061	0,190	0
	Meilleur	0,029	0,4	0,241	0,163	0,240	0,490	0,318
	Moyen	0,007	0,179	0,171	0,136	0,146	0,293	0,4
F1	Pire	0	0,246	0,182	0,127	0,103	0,222	0
	Meilleur	0,057	0,333	0,375	0,250	0,329	0,575	0,456
	Moyen	0,015	0,275	0,271	0,212	0,2	0,366	0,326

4.6 Création manuelle, peuplement automatique et questionnement de l'ontologie et de la base de connaissances

Dans cette section des travaux, nous décrivons l'ontologie créée permettant d'ajouter une base de connaissances contenant les entités et relations que nous avons pu extraire des documents patrimoniaux. Cette ontologie a pour principal objectif de permettre la représentation des relations pouvant exister entre une personne et un patrimoine immobilier. On cherche, entre autres, à pouvoir identifier la personne ayant bâtis, acheté ou vendu un patrimoine immobilier. Dans cette ontologie, on permet également l'ajout d'informations supplémentaires portant sur une personne (son titre et/ou sa nationalité) ou sur un patrimoine immobilier (sa date de construction, approximative ou exacte). Cette ontologie nous permet donc de répondre à des

questions de compétences telles que *quel architecte a bâti le plus de patrimoines immobiliers ?*, ou encore *quels sont les dix églises les plus anciennes ?*.

4.6.1 Création des classes et relations de l'ontologie

Représentation des entités au sein de l'ontologie

Afin d'ajouter les entités extraites à notre base de connaissances, nous pouvons tirer profit de leurs types. En effet, nous pouvons séparer les entités dans des catégories *Person*, *Title*, *Patimmo* etc. Chacune de ces catégories est utilisée afin de créer une classe et nous ajoutons la classe *Entity* pour toutes les regrouper sous une même super-classe. Nous avons également créé la classe *Document* dont les instances sont les fiches patrimoniales issues du PIMIQ. Pour des questions de limite de temps disponible pour nos recherches, nous avons limité les types d'entités ajoutées à la base de connaissances aux types d'entités pouvant être sujet ou objet des relations que nous avons considérées, à savoir, les entités de type *Patimmo*, *Person*, *Date*, *Nationality*, *Title*. Il serait aisément possible d'ajouter les autres types à l'ontologie et permettre l'ajout de leurs instances dans la base de connaissances, cependant, avec les uniques relations considérées actuellement, ces instances ne seraient pas mises en relation.

Représentation des relations au sein de l'ontologie

Il est logique de penser que les prédicats contenus dans les relations extraites des textes devraient se traduire par des relations entre nos différentes classes dans l'ontologie. Cependant, en *RDF*, il n'est pas possible d'ajouter des informations sur une relation. Utiliser des relations pour traduire nos prédicats ne nous permettrait donc pas de préciser la phrase d'origine de la relation extraite ou encore la fiche patrimoniale d'où est extraite cette relation. Nous avons donc décidé, afin de pouvoir tenir compte de ces informations, de traduire les prédicats issus des relations extraites des textes par des classes. Ainsi, nous pouvons ajouter toute information désirée à chaque instance de ces classes. Afin de simplifier la hiérarchie des classes, nous avons créé une classe parente *Relation* dont plusieurs classes, tel que *BuiltRelation* ou *OccupationRelation*, héritent. Une relation extraite du texte se traduit donc, dans notre ontologie, comme une instance d'une classe fille de la classe *Relation*. Une propriété, *relationExtractedFrom*, permet de faire le lien entre l'instance de la relation et l'instance de la fiche dont est issue la relation. Une propriété, *relationExtractedFromSentence*, permet de préciser la phrase dont est extraite cette relation. Afin de lier les instances de relations aux instances d'entités représentant les sujets et objets de ces relations, nous utilisons, respectivement, les propriétés d'objets *relationSubject* et *relationObject*. Nous lions, également, les fiches patrimoniales, ins-

tances de la classe *Document*, aux patrimoines immobiliers auxquelles elles sont associées par l'utilisation de la propriété *documentIsAbout*. Tel que nous l'avons décrit plus haut dans ce mémoire, nous avons pu extraire les types de certains des patrimoines immobiliers à l'aide des classes de l'ontologie *schema.org*. Nous pouvons donc lier les instances de la classe *Patimmo*, dont nous avons pu identifier le type, à la classe de *schema.org* correspondante directement à l'aide de la relation *rdf:type*, ces instances devenant alors, à la fois, des instances de la classe *Patimmo* de notre ontologie locale et des instances de classes de l'ontologie de *schema.org*.

Schéma et description précise de l'ontologie finale

La Figure 4.7 présente le schéma global de l'ontologie du patrimoine immobilier. Dans les tableaux Tableau 4.17 à Tableau 4.20, nous décrivons les classes et propriétés présentes dans notre ontologie. Dans chacun des cas, nous intégrons également le nombre d'instances présentes dans la base de connaissances finale. Le Tableau 4.17 présente les classes de haut niveau de notre ontologie. Ces classes sont sous-classes directement de la classe *owl:Thing*. Les classes *Entity* et *Relation* étant utilisées uniquement afin de regrouper leurs classes filles sous une même classe, aucune instance n'est reliée directement à ces classes. Le nombre d'instances indiqué dans chaque cas représente donc la somme des instances de leurs classes filles. Le Tableau 4.18 présente les sous-classes de la classe *Entity* tandis que le Tableau 4.19 présente les sous-classes de la classe *Relation*. Finalement, le Tableau 4.20 présente les propriétés présentes dans notre ontologie patrimoniale.

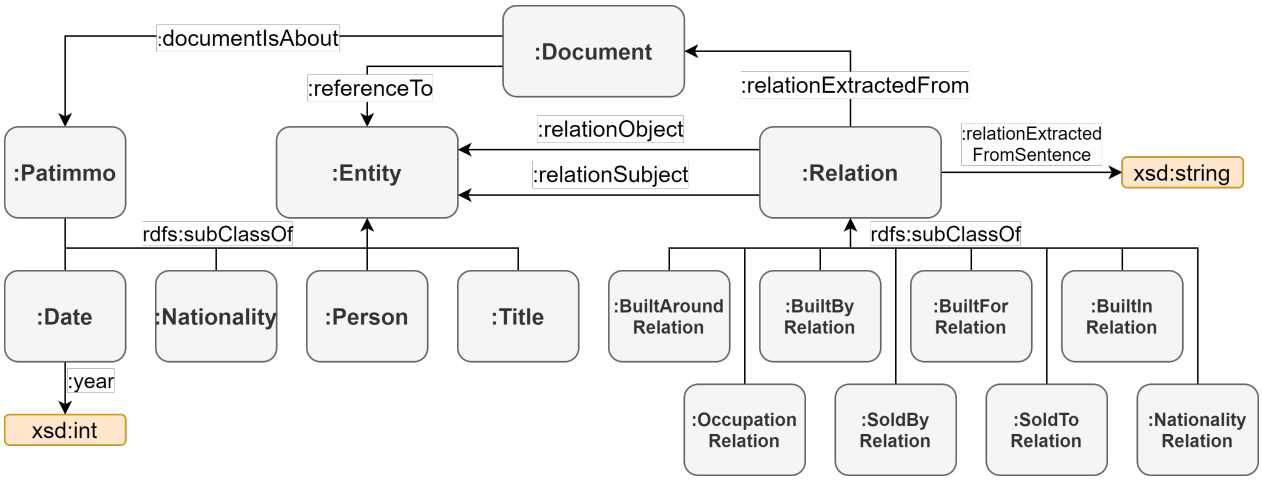

Figure 4.7 Schéma représentant les classes et relations de l'ontologie du patrimoine immobilier

Tableau 4.17 Liste des classes de haut niveau de l'ontologie patrimoniale

Classe	Description	Nombre d'instances	Exemple d'instance (label)
Document	Représente une fiche patrimoniale	7 699	Fiche: Maison Saint-Gabriel
Entity	Classe mère des classes représentant les différents types d'entités	89 213	Voir les instances des classes enfants
Relation	Classe mère des classes représentant les relations entre entités	8 733	Voir les instances des classes enfants

Tableau 4.18 Liste des sous-classes de la classe Entity de l'ontologie patrimoniale

Classe	Description	Nombre d'instances	Exemple d'instance (label)
Patimmo	Patrimoine immobilier	69 857	Maison Saint-Gabriel
Date	Date	2 170	April 1866
Nationality	Nationalité d'une personne	93	Irish
Person	Personne	15 218	Marcel Fradette
Title	Profession d'une personne	1 875	architect

Tableau 4.19 Liste des sous-classes de la classe Relation de l'ontologie patrimoniale

Classe	Description	Nombre d'instances	Exemple d'instance (label)
BuiltAround-Relation	Lien entre un Patimmo et une Date approximative de construction	342	The residence located at 6875, boulevard Saint-Jean was built around 1845.
BuiltBy-Relation	Lien entre un Patimmo et son constructeur	740	The current house was built in 1933 by Antoine Audet, blacksmith and sailor.
BuiltFor-Relation	Lien entre un Patimmo et la personne pour qui il a été construit	114	The Drainville Jewelery was built in 1924 for goldsmith Hervey Drainville (1889-1965).
BuiltIn-Relation	Lien entre un Patimmo et une Date exacte de construction	1 080	In 1727, a second church was built on a site known thereafter as the Berceau-de-Kamouraska.
Occupation-Relation	Lien entre une personne et son occupation/titre	5 744	The merchant JA Vézina then occupies the ground floor.
SoldBy-Relation	Lien entre un Patimmo et la personne qui l'a vendu	352	In 1926, Josephine Paradis sold the house to Armand Buteau.
SoldTo-Relation	Lien entre un Patimmo et la personne pour qui il a été vendu	157	The residence was sold in 1918 to Napoleon Lavoie

