EGC 2022

Actes de l'atelier GAST – Gestion et Analyse de données Spatiales et Temporelles

Thomas Guyet (Centre INRIA Lyon)
Ludovic Moncla (LIRIS CNRS, INSA Lyon)
Éric Kergosien (GERiiCO, Université Lille)
Christian Sallaberry (LIUPPA, Université de Pau et des Pays de l'Adour)

http://gt-gast.irisa.fr/gast-2022/

Mardi 25 janvier 2022, Blois

Préface

Le septième atelier « Gestion et Analyse des données Spatiales et Temporelles » (GAST) est associé à EGC'2022. Cet atelier, s'appuyant sur le Groupe de Travail GAST, regroupe des chercheurs, du domaine académique et de l'industrie, qui s'intéressent aux problématiques liées à la prise en compte de l'information temporelle ou spatiale - quantitative ou qualitative - dans leurs processus de gestion et d'analyse de données (méthodes et application d'extraction, de gestion, de représentation, d'analyse et de visualisation d'informations).
Ces actes regroupent sept soumissions présentées à l'atelier GAST'2022 :

- *Sémantisation du Modèle e-CISE pour l'Interopérabilité de Données Maritimes Géo-localisées*, Nathalie Aussenac-Gilles, Catherine Comparot, Antoine Dupuy, Ronan Tournier, Ba-Huy Tran, Cassia Trojahn
- *Impact de la Pollution de l'Air sur la Mortalité : État des Lieux et Approches*, Hana Sebia, Tarik Boumaza, Marie Le Guilly, Mohand-Saïd Hacid, Delphine Maucort-Boulch
- *Visualisation spatio-temporelle de données de mobilité touristique extérieures*, Maxime Masson, Cécile Cayèré, Marie-Noëlle Bessagnet, Christian Sallaberry, Philippe Roose, Cyril Faucher
- *Fouille de séquences temporelles avec l'AFC et l'outil GALACTIC*, Salah Eddine Boukhetta, Christophe Demko, Karell Bertet, Jérémy Richard et Cécile Cayèré
- *Apprendre à combiner l'information géographique pour générer une carte généralisée*, Azelle Courtial, Guillaume Touya
- *Recherche de motifs fréquents dans un multi-graphe étiqueté et orienté*, Application aux graphes spatio-temporels, Aurélie Leborgne, Ezriel Steinberg, Laurine Lafontaine, Florence Le Ber, Antoine Vacavant
- *Apprentissage de comportements à partir de données temporelles hétérogènes*, Nida Meddouri, François Rioult, Bruno Crémilleux

Ces articles montrent une large étendue des recherches actuelles à des fins de modélisation, d'extraction, d'analyse, ou de visualisation d'information, basées sur les dimensions temporelles et spatiales associées.
Nous espérons que les orateurs, les auditeurs et les lecteurs pourront interagir autour de ces sujets, que les questions et les défis associés à l'information temporelle et spatiale continueront à animer les débats. Nous tenons à remercier tous les auteurs pour leurs propositions d'articles ainsi que les membres du comité de lecture dont les retours ont été de qualité pour l'ensemble des articles.

En espérant que ces articles vous apporteront de nouvelles perspectives autour de la Gestion et l'Analyse des données Spatiales et Temporelles, nous vous souhaitons une bonne lecture.

Thomas Guyet
Éric Kergosien

Ludovic Moncla
Christian Sallaberry

Comité de lecture

Marie-Noëlle Bessagnet	LIUPPA, Pau
Jacques Fize	ERIC, Lyon
Jérome Gensel	UGA/LIG, Grenoble
Thomas Guyet	Centre INRIA Lyon
Eric Kergosien	GERIICO, Lille
Ludovic Moncla	LIRIS, Lyon
Amadéo Napoli	Inria/Loria, Nancy
Michael Ortega	UGA/LIG, Grenoble
Cyril de Runz	LIFAT, Tours
Christian Sallaberry	LIUPPA, Pau
Loic Salmon	ISEN, Brest

Sémantisation du Modèle e-CISE pour l'Interoperabilité de Données Maritimes Géolocalisées

Nathalie Aussenac-Gilles, Catherine Comparot, Antoine Dupuy, Ronan Tournier, Ba-Huy Tran, Cassia Trojahn

IRIT, Université de Toulouse, CNRS, Toulouse INP, UT3, UT1, UT2, Toulouse, France
prenom.nom ou prenom.nom-composé@irit.fr