Tableau 4.20 Liste des propriétés de l'ontologie patrimoniale

Propriété	Domaine	Portée	Description	Nombre d'instances
documentIsAbout	Document	Patimmo	Lien entre la fiche patrimoniale et le site patrimonial décrit	7 699
referenceTo	Document	Entity	Lien entre la fiche patrimoniale et une entité présente dans sa description	118 093
relationExtracted-From	Relation	Document	Lien entre une relation et la fiche de laquelle celle-ci est issue	8 733
relationExtracted-FromSentence	Relation	xsd:String	Lien entre une relation et la phrase de laquelle elle est issue	8 733
relationSubject	Relation	Entity	Lien entre une relation et son entité sujet	8 733
relationObject	Relation	Entity	Lien entre une relation et son entité objet	8 733
year	Date	xsd:int	Lien entre une instance de Date et l'année en format entier, permet ainsi de faciliter les comparaisons entre dates	414

Exemple d'instances dans notre ontologie permettant de représenter une relation extraite de la fiche d'un patrimoine immobilier

La Figure 4.8 présente la représentation au sein de notre base de connaissances d'une relation extraite du texte d'une fiche patrimoniale. Les rectangles gris représentent nos classes, les verts nos instances, les oranges nos littéraux tandis que le rectangle bleu représente une classe issue d'une ontologie externe, à savoir, l'ontologie de *schema.org*. La fiche patrimoniale est celle de l'*Église West Brome United Chruch*. La relation que nous avons pu extraire est issue du triplet ⟨*church, was built by, Simon Shufelt*⟩ extrait automatiquement par l'outil *StanfordOIE* de la phrase "This Methodist church was built around 1857 by Simon Shufelt." Tel que nous pouvons le constater, l'entité *Simon Shufelt* a bien été identifiée comme étant de type *Person* et nous avons également pu lier l'entité *Church West Brome United Church* à la classe *Church* de l'ontologie schema.org. Nous avons pu également, automatiquement, faire le lien entre l'entité *Church* de la phrase et le patrimoine immobilier *Church West Brome United Church*.

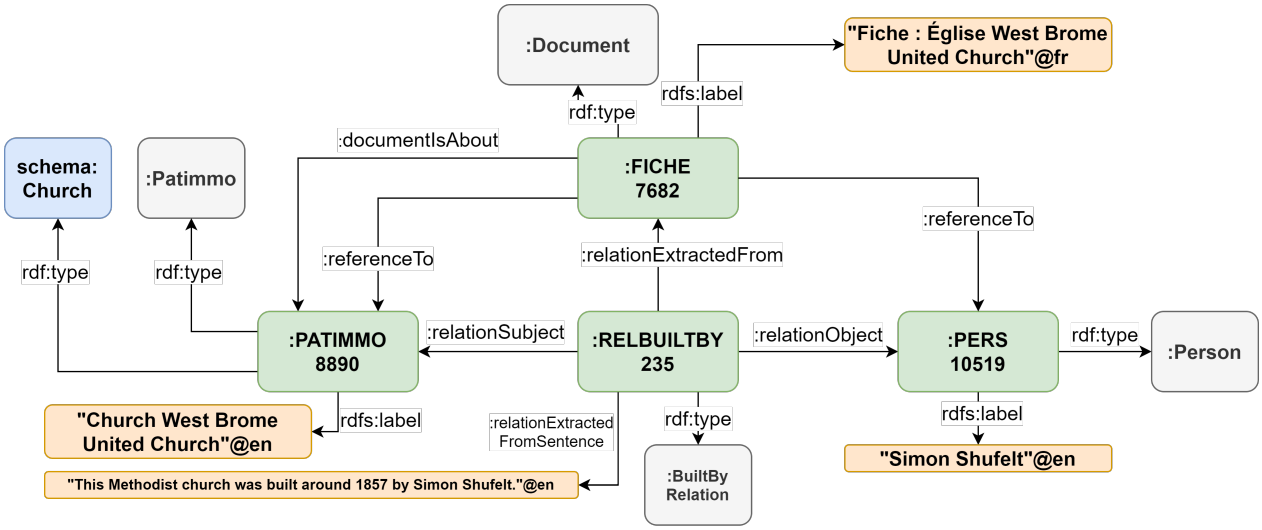

Figure 4.8 Exemple d'une relation extraite d'un texte de fiche patrimoniale telle qu'elle est représentée au sein de l'ontologie

4.6.2 Peuplement de la base de connaissances

Le peuplement de la base de connaissances est effectué en quatre étapes :

— Pour chaque fiche patrimoniale, nous générons une instance de la classe *Document* pour la représenter ainsi qu'une instance de la classe *Patimmo* afin de représenter le patrimoine immobilier considéré ;

— Pour l'ensemble des entités extraites par notre modèle et dont nous avons une classe associée au type dans notre ontologie (à savoir : *Patimmo*, *Date*, *Nationality*, *Person* et *Title*), nous générons une instance de ladite classe ;
— Pour chaque triplet ⟨*Sujet (Type du sujet), Prédicat, Objet (Type de l'objet)*⟩ issu de l'étape d'extraction des triplets d'intérêts ainsi que pour chaque nouveau triplet issu de notre modèle d'apprentissage supervisé et étant validé par les patrons de l'étape d'extraction des triplets d'intérêts, nous générons une instance d'une classe fille de la classe *Relation*. La classe dont l'instance doit être générée est identifiée par le prédicat du triplet ainsi que par le type du sujet et de l'objet. Les triplets réciproques tels que ⟨*Sujet (Person), Built, Objet (Patimmo)*⟩ et ⟨*Sujet (Patimmo), Built by, Objet (Person)*⟩ sont fusionnés sous la même relation, par exemple *BuiltBy* ;
— Nous connectons chacune des instances des classes filles de la classe *Relation*, précédemment créée, aux instances des entités des sujets et objets ainsi qu'aux l'instance du document dont sont issues les relations.

4.6.3 Requêtes à la base de connaissances

Un avantage majeur du Web sémantique et de la conception d'ontologies est de faciliter l'accès au contenu des bases de connaissances. Il est, en effet, possible, à l'aide de requêtes écrites grâce au langage *SPARQL*, d'obtenir des réponses rapidement à des questions qui prendraient beaucoup de temps à répondre manuellement (par recherche manuelle dans les textes de l'information désirée.) L'exemple *Quels patrimoines immobiliers, en lien avec quelle fiche patrimoniale, ont été construits avant 1680 ?* met bien en évidence cet avantage. La recherche manuelle de l'information au sein des textes est une tâche très fastidieuse. La requête *SPARQL* permettant de répondre à cette question, quant à elle, ne prend que dix lignes et quelques minutes à écrire. Dans la suite de ce mémoire, nous présentons dans les tableaux Tableau 4.21 à Tableau 4.28 sept exemples de questions de compétences auxquelles il est possible de répondre. Dans chaque cas, nous comparons l'impact de l'augmentation du modèle suite à l'utilisation du modèle d'apprentissage supervisé sur les résultats de la requête. Le PIMIQ possède également plusieurs fichiers csv contenant des données structurées contenant une partie de l'information présente sur les fiches patrimoniales. Dans ces fichiers, on peut trouver une liste des patrimoines immobiliers, une liste des personnes en lien avec ces patrimoines immobiliers, une liste de certaines relations entre les personnes et les patrimoines immobiliers ainsi qu'une liste des occupations des personnes. Pour chaque question de compétence, on ajoute également le résultat obtenu pour la requête appliquée à la base de données contenant uniquement les données issues de ces fichiers en lien avec les fiches patrimoniales que nous

avons utilisés. On peut ainsi comparer, dans un premier temps, l'impact de l'extraction automatique d'entités et de relations à partir des textes par rapport aux données contenues dans les fichiers. Dans un second temps, on peut comparer l'impact de l'utilisation de notre modèle d'apprentissage supervisé sur les résultats obtenus.

Afin de répondre à ces questions de compétences, les requêtes SPARQL nécessitent l'utilisation de différents opérateurs :

— *DISTINCT* : élimine les résultats dupliqués ;
— *GROUP BY* : regroupe les résultats partageant une même entité ;
— *COUNT* : comptabilise le nombre de résultats pour une entité ;
— *FILTER* : ajoute un opérateur de restriction sur les résultats de sortie ;
— *ORDER BY* : établit l'ordre dans lequel apparaissent les résultats de sortie ;
— *LIMIT* : limite le nombre maximum de résultats de sortie.