Résumé. Une étroite coopération transfrontalière est essentielle pour la surveillance des frontières, et en particulier lors de crises maritimes, ce qui nécessite un partage d'informations transfrontalier. Le développement du *Common Information Sharing Environment* (CISE) constitue une initiative collaborative pour promouvoir le partage automatisé d'informations entre les autorités de surveillance maritime. L'adoption de CISE et de ses différentes versions est cependant limitée par sa sérialisation existante via uniquement un schéma XML. Contrairement aux ontologies, ce format est limité pour fournir une sémantique plus riche, une interopérabilité sémantique et des capacités de raisonnement sémantique. Nous présentons ici le processus de conversion de la version plus récente du modèle CISE (e-CISE) dans le format standard Ontology Web Langage (OWL), contribuant ainsi à améliorer les travaux antérieurs sur la transformation de schémas XML (XSD) en ontologies OWL.

1 Introduction

Rendre les systèmes de surveillance maritime interopérables est crucial pour la coopération entre les pays, en particulier en cas de crises maritimes dans des zones frontalières entre pays. Dans cet objectif, l'hétérogénéité entre les système nationaux et les structures de données des différents acteurs soulèvent de nombreux problèmes. En outre, la plupart des données partagées dans cette infrastructure ont une composante spatio-temporelle.

Afin de permettre aux autorités d'échanger des informations de manière automatique et sécurisée, l'environnement commun de partage d'informations (CISE) [1] a été proposé. Il fournit un cadre décentralisé et un modèle de données pour l'échange d'informations point à point entre les secteurs et les frontières. Il implique plus de 300 autorités européennes et nationales ayant des responsabilités en matière de surveillance maritime, effectuant de nombreuses tâches de surveillance opérationnelle différentes. Les autorités nationales bénéficient directement d'être connectées au réseau CISE, dans divers secteurs tels que la sûreté et la sécurité du transport maritime, le contrôle de la pêche, la pollution, et la défense. Depuis 2014, CISE est l'une des actions soutenant la mise en œuvre de la stratégie de sécurité maritime de l'Union Européenne (EUMSS).

1. http://www.emsa.europa.eu/cise.html

L'adoption du modèle de données CISE [2] et de ses différentes versions – en particulier, *Extended-CISE* (e-CISE) (Antonopoulos et al., 2020) – est cependant limitée par sa sérialisation existante en XML via un schéma XML uniquement, qui ne peut ni fournir une sémantique riche, ni garantir une interopérabilité sémantique des données ni fournir de support à un raisonnement. Or ces fonctionnalités s'avèrent très utiles pour associer des données venant de différentes sources de données CISE, les vérifier ou encore en déduire de nouveaux éléments. Un premier effort dans cette direction était la représentation ontologique du modèle de données CISE proposée dans (Riga et al., 2021). Cependant, la ressource et son processus de construction ne sont pas publiquement disponibles.

De manière pratique, le passage de données XML ou de schémas XSD vers une représentation sémantique, que ce soit en RDF, RDFS ou en OWL, est une question étudiée de longue date dans le domaine du Web sémantique. Cependant, une transformation automatique simple est reconnue rarement correcte. Ce passage se heurte à la difficulté de gérer des noeuds anonymes, de traiter la représentation de noeuds complexes, de capturer la sémantique des balises purement structurelles, ou encore de produire des constructeurs liés à la structuration (Minutolo et al., 2014; Hacherouf et al., 2017; Bedini et al., 2011; Vinasco-Alvarez et al., 2020).

Dans cet article, nous présentons un processus de conversion du modèle e-CISE en Ontology Web Langage (OWL). e-CISE étend le modèle CISE en améliorant le vocabulaire maritime de CISE et en étendant sa portée à la surveillance terrestre et à l'échange d'informations opérationnelles. Notre approche contribue à améliorer les travaux antérieurs sur la transformation de schémas XSD en ontologies OWL. Elle est basée sur l'extension et la combinaison d'outils existants. A terme, une deuxième contribution sera de mettre à disposition de tous le schéma e-Cise sous forme d'ontologie.

Ce travail est réalisé dans le cadre du projet H2020 EFFECTOR [3]. Ce projet propose un cadre d'interopérabilité et des services de fusion et d'analyse de données associés pour la surveillance maritime et la sécurité des frontières. Ainsi, EFFECTOR vise d'améliorer le processus d'aide à la décision et de favoriser la collaboration des acteurs maritimes au niveau local, régional et transnational. Le modèle de données CISE joue un rôle essentiel dans le projet car les messages échangés par les différents acteurs sont basés sur les différentes versions de ce modèle. Une représentation sémantique et des alignements entre ces différentes versions contribue donc à l'enjeu de l'interoperabilité.