Ces questions nécessitent l'appel à une grande partie des classes et relations présentes dans l'ontologie, mais, il serait possible d'imaginer bien plus de questions de connaissance, aussi bien globale, telle que la question *Combien d'entités de chaque type y a-t-il dans la base de connaissances ?*, que précises, tel que la question *Quels patrimoines immobiliers, en lien avec quelle fiche patrimoniale, ont été construits avant 1680 ?*. Nous avons sélectionné ces questions afin de mettre en évidence le type de question de compétences auquel il est possible de répondre. On peut constater dans le Tableau 4.21 que les types *Nationality* et *Date* ne sont pas présents dans les données des fichiers structurés du PIMIQ. L'extraction automatique de ces types d'entités à partir de textes offre donc la possibilité d'utiliser celles-ci pour la base de connaissances, ce qui n'était pas possible par utilisation uniquement des fichiers structurés du PIMIQ. On note également que, pour les autres types, le nombre d'extractions effectuées est plus élevé que le nombre de données présentes dans les fichiers du PIMIQ. Le nombre de patrimoines immobiliers présents dans notre base de connaissances a également fortement augmenté après utilisation du modèle d'apprentissage supervisé. Dans le Tableau 4.22, on présente les vingt patrimoines immobiliers qui sont les plus fréquents dans notre base de connaissances après utilisation de notre modèle d'apprentissage supervisé. Les patrimoines immobiliers en gras représentent ceux dont le type a déjà pu être identifié à l'aide de la taxonomie de *schema.org* dans notre module d'identification des types de sites immobiliers. Pour la plupart d'entre aux, on peut constater qu'ils n'étaient pas présents avant utilisation de notre modèle. Celui-ci a donc permis de les identifier. L'utilisation de notre modèle permet donc d'extraire automatiquement des entités des textes qui ne sont pas prises en compte dans

les fichiers du PIMIQ. Une partie des entités identifiées comme étant de type *Patimmo* par notre modèle s'avèrent cependant ne pas être de ce type. Il s'agit majoritairement d'entités qui ne correspondent a aucun type de StanfordNER et donc sur lesquels notre modèle d'apprentissage n'a pas pu s'entraîner et effectuer de bonnes estimations.

Dans le Tableau 4.23, nous présentons la requête SPARQL permettant de répondre à la question de compétences *Combien d'instances distinctes de chaque relation y a-t-il dans la base de connaissances* ainsi que les résultats de cette requête. On peut y constater que plusieurs relations ne sont pas représentées dans les fichiers structurés du PIMIQ. Les relations *Occupation* et *Built By*, quant à elle, présentent plus d'instances dans ces fichiers que dans nos extractions. Cette différence entre le nombre de ces relations présentes au sein des fichiers de PIMIQ et de nos extractions provient de la grande variété de manières de présenter textuellement ces relations. Cette variété a pour conséquence que ces relations peuvent être difficilement identifiables automatiquement à partir des textes. Dans le cas de la relation *Occupation*, plusieurs instances trouvées dans les fichiers du PIMIQ sont absentes des textes ou conclues à l'aide d'éléments indirects (par exemple le triplet ⟨*Timothée Franchère, occupation, Milicien*⟩ se retrouve dans les fichiers du PIMIQ, la seule information pouvant indiquer cette relation dans les textes se trouvant dans la phrase *Joseph et Timothée Franchère ont tous deux été membres de la milice du Bas-Canada lors de la guerre qui a opposé les États-Unis à la Grande-Bretagne en 1812.*). Dans les tableaux Tableau 4.24 à Tableau 4.28, on peut voir que dans chaque cas, les instances issues des fichiers structurés du PIMIQ ne permettent pas d'obtenir de résultats. En effet, dans les fichiers du PIMIQ, le type des patrimoines immobiliers n'est pas spécifié, il n'est donc pas possible de distinguer, sans traitement supplémentaire, quels patrimoines immobiliers issus de ces fichiers est une église (par exemple). L'absence de date, dans ces fichiers, ne nous permet pas, non plus, de dater les patrimoines immobiliers, rendant impossible de répondre à des questions telles que *Quels patrimoines immobiliers ont été construits avant 1680 ?*. L'absence de nationalité dans les fichiers structurés du PIMIQ empêche également de répondre à la question *Quelles sont les cinq nationalités les plus courantes des architectes ?*. Notre approche par extraction des entités et relations au sein des textes à l'aide d'outils d'extraction ouverte d'informations permet donc d'augmenter notablement le nombre de relations distinctes extraites des documents du PIMIQ.

Tableau 4.21 Question de compétences : Combien d'entités de chaque type y a-t-il dans la base de connaissances ?

Question de compétences	Combien d'entités distinctes de chaque type y a-t-il dans la base de connaissances ?	

Requête SPARQL	SELECT ?type (COUNT(DISTINCT ?entity) as ?total)
	WHERE
	{
	?entity rdf :type ?type.
	?type rdfs :subClassOf :Entity.
	}
	GROUP BY ?type	

Données issues des fichiers structurés du PIMIQ	Person	4 794
	Patimmo	2 825
	Title	116
	Nationality	0
	Date	0
StanfordNER + StanfordOIE	Person	10 547
	Patimmo	9 816
	Title	1 901
	Nationality	93
	Date	2 186
StanfordNER + StanfordOIE + modèle d'apprentissage supervisé	Person	15 320
	Patimmo	69 857
	Title	1 901
	Nationality	93
	Date	2 186

Tableau 4.22 Liste des 20 patrimoines immobiliers les plus fréquents

Patrimoine immobilier	Fréquence
house	3 5
building	2 776
built	2 691
property	1 259
residence	1 074
part	998
construction	993
land	887
site	816
place	808
plans	763
parish	609
church	598
latter	570
work	483
buildings	463
roof	452
name	422
lot	399
chapel	397

En gras : patrimoine immobilier dont le type a déjà pu être identifié à l'aide de la taxonomie de *schema.org*.

Tableau 4.23 Question de compétences : Combien d'instances distinctes de chaque relation y a-t-il dans la base de connaissances ?

Question de compétences	Combien d'instances distinctes de chaque relation y a-t-il dans la base de connaissances ?	

Requête SPARQL	SELECT ?type (COUNT(DISTINCT ?relation) as ?total)
	WHERE
	{
	?relation rdf :type ?type.
	?type rdfs :subClassOf :Relation.
	}
	GROUP BY ?type	

Données issues des fichiers structurés du PIMIQ	OccupationRelation	6 486
	BuiltInRelation	0
	SoldToRelation	0
	BuiltByRelation	956
	BuiltForRelation	0
	BuiltAroundRelation	0
	SoldByRelation	0
	NationalityRelation	0
StanfordNER + StanfordOIE	OccupationRelation	5 444
	BuiltInRelation	336
	SoldToRelation	28
	BuiltByRelation	6
	BuiltForRelation	57
	BuiltAroundRelation	198
	SoldByRelation	53
	NationalityRelation	190
StanfordNER + StanfordOIE + modèle d'apprentissage supervisé	OccupationRelation	5 715
	BuiltInRelation	1 083
	SoldToRelation	191
	BuiltByRelation	703
	BuiltForRelation	116
	BuiltAroundRelation	341
	SoldByRelation	355
	NationalityRelation	205

Tableau 4.24 Question de compétences : Quelles personnes ont bâti des églises et quelle est leur profession ?

Question de compétences	Quelles personnes ont bâti des églises et quelle est leur profession ?	

Requête SPARQL	SELECT DISTINCT ?person_label ?occupation_label
	WHERE
	{
	?person rdf :type :Person ;
	rdfs :label ?person_label.
	?relBuilt rdf :type :BuiltByRelation ;
	:relationObject ?person ;
	:relationSubject/rdf :type schema :Church.)
	?relOccupation rdf :type :OccupationRelation ;
	:relationSubject ?person ;
	:relationObject ?occupation.
	?occupation rdfs :label ?occupation_label.
	}	

Données issues des fichiers structurés du PIMIQ		
StanfordNER + StanfordOIE	Elzéar Métivier	contractor
StanfordNER + StanfordOIE + modèle d'apprentissage supervisé	Elzéar Métivier	contractor
	Father Joseph-Onésime Brousseau	Pastor
	Thomas Pearson	carpenter

Tableau 4.25 Question de compétences : Quels patrimoines immobiliers, en lien avec quelle fiche patrimoniale, ont été construits avant 1680 ?

Question de compétences	Quels patrimoines immobiliers, en lien avec quelles fiches patrimoniales, ont été construits avant 1680 ?	

Requête SPARQL	SELECT DISTINCT ?patimmo_label ?fiche_label
	WHERE
	{
	?patimmo rdf :type :Patimmo ;
	rdfs :label ?patimmo_label.
	?builtinrelation rdf :type :BuiltInRelation ;
	:relationSubject ?patimmo ;
	:relationObject/ :year ?year.
	?fiche :referenceTo ?patimmo ;
	rdfs :label ?fiche_label.
	FILTER (?year < ”1680”8sd :int)
	}	

Données issues des fichiers structurés du PIMIQ		
StanfordNER + StanfordOIE	residence	Fiche : Maison Joseph-Canac-Dit-Marquis
	church	Fiche : Site patrimonial de la Place-de-l'Église
	church	Fiche : Basilique de Sainte-Anne-de-Beaupré
StanfordNER + StanfordOIE + modèle d'apprentissage supervisé	fort	Fiche : Bureau de poste de Trois-Rivières
	church	Fiche : Site patrimonial de la Place-de-l'Église
	first church	Fiche : Site patrimonial de la Place-de-l'Église
	first wooden chapel	Fiche : Église de Sainte-Marie-Madeleine
	chapel	Fiche : Église de Sainte-Marie-Madeleine
	wooden chapel	Fiche : Église de Sainte-Marie-Madeleine
	half-timbered church	Fiche : Basilique de Sainte-Anne-de-Beaupré
	church	Fiche : Basilique de Sainte-Anne-de-Beaupré
	first mill	Fiche : Maison Thibault-Soulard
	mill	Fiche : Maison Thibault-Soulard
	stone house	Fiche : Maison Laberge
	first stone house	Fiche : Maison Laberge
	Grondines windmill	Fiche : Moulin à vent de Grondines
	residence	Fiche : Maison Joseph-Canac-Dit-Marquis
	first residence	Fiche : Maison Joseph-Canac-Dit-Marquis

Tableau 4.26 Question de compétences : Quelles sont les cinq nationalités les plus courantes des architectes et combien y a-t-il d'architectes pour chacune ?

Question de compétences	Quelles sont les cinq nationalités les plus courantes des architectes et combien y a-t-il d'architectes pour chacune ?	

Requête SPARQL	SELECT DISTINCT ?nat_label (COUNT (DISTINCT ?person) as ?count)
	WHERE
	{
	?person rdf :type :Person.
	?relocc rdf :type :OccupationRelation ;
	:relationSubject ?person ;
	:relationObject/rdfs :label ?occ_label.
	FILTER regex(?occ_label, "architect").
	?relnat rdf :type :NationalityRelation ;
	:relationSubject ?person ;
	:relationObject/rdfs :label ?nat_label.
	}
	GROUP BY ?nat_label
	ORDER BY DESC(?count)
	LIMIT 5	

Données issues des fichiers structurés du PIMIQ		
StanfordNER + StanfordOIE	American	12
	French	4
	Belgian	2
	British	2
	Canadian	2
StanfordNER + StanfordOIE + modèle d'apprentissage supervisé	American	12
	French	4
	Belgian	2
	British	2
	Canadian	2

Tableau 4.27 Question de compétences : Quels sont les cinq types de patrimoines immobiliers les plus courants et combien y en a-t-il ?

Question de compétences	Quels sont les cinq types de patrimoines immobiliers les plus courants et combien y en a-t-il ?	

Requête SPARQL	SELECT DISTINCT ?type (COUNT(DISTINCT ?patimmo) as ?count)
	WHERE
	{
	?patimmo rdf :type :Patimmo, ?type.
	FILTER(?type != :Patimmo)
	FILTER(?type != :Entity)
	FILTER(?type != owl :Thing)
	}
	GROUP BY ?type
	ORDER BY DESC(?count)
	LIMIT 5	

Données issues des fichiers structurés du PIMIQ		
StanfordNER + StanfordOIE	http ://schema.org/House	3 434
	http ://schema.org/Residence	1 073
	http ://schema.org/Church	690
	http ://schema.org/City	592
	http ://schema.org/School	279
StanfordNER + StanfordOIE + modèle d'apprentissage supervisé	http ://schema.org/House	3 756
	http ://schema.org/Residence	1 116
	http ://schema.org/Church	744
	http ://schema.org/City	598
	http ://schema.org/School	310

Tableau 4.28 Question de compétences : Quels sont toutes les relations, avec sujets et objets que nous pouvons trouver dans la fiche : *Maison Richard-Cruice* ?

Question de compétences	Quels sont toutes les relations, avec sujets et objets que nous pouvons trouver dans la fiche : *Maison Richard-Cruice* ?		

Requête SPARQL	SELECT DISTINCT ?sub_lab ?rel_type ?obj_lab
	WHERE
	{
	?fiche rdfs :label "Fiche : Maison Richard-Cruice"@fr.
	?rel :relationExtractedFrom ?fiche ;
	:relationSubject/rdfs :label ?sub_lab ;
	:relationObject/rdfs :label ?obj_lab ;
	rdf :type ?rel_type.
	}		

Données issues des fichiers structurés du PIMIQ			
StanfordNER + StanfordOIE	house	:BuiltInRelation	1850s
	house	:BuiltInRelation	early 1850s
	doctor Peter Howard Church	:Occupation-Relation	owner
	Peter Howard Church	:Occupation-Relation	doctor
	house	:BuiltInRelation	1850s for Richard William Cruice
	doctor Peter Howard Church	:Occupation-Relation	Crown Attorney
	house	:BuiltInRelation	early 1850s for Richard William Cruice
	Charles Symmes	:Occupation-Relation	merchant
StanfordNER + StanfordOIE + modèle d'apprentissage supervisé	house	:BuiltInRelation	1850s
	house	:BuiltInRelation	early 1850s
	doctor Peter Howard Church	:Occupation-Relation	owner
	Peter Howard Church	:Occupation-Relation	doctor
	house	:BuiltInRelation	1850s for Richard William Cruice
	doctor Peter Howard Church	:Occupation-Relation	Crown Attorney
	house	:BuiltInRelation	early 1850s for Richard William Cruice
	Charles Symmes	:Occupation-Relation	merchant

4.7 Conclusion

Dans ce chapitre, nous avons présenté un pipeline permettant d'extraire, à partir de textes issus de documents patrimoniaux, une liste d'entités ainsi qu'une liste de relations entres ces entités. Par la suite, nous avons présenté une ontologie créée dans le but de structurer la base de connaissances dans laquelle sont ajoutées les entités et relations extraites. Nous avons présenté, également, l'impact de l'utilisation d'un modèle d'apprentissage supervisé sur le taux d'entités convenablement extraites. Les sorties suite à l'exécution des requêtes SPARQL issues des questions de connaissances ont permis de mettre en évidence l'impact positif de nos travaux et notamment de l'utilisation du modèle d'apprentissage supervisé. On a pu y constater que, dans la plupart des cas, la base de connaissances issue des extractions utilisant ce modèle offraient un plus grand nombre de résultats que l'utilisation des fichiers structurés déjà existants. On a en effet pu identifier, par rapport aux données déjà présentes dans les fichiers du PIMIQ, trois fois plus d'entités de type *Person*, 25 fois plus d'entités de type *Patimmo* (cependant, comme discuté dans la section 5.3, une grande partie de ces entités ne sont pas désambiguïsées) et 16 fois plus d'entités de type *Title*. On a également pu identifier des entités dont le type est inexistant dans les fichiers du PIMIQ, à savoir, 93 entités de type *Title* et 2 186 entités de type *Date*. Par rapport aux données déjà présentes dans les fichiers du PIMIQ, on a retrouvé 11% moins de relations d'occupation de poste (*OccupationRelation*) et 26% moins de relations de construction (*BuiltByRelation*). On a cependant pu identifier des relations inexistantes dans les fichiers du PIMIQ, à savoir, 1 083 relations de date de construction (*BuiltInRelation*), 191 relations de vente de patrimoine immobilier à une personne (*SoldToRelation*), 116 relations de construction de patrimoine immobilier pour une personne (*BuiltForRelation*), 341 relations de date approximative de construction de patrimoine immobilier (*BuiltAroundRelation*), 355 relations de vente par une personne de patrimoine immobilier (*SoldByRelation*) et 205 relations de nationalité d'une personne (*NationalityRelation*). Notre approche offre donc un avantage majeur dans le domaine patrimonial car elle permet d'extraire des relations de texte jusqu'alors ignorées. Notre approche permet également d'extraire automatiquement les entités et relations des documents patrimoniaux, évitant ainsi la tâche fastidieuse de devoir effectuer cette extraction manuellement afin de peupler la base de connaissances.

CHAPITRE 5 CONCLUSION

Dans ce chapitre, nous effectuons un retour sur les travaux qui ont été effectués dans le cadre de ce mémoire. Nous revenons brièvement sur les résultats obtenus puis discutons des limitations actuelles des approches proposées. Finalement, nous présentons les améliorations futures qui pourront être apportées à ce projet.

5.1 Synthèse des travaux

Dans le cadre de nos recherches, l'objectif principal de nos travaux était de cibler et appliquer les méthodes permettant l'extraction d'entités et de relations au sein de textes francophones dans le domaine patrimonial en utilisant, d'une part, les arbres syntaxiques des phrases, et d'autre part, les outils existants d'extraction ouverte d'informations avec un modèle d'apprentissage supervisé. L'intérêt de ces extractions étant de les ajouter à une base de connaissances. Nous avons présenté deux approches différentes d'extraction d'entités et relations au sein de contenu textuel. Dans les deux cas, nous avons réussi à extraire des relations (et entités dans le second cas) à partir des textes.

Dans un premier temps, nous avons discuté d'une méthode d'extraction par patrons syntaxiques qui nous permettait d'identifier puis d'extraire des relations de composition entre artefacts et matériaux au sein de rapports archéologiques. Cette méthode s'appliquait directement sur le texte français sans nécessiter une phase de traduction pouvant ajouter une forme d'erreur. Nous avons pu déceler que, dans ces rapports archéologiques, la relation de composition s'exprimait très majoritairement par deux patrons qui nous ont permis, par la suite, d'extraire un grand nombre d'instances de ces relations. Ces instances ont, ensuite, pu être ajoutées à une base de connaissances afin de permettre l'exécution aisée de requêtes pour répondre à des questions de connaissances. Cette méthode nécessitait cependant de cibler d'avance les relations à extraire et ne permettait pas une extraction ouverte de relations.

Dans un second temps, nous avons présenté une méthode d'extraction ouverte d'entités et de relations au sein de documents patrimoniaux par l'utilisation d'outils déjà existants. Nous avons également discuté d'une méthode permettant l'augmentation du nombre d'entités détectées, mais également d'une méthode permettant la détection de nouveaux types d'entités

non détectées par les outils déjà existants. Cette méthode nous a, principalement, aidés à trouver des entités pour notre nouveau type *Patimmo*. Par l'étude des résultats obtenus par l'application de notre modèle de détection de nouvelles entités, nous avons pu démontrer que celui-ci proposait un avantage réel sur l'augmentation du nombre d'entités détectées par les outils déjà existants au sein d'un corpus de textes. À l'aide des entités et relations extraites, nous avons pu créer une seconde base de connaissances, distincte de la première, qui nous a permis d'exécuter différentes requêtes afin de répondre à diverses questions sur les patrimoines immobiliers du domaine patrimonial.