Le reste de l'article est organisé comme suit : la section 2 introduit le modèle CISE, ses variantes et son extension e-CISE. La section 3 présente les principaux travaux en lien avec la génération de fichiers OWL à partir des fichiers XML/XSD. La Section 4 décrit le processus de conversion implémenté pendant que la Section 4.4 discute les difficultés rencontrées et finalement la Section 5 conclut l'article et dessine les perspectives pour les travaux futurs.

2 Modèles de données CISE et e-CISE

Le modèle de données CISE a pour ambition de servir de format pour le partage d'information de surveillance maritime entre secteurs et pays. Dans cette optique, ce modèle de données décrit sept entités principales (`Agent`, `Object`, `Location`, `Document`, `Event`, `Risk`, `Period`) et onze entités auxiliaires (`Vessel`, `Cargo`, `Operational Asset`, `Person`,

2. `http://emsa.europa.eu/cise-documentation/cise-data-model-1.5.3/`
3. `https://www.effector-project.eu/`

`Organization`, `Movement`, `Incident`, `Anomaly`, `Action`, `Unique Identifier`, `Metadata`). Le modèle CISE permet aux différentes autorités de bénéficier d'un vocabulaire commun pour décrire les événements observés. Ce modèle est décliné dans les formalismes RDF et XSD. Sur la Figure 1, les entités du modèle CISE correspondent aux hexagones non colorés.

Le modèle de données e-CISE enrichit le vocabulaire du modèle de données CISE concernant les domaines maritime et terrestre en introduisant de nouvelles associations d'entités, des capacités et un ensemble plus riche de types dans plusieurs énumérations. Entre autres, e-CISE fournit un ensemble plus riche de types de navires, d'Automatic Identification System (AIS) et de capteurs radar ; il liste également un ensemble plus complet d'anomalies et de règles maritimes, mais aussi terrestres ; il offre enfin de nouvelles capacités de détection des entités. Ce modèle est construit sur la dernière version du modèle de données CISE utilisé dans le projet EUCISE2020 (EUCISE2020, 2020). Sur la Figure 1, les entités centrales du modèle e-CISE correspondent aux hexagones de couleur, qui viennent compléter les entités du modèle CISE.

FIG. 1 – *Aperçu des vocabulaires du modèle de base CISE (entités en blanc) et du modèle e-CISE (CISE enrichi par les entités en couleur).*

Une composante clé du modèle CISE (et donc aussi de e-CISE) concerne la dimension spatiale. Pour ce faire, l'entité `Location`, une sous-classe de `Entity` est centrale. Une localisation peut être décrite de trois manières principales : en utilisant un nom de lieu, une géométrie ou une adresse. Le contexte spécifique déterminera quelle méthode de description d'un emplacement est la plus appropriée. L'ISO 19112 définit un emplacement comme "un

lieu géographique identifiable". Dans cet esprit, "Pont Neuf" et "Toulouse" sont deux emplacements identifiés sous forme de chaînes de caractères. Ce type d'identifiant est courant bien qu'il puisse être très ambigu car de nombreux endroits partagent des noms identiques ou similaires. Un emplacement est décrit par différentes propriétés et attributs, tels qu'une `Geometry` and `LocationZone`. En particulier, la propriété `Geometry` permet la définition d'une région geo-réferencée, ayant comme attributs `altitude`, `latitude`, `longitude`, `WKT` et `XMLGeometry`.

Les modèles CISE et e-CISE sont décrits dans un document de spécification (diagrammes de classes UML) et implémentés en XSD (schéma XML). Les fichiers XSD de CISE ont été produits à partir de transformations, i.e. un ensemble de règles de correspondance indiquant comment générer des éléments XSD à partir des éléments UML correspondants. Différentes options sont généralement proposées (par ex. celles décrites dans [4]). Le choix de la transformation en XSD des classes d'association pour e-Cise (1 fichier XSD par classe) fait qu'une classe d'association est décrite 2 fois (une description par classe participant à l'association).

3 De XSD à RDF/OWL : approches existantes

Le passage de données XML ou de schémas XSD vers une représentation sémantique, que ce soit en RDF ou en OWL, est une question étudiée de longue date dans le domaine du Web sémantique. En effet, nombre de ressources du web sont disponibles sous ce format semi-structuré. De plus, il existe une sérialisation XML de RDF, sous forme de langage de balises. Lorsqu'un schéma XSD est disponible et qu'une collection de documents a été construite en conformité à ce schéma, il est tentant de considérer les balises XML comme définissant les classes d'une ontologie ou le type d'entités en RDF. La conversion des documents XML décrits selon un schéma peut être alors traitée comme la définition d'instances de ces classes, ce qui revient à peupler l'ontologie définie par le schéma. Pour gérer ce processus, si l'on reste dans un format XML, on peut considérer que le problème technique est le passage d'un schéma XML à un autre. Pour cela, il existe le langage XSLT [5], boîte à outils naturelle pour convertir un schémas XML vers un autre schéma XML.