Nos travaux apportent une contribution dans le domaine de l'utilisation d'outils de traitement automatique de langue naturelle pour permettre le peuplement de bases de connaissances à partir de textes français. Nous avons, notamment, pu présenter une méthode permettant d'améliorer les extractions d'entités au sein de textes à l'aide de modèles d'apprentissage automatique. Nous avons pu montrer que les résultats de l'application de notre méthode étaient positifs sur deux types d'entités du domaine patrimonial. Cette méthode peut être aisément appliquée à d'autres corpus de textes et sur d'autres types d'entités. Nous avons également montré une méthode permettant (après utilisation d'outils d'extraction ouverte d'informations et de notre modèle d'apprentissage) de créer une ontologie représentant les types d'entités et les relations présentes au sein des textes. Finalement, nous avons pu peupler la base de connaissances associée à cette ontologie pour y effectuer des requêtes qui ont démontré l'intérêt de l'utilisation d'une telle base de connaissances. Nos travaux offrent donc un intérêt majeur pour l'utilisation des outils du web sémantique à partir de données textuels. En effet, nos travaux permettent de transformer des données textuelles en français de rang une étoile selon le modèle de Michael Hausenblas [30] (présenté dans la revue de littérature de ce mémoire) en données liées et structurées de rang cinq étoiles.

Nos travaux apportent également une contribution importante dans le domaine patrimonial et notamment en français. Dans ce domaine et particulièrement sur la culture québécoise, il n'existe que peu d'initiatives dans ce sens sur des textes français. Nos travaux mettent en lumière les possibilités d'extractions de textes culturels permettant une réutilisation aisée des données extraites.

5.2 Retour sur les objectifs initiaux et questions de recherche

Notre premier objectif, dans le cadre de ces travaux, était de permettre l'extraction automatique de relations de composition entre artefacts et matériaux au sein de rapports archéo-

logiques en français, notre question de recherche étant de savoir dans quelle mesure il était possible d'exploiter l'analyse syntaxique d'un texte à cette fin. Tel que nous l'avons décrit, nous avons pu utiliser l'analyse syntaxique des textes afin de permettre la génération de patrons syntaxiques permettant d'identifier une relation donnée. Nous avons donc pu compléter cet objectif avec cependant certaines limitations qui sont présentées dans la section suivante.

Notre deuxième objectif était de permettre une extraction automatique d'entités et de relations présentes dans des documents patrimoniaux du PIMIQ, notre question de recherche étant de savoir si la traduction des textes en anglais suivi de l'utilisation d'outils d'extraction ouverte d'informations nous permettait d'extraire des relations présentes dans des textes et de produire une base de connaissances permettant de répondre à des questions sur ces mêmes textes. Bien que nous n'ayons pas pu évaluer le biais apporté par la traduction des textes ainsi que le rappel global sur le nombre d'entités et relations extraites, nous avons pu compléter l'objectif en permettant l'extraction ouverte d'entités et de relations de ces textes.

Notre troisième objectif et contribution la plus importante de nos travaux était de savoir s'il était possible d'améliorer les extractions effectuées par les outils matures d'extraction ouverte d'informations, notre question de recherche étant de savoir si l'utilisation d'un modèle d'apprentissage supervisé pouvait permettre cela. Tel que nous avons pu le montrer, l'utilisation de notre modèle d'apprentissage nous a permis d'identifier de nombreuses nouvelles entités, améliorant ainsi le rappel global sur le nombre d'entités extraites. Nous avons donc répondu à cet objectif.

5.3 Limitations de la solution proposée

Lors de nos travaux, nous avons pu constater que la première méthode adoptée nous limitait grandement sur le nombre de relations extraites, la relation de composition entre artefacts et matériaux étant la seule considérée. Cette méthode ne nous permettait pas, non plus, d'identifier de nouvelles entités, une liste d'entités possibles étant fournie à l'avance. La seconde méthode adoptée avait pour objectif de palier à ces limitations.

Nous avons pu constater, par l'application de notre seconde méthode d'extraction d'entités et relations, que le modèle d'augmentation du nombre d'entités détectées ne pouvait pas être utilisé pour n'importe quel type d'entités. En effet, lorsque nous avons tenté d'augmenter la granularité de la détection afin de préciser le type des patrimoines immobiliers extraits,

nous avons pu constater que les résultats obtenus étaient médiocres, les paramètres d'entrée, comme discutée, ne permettant plus de distinguer les différents types à ce niveau de granularité. Dans l'application de cette seconde méthode, nous n'avons, également, pas tenu compte du biais probable ajouté par la traduction du français vers l'anglais des textes.

Une limitation actuelle importante de notre approche est le manque de désambiguïsation des entités de type *Patimmo*. En effet, on peut constater dans le Tableau 4.22 qu'une grande partie des entités identifiées comme patrimoines immobiliers par notre système ne sont pas suffisamment bien identifiées. On peut voir, par exemple, que notre modèle a permis d'identifier *chapel* comme étant une entité de type *Patimmo* et que nous avons retrouvé 397 fois dans nos extractions. Ces chapelles ne sont, cependant, pas distinguables entre elles. L'instance *chapel* extraite de la phrase *Une première chapelle est érigée en 1863* issue de la fiche patrimoniale *Site patrimonial de la Paroisse-de-Saint-Edmond* n'est, par exemple, pas désambiguïsée comme étant la chapelle propre à ce site. Parmi les 69 857 instances de la classe *Patimmo*, on a pu en identifier uniquement 5 833 possédant au moins un nom propre.

Dans la seconde partie des travaux effectués, nous avons présenté une méthode d'extraction d'entités et de relations utilisant un module de traduction des textes. En sortie, nous avons donc des entités et relations en anglais alors que le texte initial se trouvait être en français. À l'heure actuelle, aucune traduction des éléments de sortie n'est effectuée pour repasser au français. Il est donc nécessaire, pour le moment, pour l'utilisateur de la base de connaissances, de lui-même faire le lien entre les entités et relations en anglais dans la base de connaissances avec le texte en français des fiches patrimoniales.

5.4 Améliorations futures

Lors de nos travaux, nous avons identifié un type spécifique au domaine patrimonial, le type *patrimoine immobilier*. Lors de travaux futurs, il serait intéressant de discerner d'autres types spécifiques et de permettre leurs extractions. Par la suite, il serait également possible de discerner d'autres relations à extraire, puis, à ajouter dans la base de connaissances.

Une direction future du projet pourrait porter également sur l'utilisation d'un système de retour d'informations (ou feedback) afin d'évaluer la qualité des extractions effectuées. Une entité qui serait reportée comme inexacte pourrait être éliminée de la base de connaissances et les modèles d'apprentissage supervisé pourraient être réajustés en conséquence.

Il serait également intéressant d'analyser l'impact, dans chaque modèle d'apprentissage supervisé, de chaque entrée à l'aide de méthodes d'analyse de paramètres afin d'identifier, parmi la liste des entrées des modèles, celles ayant le moins d'impact et évaluer si leurs suppressions du modèle pourraient en améliorer les résultats.

Une étape de traduction des sorties de l'anglais au français serait donc intéressante à ajouter. Comme discuté, le modèle ne permet pas, à l'heure actuelle, de différencier certains types d'entités selon le niveau de granularité. Il serait intéressant d'adapter ce modèle d'apprentissage et notamment les entrées utilisées par celui-ci afin de permettre l'identification d'un spectre plus large d'entités.

RÉFÉRENCES

[1] "Frmg wiki." [En ligne]. Disponible : http://alpage.inria.fr/frmgwiki/

[2] F. Bauer et M. Kaltenböck, "Linked open data : The essentials," *Edition mono/monochrom, Vienna*, vol. 710, 2011.

[3] M. Song *et al.*, "Pkde4j : Entity and relation extraction for public knowledge discovery," *Journal of Biomedical Informatics*, vol. 57, p. 320 – 332, 2015. [En ligne]. Disponible : http://www.sciencedirect.com/science/article/pii/S1532046415001756

[4] C. D. Manning *et al.*, "The Stanford CoreNLP natural language processing toolkit," dans *Association for Computational Linguistics (ACL) System Demonstrations*, 2014, p. 55–60. [En ligne]. Disponible : http://www.aclweb.org/anthology/P/P14/P14-5010

[5] V. Kashyap, C. Bussler et M. Moran, *The Semantic Web : Semantics for Data and Services on the Web*, 1er éd. Springer Publishing Company, Incorporated, 2008.

[6] E. Marchand, M. Gagnon et A. Zouaq, "Extraction of a knowledge graph from french cultural heritage documents," dans *DOING : Intelligent Data – From Data to Knowledge*, 2020.

[7] A. M. Turing, *Computing Machinery and Intelligence*. Dordrecht : Springer Netherlands, 2009, p. 23–65. [En ligne]. Disponible : https://doi.org/10.1007/978-1-4020-6710-5_3

[8] A. Pinar Saygin, I. Cicekli et V. Akman, "Turing test : 50 years later," *Minds and Machines*, vol. 10, no. 4, p. 463–518, 2000. [En ligne]. Disponible : https://doi.org/10.1023/A:1011288000451

[9] M. J. Nye, "Speaking in tongues," Jun 2016. [En ligne]. Disponible : https://www.sciencehistory.org/distillations/magazine/speaking-in-tongues

[10] M. Jafari *et al.*, "Automatic text summarization using fuzzy inference," dans *2016 22nd International Conference on Automation and Computing (ICAC)*, Sep. 2016, p. 256–260.

[11] Mausam *et al.*, "Open language learning for information extraction," dans *Proceedings of Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CONLL)*, 2012. [En ligne]. Disponible : https://dl.acm.org/citation.cfm?id=2391009

[12] L. D. Corro et R. Gemulla, "Clausie : Clause-based open information extraction." [En ligne]. Disponible : http://resources.mpi-inf.mpg.de/d5/clausie/clausie-www13.pdf

[13] N. Bhutani, H. V. Jagadish et D. Radev, "Nested propositions in open information extraction," communication présentée à 2016 Conference on Empirical Methods in

Natural Language Processing, Austin, Texas, nov 2016, p. 55–64. [En ligne]. Disponible : https://www.aclweb.org/anthology/D16-1006

[14] K. Gashteovski, R. Gemulla et L. Del Corro, "Minie : Minimizing facts in open information extraction," dans *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, p. 2630–2640. [En ligne]. Disponible : https://www.aclweb.org/anthology/D17-1278

[15] A. Lauscher, Y. Song et K. Gashteovski, "Minscie : Citation-centered open information extraction," dans *Proceedings of ACM/IEEE Joint Conference on Digital Libraries*, 2019. [En ligne]. Disponible : https://madoc.bib.uni-mannheim.de/49216/

[16] W. Léchelle, F. Gotti et P. Langlais, "Wire57 : A fine-grained benchmark for open information extraction," 09 2018. [En ligne]. Disponible : https://arxiv.org/abs/1809.08962

[17] A. Zouaq, M. Gagnon et L. Jean-Louis, "An assessment of open relation extraction systems for the semantic web," *Information Systems*, vol. 71, p. 228 – 239, 2017. [En ligne]. Disponible : http://www.sciencedirect.com/science/article/pii/S0306437916304999

[18] M. Honnibal et I. Montani, "spaCy 2 : Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing," 2017. [En ligne]. Disponible : https://spacy.io/

[19] R. Jiang, R. E. Banchs et H. Li, "Evaluating and combining name entity recognition systems," dans *Proceedings of the Sixth Named Entity Workshop*. Berlin, Germany : Association for Computational Linguistics, août 2016, p. 21–27. [En ligne]. Disponible : https://www.aclweb.org/anthology/W16-2703

[20] A. Fader, S. Soderland et O. Etzioni, "Identifying relations for open information extraction," dans *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*. Edinburgh, Scotland, UK. : Association for Computational Linguistics, juill. 2011, p. 1535–1545. [En ligne]. Disponible : https://www.aclweb.org/anthology/D11-1142

[21] G. Angeli, M. J. Johnson Premkumar et C. D. Manning, "Leveraging linguistic structure for open domain information extraction," dans *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1 : Long Papers)*. Beijing, China : Association for Computational Linguistics, juill. 2015, p. 344–354. [En ligne]. Disponible : https://www.aclweb.org/anthology/P15-1034

[22] "Syntaxnet." [En ligne]. Disponible : https://opensource.google/projects/syntaxnet

[23] S. Auer *et al.*, “Dbpedia : A nucleus for a web of open data,” dans *The Semantic Web*, K. Aberer *et al.*, édit. Berlin, Heidelberg : Springer Berlin Heidelberg, 2007, p. 722–735.

[24] D. Vrandečiundefined, “Wikidata : A new platform for collaborative data collection,” dans *Proceedings of the 21st International Conference on World Wide Web*, ser. WWW ’12 Companion. New York, NY, USA : Association for Computing Machinery, 2012, p. 1063–1064. [En ligne]. Disponible : https://doi.org/10.1145/2187980.2188242

[25] S. Weibel, “The dublin core : A simple content description model for electronic resources,” *Bulletin of the American Society for Information Science and Technology*, vol. 24, n°. 1, p. 9–11, 2005. [En ligne]. Disponible : https://asistdl.onlinelibrary.wiley.com/doi/abs/10.1002/bult.70

[26] P. F. Patel-Schneider, “Analyzing schema.org,” dans *The Semantic Web – ISWC 2014*, P. Mika *et al.*, édit. Cham : Springer International Publishing, 2014, p. 261–276.

[27] World Wide Web Consortium’s RDF Data Access Working Group. (2008) SPARQL Query Language for RDF. [En ligne]. Disponible : https://www.w3.org/TR/rdf-sparql-query/

[28] “Propertyreificationvocabulary.” [En ligne]. Disponible : https://www.w3.org/wiki/PropertyReificationVocabulary

[29] F. Pedregosa *et al.*, “Scikit-learn : Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, p. 2825–2830, 2011.

[30] D.-L. Magazine, “The five stars of online journal articles—a framework for article evaluation,” *D-Lib Magazine*, vol. 18, n°. 1/2, 2012.

ANNEXE A Tableau des triplets ⟨phrase,artefact,matériau⟩ sans chemin syntaxique direct entre l'artefact et le matériau

Tableau A.1 Triplets ⟨phrase,artefact,matériau⟩ sans chemin syntaxique direct entre l'artefact et le matériau

Phrase	Artefact	Matériau
Afin d'illustrer les résultats de l'expérience, le juge présenta un modèle du petit poêle portatif communément employé par les paysans russes il consistait en un four en fonte de petites dimensions, entouré sur trois côtés d'un écran de briques ou de faïence.	poele	fonte
1 2 verre incolore boisson contenant bouteille de lait fragments de corps se 20ième.	bouteille	verre
La fonction entreposage est représentée par des fragments de bouteille de boisson gazeuse en verre américain, un fragment de couvercle en métal ainsi qu'un fragment de pot en verre teinté régulier vert.	pot	metal
céramique de table assiette.	table	ceramique
* Différence entre le total lithique et le total du tableau s'explique du fait que quelques.	tableau	lithique
Aussi, des traces de rubéfaction et et du bois carbonisé se trouvent dans le premier niveau dans le champ est, sur toute la superficie inventoriée.	niveau	bois
Forme et composition du foyer : piquets de bois pour poêle de métal.	poele	bois
1 4 terre cuite fine alimentation céramique de table petit pot.	table	ceramique
7 morceaux de cuir I croix de chapelet I bouton.	chapelet	cuir
l'assemblage lié à l'alimentation et à la consommation de boissons, la présence d'objets rarement retrouvés sur les sites archéologiques, tels les restes d'un pichet en faïence, d'une marmite en terre cuite grossière et d'un égouttoir (photos 21 à 23).	pichet	terre cuite

41 TCF argileuse blanche Pipe F 1 4.	pipe	argileuse blanche
assises (7 visibles) régulières de moellons de pierre calcaire ébauchés et jointoyés avec du mortier sur ses trois.	mortier	pierre
Un seul tesson de terre cuite fine blanche provenant d'une assiette creuse ou d'un bol a été recueilli de l'excavation.	assiette creuse	blanche
verre de couleur, bleu Perle à broderie.	perle	verre
La sous-opération 1F correspond à un sondage de 0,50 m de côté qui a été creusé à la pelle carrée à l'est d'une structure en bois (1G1-st).	pelle	bois
fragments dont 2 de verre plat et deux très minces avec de légères courbes (globe de lampe à l'huile ?).	plat	verre
plomb et un tesson de verre à bouteille de couleur verte.	bouteille	verre
En considérant les os de Poulet et de Phasianinés ensemble, tous les éléments du squelette sont représentés incluant deux crânes et des os du bas des pattes sous la cuisse.	poulet	os
remblai des fosses recelait : de la terre cuite grossière locale, du creamware, du pearlware, un fragment de pipe, du verre à vitre teinté vert, du verre incolore au plomb, des clous forgés, une bande en fer forgé et un os des artéfacts datés du début du XIXe siècle.	vitre	verre
Cerceau de bois à l'intérieur à la base formé d'une pièce plus ou moins ovale de 3 cm max de largeur.	forme	bois
La perle de verre provient quant à elle du sondage n 31, à l'extrémité sud de la concentration B.	perle	verre
Présence de coquilles d'oeufs et de verre à vitre à la base de cette couche mais sur les planchettes de bois visible dan la paroi nord de la tranchée 15.	verre	bois
Verre plat très épais et ciselé (gauffré).	plat	verre

-pete beige à noyau gris avec glaçure vert olive : un fragment de le hase d'une terrine et un tesson d'une pl!'re n.	terrine	glacure
3 TCG sans glaçure Brique F 1 1 4.	brique	sans glacure
Ailleurs sur la station on trouve de façon éparse des os écrus ou calcinés (tableau 38), 4 perles de verre, 1 couteau de métal, 4 plombs de fusil, des morceaux de métal et 2 mires (planche 1 d) (fig.	couteau	os
d'épaisseur, a laissé place à un sable brun orangé (4E4), friable et légèrement graveleux, affichant quelques petits fragments de bois brûlé, sept clous, cinq tessons de verre à vitre et une boucle de bretelle.	vitre	verre
Des tessons de grès grossier semblent se rapporter à un nombre restreint de contenants (2 ou 3), dont une cruche avec un emblème en relief sur l'épaule.	cruche	gres
Tableau 4 : Résultats de l'analyse des charbons de bois trouvés dans le foyer amérindien.	tableau	bois
Photo 14 - Bol domestique du troisième quart du XIXe siècle, reconstitué à partir de fragments de céramique en terre cuite fine blanche échantillonnés à l'intérieur de la latrine ST-9 (lot 1C3).	bol	blanche
Dans la portion est du sommet du vestige, des résidus de métal ferreux de forme circulaire mesurant 0.	forme	metal
Le verre vert foncé n'est représenté que par deux formes, la bouteille et.	bouteille	verre
Parmi les objets de verre reliés à l'alimentation, on remarque des fragments de gobelet, des bouteilles à condiment et une bouteille à boisson gazeuse portant la marque CHARLES WILSON/-MONTREA12.	bouteille	verre

tessons de verre à vitre, teintés bleu, un fragment de contenant en grès rhénan décoré au bleu de cobalt, un fond de sol muni de trois pattes, en faïence à cul noir, un tuyau de pipe en terre cuite blanche portant un décor fait à la molette.	pipe	terre
Photo 59 - Site MTL02-23-1 — Le cliché montre bien l'infrastructure de support constituée de deux fortes poutres de bois passant sous le chemin de fer dans l'axe de.	cliche	fer
Puisque le bâtiment est en bois et fait en pièces sur pièces, il est vraisemblable que ce vestige maçonné ait pris fin au niveau du premier plancher des parties ouest et centrale.	niveau	bois
Photo 45 Divers artefacts provenant du site BkFi-37 : a) fragments de tuyau et de fourneau en terre fine argileuse blanche b) tige de verre à vin en verre incolore c) fragment de pierre à fusil en silex d) terre cuite commune Stafforshire, e) métal cuivreux f) verre à bouteille de couleur verte g) sceau en plomb.	tuyau	metal
Verre de bouteille, briques, TCF, clous, os frais de boucherie, pot de fleur, plastique, clous découpés et tréfilés, penture de porte , bille de verre transparente, grès grossier à glaçure brune, pipe de plâtre.	pot	verre
Autre modèle de tente de toile avec une structure de bois externe.	modele	bois
Les autres types de fragments proviennent d'un mince recouvrement de mortier appliqué sur une pièce de bois et sur un assemblage de plusieurs pièces.	mortier	bois
récoltés dans ce lot sont un gros clou en fer forgé très corrodé et une partie de tuyau de pipe en terre cuite fine argileuse blanche non vernissée.	pipe	terre
d) Bague (incomplète) dont l'anneau est en laiton et le chaton de forme ovale, en verre opaque mauve.	bague	laiton

os travaillé bois travaillé andouiller travaillé ivoire travaillé manche de couteau en os hampe en os (frag.	couteau	bois
20 Le moulin à scie est inscrit sur les plans historiques à partir de 1921, mais la documentation suggère que le bois arrivait par la draye dès 1914 (ANQC, SHS Doc.	scie	bois
J - Fragment de rebord de bol ou de tasse en faïence blanche (BjFj-143-3C10).	bol	blanche
ACTIVITÉS ET OBJETS : corde de bois (2,60 m) à 3 m de l'entrée 2 pots de verre autour du campement.	verre	bois
en fil de métal traversant entièrement le petit disque en.	disque	metal
un isolateur électrique en grès grossier, du verre à vitre teinté bleu et des tessons de verre vert foncé dont des goulots finis à la pince et un cul de type Houtard.	vitre	verre
et de grès, charbons de bois et grosses pierres des champs en moyenne quantité dans la masse, quelques fragments de brique et lentilles de mortier pulvérisé dans la masse, lit de moellons subanguleux parfois fracturés disposés à plat (calcaire et grès) en surface.	lit	bois
LAG-13-01, sous-opération 3B, trace de mortier visible sur certains blocs de pierre qui semblent former une assise (LAG1301-.	mortier	pierre

ANNEXE B Tableaux d'évaluation de la précision de l'extraction des relation de fiches patrimoniales

Tableau B.1 Évaluation de 100 types de site identifiés

Nom du site	Type	Relation exacte (1/0)
Maison Édouard-Tremblay	House	1
Église de Saint-Jacques-le-Majeur	Church	1
Maison provinciale des Soeurs de l'Immaculée-Conception	House	1
Maison Leblond-Tanguay	House	1
École Sainte-Ursule	School	1
Maison Thomas-McGreevy	House	1
Maison Lacharité	House	1
Édifice de la National School	School	0
Maison Janvier-Arthur-Vaillancourt (1)	House	1
Maison Félix-Bidégaré	House	1
Hôtel Fédéral	Hotel	1
Église de La Décollation-de-Saint-Jean-Baptiste	Church	1
Maison Adolphe-Faust	House	1
Chapelle du troisième cimetière de Sainte-Anne-de-Beaupré	Cemetery	0
Ancienne école	School	1
Phare de l'île aux Perroquets	House	0
Hôtel Marcheterre	Hotel	1
Magasin J. O. Hubert	Store	1
Entrepôt Robert-Gillespie II	House	0
Hôtel Silver Maple	Hotel	1
École-Chapelle	School	1
Maison du capitaine Roger Desgagnés	House	1
Maison Coghlan-Rolston	House	1
Maison Maheux-Marcoux	House	1
Maison Labrèche	House	1

Église de Saint-Séverin	Church	1
Magasin-entrepôt Andrew-Macfarlane	House	1
Magasin-entrepôt Jean-Louis-Beaudry	House	1
Église Saint-Nazaire	Church	1
Ancien magasin Horace-Stewart	Store	1
Hôtel Petite-Nation	Hotel	1
Maison-magasin Lawrence-Kidd	House	1
Église Montreal Ghanaian Seventh-Day Adventist	Church	1
Ancien magasin général Nérée-Gagnon	Store	1
Église de Saint-Grégoire-de-Nazianze	Church	1
Église de Saint-Louis-de-Courville	Church	1
Charnier du cimetière de Saint-Joseph	Cemetery	0
Maison Bouthillier	House	1
Église de Saint-Timothée	Church	1
Maison du notaire Cimon	House	1
Église Saint-Anselme	Church	1
Ancienne église méthodiste	Church	1
Maison Alonzo-Boisvert	House	1
Maison Marc-Aurèle-De Foy-Suzor-Coté	House	1
Maison Joseph-Charlebois	House	1
Maison Étienne-Tremblay	House	1
Maison Isidore-Vallée	House	1
Ancienne église anglicane de Forestville	Church	1
Maison Higginson	House	1
Église de Sainte-Marthe	Church	1
École du 10e rang	School	1
Maison Ritchie	House	1
Cimetière Greenwood	Cemetery	1
Maison du père Fafard	House	1
Maison Thibodeau	House	1
Maison David-MacLaren	House	1
Maison du Docteur-Jules-Dorion	House	1
Maison Willson	House	1
Maison Brunelle	House	1

Maison Dunière	House	1
Maison Dupéré	House	1
Magasin général Martin	Store	1
Calvaire du cimetière de Sainte-Anne	Cemetery	0
Église Seventh Day Adventist Central Hispana	Church	1
Maison Jean-Boudreau	House	1
Magasin-entrepôt Hector-Lamontagne	House	0
Maison Louis-Carmichael	House	1
Église de Saint-Pierre-du-Sud	Church	1
Maison Steinbach	House	1
Site patrimonial de l'Église-Saint-Nom-de-Jésus	Church	1
Ancienne École Saint-Sacrement	School	1
Maison Célestin-Ménard	House	1
Ancienne école d'Old-Harry	School	1
Maison Wolfe	House	1
Maison Lamothe	House	1
Maison Shaughnessy	House	1
Pont Félix-Gabriel-Marchand	Bridge	1
Maison J.-J.-Codville	House	1
Maison du docteur Toutant	House	1
Maison du 9e Rang	House	1
Pont Romain-Caron	Bridge	1
Maison Desrosiers	House	1
Parc des chutes d'Armagh	Park	1
Maison Chabot-Bédard	House	1
Site patrimonial de l'Entrée-Supérieure-de-l'Ancien-Canal-de-Beauharnois	Canal	0
Entrepôt Gillespie-Moffatt I	House	0
Église de Saint-Elzéar	Church	1
Maison Gordon-Wills	House	1
Église Saint Paul	Church	1
Maison Henry-Marshall-Fulford	House	1
Maison Trottier-Lanouette	House	1
Maison Charles-Marié	House	1

Pont des Draveurs	Bridge	1
Entrepôt Pierre-Del Vecchio	House	0
Magasin-entrepôt John-Pratt	House	0
Église de Saint-Jovite	Church	1
Cimetière de Gray Valley	Cemetery	1
Cimetière Baldwin-Wheeler	Cemetery	1
Maison Labissonnière	House	1
Magasin McGlashan	Store	1

Tableau B.2 Évaluation de toutes les instances entre personne et site de la relation "*built*"

Subjet	Objet	Relation exacte (1/0)
Albiny Ethier	Hôtel Isaïe-Godmer	1
Abondius Juteau	Maison Abondius-Juteau	1
Abraham Joseph	Maison Abraham-Joseph	1
Adélard Laviolette	Maison Adélard-Laviolette	1
Alexis Pilon	Maison Alexis-Pilon	1
Arthur Hatin	Maison Arthur-Hatin	1
Elzéar Auclair	Maison Auclair-Verret	1
Willy Bigué	Maison Bigué	1
Charles Dubois	Maison Charles-Dubois	1
Charles Simard	Maison Charles-Simard	1
Charles Tremblay	Maison Charles-Tremblay dit Gadelle	1
Napoleon Constantine	Maison Constantin	1
Jean-Baptiste Proulx dit Clément	Maison du Centenaire	1
Ferdinand Paquette	Maison du forgeron	1
Jonathan Noble	Maison du sacristain de Saint-Jacques-le-Majeur	1
Elzéar Lafleur	Maison Elzéar-Lafleur	1
Joseph Bruneau	Maison Frank-Mineault	1

Gilles Ménard	Maison Gilles-Ménard	1
Charles Gourde	Maison Gourde	1
Matha Grenier	Maison Grenier-Constantineau	1
Gregoire Dufour	Maison Grégoire-Dufour	1
Jean-Baptiste Garneau	Maison Guillot	1
Honoré Cadieux	Maison Honoré-Cadieux	1
John Amy	Maison John-Amy	1
John Shouldice	Maison John-Shouldice	1
Joseph Morin	Maison Larchevêque-Lelièvre	1
Leonis Lavoie	Maison Leonis-Lavoie	1
Léo Daoust	Maison Léo-Daoust	1
Napoleon Grant	Maison Malcolm	1
Malcom Nolet	Maison Malcom-Nolet	1
J. A. Perkins	Maison Olidor-Guitard	1
Oscar Dufour	Maison Oscar-Dufour	1
Samuel Ouellette	Maison Ouellette	1
Ovila Rompré	Maison Ovila-Rompré	1
Parker	Maison Parker-Lahaie	1
Maurice Sansfaçon	Maison Sanfaçon-Garneau	1
Simard	Maison Simard-Tremblay	1
Charles-Ulric Tartre	Maison Tartre	1
Onésime Trottier	Maison Thibault	1
Jean-Baptiste Thibeault	Maison Thibeault	1
Théophile Grenier	Maison Théophile-Grenier	1
Montefiore Joseph	Maison Tiddlewinks	1
Britishman George Norris Trent	Maison Trent	1
Louis Trottier	Maison Trottier-Lanouette	1
Ubald Déziel	Maison Ubald-Déziel	1
Alphonse Verret	Maison Verret-Auclair	1
William H. McConnell	Maison William-H.-McConnell	1
Xavier Giroux	Maison Xavier-Giroux	1

Thomas Patchel	Maison Ébacher	1
Édouard Tremblay	Maison Édouard-Tremblay	1
Édouard Villeneuve	Maison Édouard-Villeneuve	1
Évariste Simard	Maison Évariste-Simard	1
Jean-Baptiste Hébert	Manoir Hébert	1
Louis Gagnon	Moulin du Ruisseau-Michel	1
Cyrille Côté	Pont Détroit	0
Augustus Brown	Pont Félix-Gabriel-Marchand	1
François Valade	Site patrimonial de la Maison-Thomas-Whitehead	1
Elzéar Métivier	Église de Notre-Dame-Auxiliatrice-de-Buckland	1
Saint-Damien-de-Buckland	Église de Saint-Martin	0
Elzéar Métivier	Église de Sainte-Sabine	1
Poudrier	Église Saint-Cajetan	0
Simon Shufelt	Église West Brome United Church	1
McCool	Ancien magasin général McCool	1
Horace Stewart	Ancien magasin Horace-Stewart	1
He	Hôtel British	0
Emery	Magasin Allaire	1
Pierre Lacasse	Magasin Marcel-Roy	1
Adolphe	Maison Adolphe-Leblanc	1
Anne Hamilton	Maison Anne-Hamilton	1
George Anderson	Maison Atholl-Doune	1
Bolton McGrath	Maison Bolton-McGrath	1
Jacques Perras	Maison Bélisle	1
Joseph Hilaire Chasles	Maison Chasles	1
Jules Buteau	Maison des docteurs	1
Ephraim Guimont	Maison Ephraïm-Guimond	1
Ernest Paradis	Maison Ernest-Paradis	1
Ferdinand Rollin	Maison Ferdinand-Rollin	1
Georges Painchaud	Maison Georges-Painchaud	1
John Fernie Higginson	Maison Higginson	1

James Coleman	Maison James-Coleman	1
James J. McArthur	Maison James-McArthur	1
James McConnell	Maison James-McConnell	1
James McGarry	Maison James-McGarry	1
James Mulligan	Maison James-Mulligan	1
Louis Viger	Maison Louis-Viger	1
Napoleon Blais	Maison Napoléon-Blais	1
Polyxène Beaudry	Maison Prolyxène-Beaudry	1
Robert Armor	Maison Robert-Armour	1
Lamontagne	Maison Théodore-Jean-Lamontagne	1
Pierre Del Vecchio	Maison-magasin Pierre-Del-Vecchio II	1
Paul Boileau	Église de Péribonka	1

Tableau B.3 Évaluation de 100 instances entre site et année de la relation "*built in*"

Sujet	Object	Relation exacte (1/0)
Académie d'Inverness	1889	1
Auberge Lakeview	1874	1
Barrage Jean-Guérin	1911	1
Bibliothèque de Chibougamau	1961	1
Bibliothèque municipale de Saint-Malachie	1917	1
Bijouterie Drainville	1924	1
Bloc Pagé	1909	1
Cimetière de Saint-Nérée	1883	0
Cinéma Impérial	1913	1
Eglise Saint-James Carmichael	1909	1
Magasin Williams	1872	1
Maison Adélard-Laviolette	1886	1
Maison Alexis-Pilon	1905	1
Maison Allard	1905	1
Maison Ambroise-Goulet	1885	1
Maison Antoine-Lambert	1885	1

Maison Armstrong	1911	1
Maison Arthur-Hatin	1925	1
Maison Basile-Carrière	1888	1
Maison Beaumier	1850	1
Maison Bigué	1920	1
Maison Boulet-Renaud	1931	1
Maison Bélanger-Giroux	1847	1
Maison Charest-Leboeuf	1818	1
Maison Charles-Simard	1905	1
Maison Charles-Tremblay dit Gadelle	1935	1
Maison Côme-Cartier	1904	1
Maison des Jésuites	1684	1
Maison des Jésuites	1684	1
Maison des Prêtres-Chaumont	1884	1
Maison des Soeurs du Bon-Pasteur	1972	1
Maison du capitaine J.A.Z. Desgagnés	1844	1
Maison du capitaine Jean-Paul Desgagnés	1944	1
Maison du capitaine Roger Desgagnés	1944	1
Maison du docteur Louis-Joseph-Janelle	1916	1
Maison du docteur Léonard Frève	1820	1
Maison du Docteur-Beauchemin	1900	1
Maison du forgeron	1901	1
Maison Edward-L.-Quirk	1892	1
Maison Ernest-Fleury	1897	1
Maison Euloge-Tremblay	1911	1
Maison Fleurent	1910	1
Maison Fortin	1927	1
Maison Félix-Bidégaré	1843	1
Maison Gilles-Ménard	1957	1
Maison Goulet	1898	1
Maison Grenier-Constantineau	1876	1
Maison Hermel-Tremblay	1891	1
Maison Honoré-Cadieux	1936	1
Maison James-Monk	1803	1

Maison John-Foran	1858	1
Maison Lamarre	1740	1
Maison LaRue-Bouchard	1895	1
Maison Laurent-Létourneau	1906	1
Maison Lefebvre-Veillette	1890	1
Maison Marguerite-Hay	1853	1
Maison Nazaire-Boudreault	1843	1
Maison Nobert	1896	1
Maison Nérée-Beauchemin	1867	1
Maison Odilon-Guindon	1898	1
Maison Oscar-Dufour	1912	1
Maison Papineau	1785	1
Maison Patenaude	1723	1
Maison Perrine-Charles-Cherrier	1817	0
Maison René-Lévesque	1905	1
Maison René-Lévesque	1905	1
Maison René-Robert	1887	1
Maison Richard	1932	1
Maison Riverview	1865	1
Maison Stillwaters	1862	1
Maison Taker	1896	1
Maison Tartre	1905	1
Maison Thomas-Reilly	1908	1
Maison Tremblay	1940	1
Maison Trottier-Lanouette	1854	1
Maison Ubald-Déziel	1887	1
Maison Veillette	1910	1
Maison Verret-Auclair	1924	1
Maison William-Bourgeau	1886	1
Maison à l'Enseigne-du-Patriote	1814	1
Maison Édouard-Tremblay	1928	1
Maison Évariste-Chénier	1895	1
Maison Évariste-Simard	1917	1
Moulin no 2	1903	1

Moulin à vent Dansereau	1822	0.5
Moulin à vent de Pointe-Claire	1709	1
Pont Bordeleau	1875	1
Pont Champagne	1941	1
Pont couvert de Powerscourt	1862	1
Pont couvert de Saint-Mathieu	1936	1
Pont couvert Grandchamp	1883	0.5
Pont de la rivière Abénakis	1949	1
Pont du ruisseau Aubin	1924	1
Pont du ruisseau Leblanc	1926	1
Pont Félix-Gabriel-Marchand	1898	1
Pont Galipeau	1951	1
Pont Narrows	1881	1
Pont Rouge	1928	1
Pont Sainte-Catherine	1925	1
Pont Taschereau	1916	1

ANNEXE C Liste des types de patrimoines immobiliers associés à la taxonomie de schema.org

La liste suivante contient les différents types de patrimoines immobiliers que nous avons pu trouvé dans la taxonomie de schema.org :

- Accomodation
- Airport
- Apartment
- Aquarium
- Attorney
- Bakery
- Beach
- Brewery
- Bridge
- Campground
- Canal
- Cemetery
- Church
- City
- Continent
- Corporation
- Country
- Crematorium
- Dentist
- Distillery
- Electrician
- Florist
- Hospital
- Hotel
- House
- Library
- Locksmith
- Motel
- Mountain

— Museum
— NightClub
— Notary
— Park
— Pharmacy
— Physician
— Playground
— Plumber
— Pond
— Reservoir
— Residence
— Resort
— Restaurant
— Room
— School
— State
— Store
— Waterfall
— Zoo