Cependant, les balises ne se situant pas toutes au même niveau d'abstraction, une transformation automatique simple est rarement efficace et correcte. Lorsque les éléments définis entre balises sont eux mêmes complexes, et relèvent de plusieurs types en lien avec plusieurs propriétés, ils peuvent même contribuer à la fois à enrichir l'ontologie et à la peupler. Dans tous les cas, le passage du XSD à des classes OWL ou des types RDF se heurte à la difficulté de gérer des noeuds anonymes, de traiter la représentation de noeuds complexes, la génération d'énumérations, la gestion des types XSD, ou encore de représenter les balises purement structurelles et non sémantique, ou encore de produire des balises liées à la structuration.

Plusieurs outils ont été proposés dans l'état de l'art. Parmi eux, les outil dit de "lifting" convertissent un schéma XML (XSD) en un schéma RDF, tel que RDFS ou OWL. Un tel outil est XML2OWL (à partir d'un document d'instance XML ou d'un XSD), ou XSD2OWL [6]

4. `https://www.ibm.com/docs/en/rational-soft-arch/9.6.1?topic=files-uml-xsd-transformations`
5. https ://www.w3.org/community/rax/wiki/XML_to_RDF_Transformation_processes_using_XSLT#Preprocessing_of_XML
6. https ://gist.github.com/pebbie/5704765

(à partir d'un XSD). Une autre option consiste à utiliser TopBraid Composer, et un plugin XSD vers OWL (notez toutefois qu'il s'agit d'un logiciel commercial). MIT Simile fournit une liste de quelques autres "RDF-izers". La Table 1 présente une synthèse des outils étudiés. Malheureusement, les outils présentés dans les articles de recherche (comme les deux que nous venons de citer) soient rarement accessibles. Il est encore regrettable que les deux autres outils génèrent des ontologies très complexes et négligent la représentation des relations n-aires entre classes. Il est aussi souhaitable d'avoir un mécanisme pour la construction automatique des descriptions des éléments (via `rdfs:comment` par exemple) à partir de la documentation. Pour répondre à ces besoins, nous avons développé un processus implémenté en Python. Il s'inspire du travail de (Aryan, 2013) et reprend en les étendant les règles de transformation telles que définies par Ontmalizer [7].

Outil	**Langage**	**Avantages**	**Inconvénients**
PIXCO (Hacherouf et al., 2017)	Java		Inaccessibilité
JANUS (Bedini et al., 2011)	Java	GUI	Inaccessibilité
CityGML (Vinasco-Alvarez et al., 2020)	Python et Java	Simplicité	Résultat complexe
Ontmalizer (Yüksel, 2013)	Java	Simplicité	Résultat complexe
XSD2RDF.py (Aryan, 2013)	Python	Simplicité	Incapacité de traiter les relations n-aires et des individus

TAB. 1 – *Comparaison des outil de conversion de XML en RDF/OWL.*

4 Processus de conversion XML en OWL

Dans ce qui suit, nous présentons un schéma du processus de conversion de XSD en OWL ainsi que les règles de transformation que nous considérons et des fragments de l'ontologie générée à partir des spécifications du schéma de e-CISE.

4.1 Schéma du processus

La Figure 2 illustre le schéma du processus de conversion XSD en OWL. Ce processus est implémenté en Python et fait appel à plusieurs sources de données externes, comme décrit ci-dessous.

Le pilotage de l'outil de transformation des données au format XSD vers le formalisme OWL se fait via l'utilisation d'un fichier Makefile. Ce fichier permet de lancement via une console des commandes de génération des ontologies CISE et e-CISE (`make cise` et `make ecise`). Le lancement d'une commande lance la génération de l'ontologie souhaitée (CISE ou e-CISE dans la version actuelle du programme). Le script correspondant effectue la lecture des sources XSD et des sources externes afin d'extraire les éléments nécessaires à la construction de l'ontologie. Pour chaque type d'élément XSD, une règle de conversion est appliquée vers le formalisme OWL.

7. `https://www.w3.org/wiki/HCLSIG/Tools\#Ontmalizer` (Yüksel, 2013)

Sémantisation du modèle e-CISE

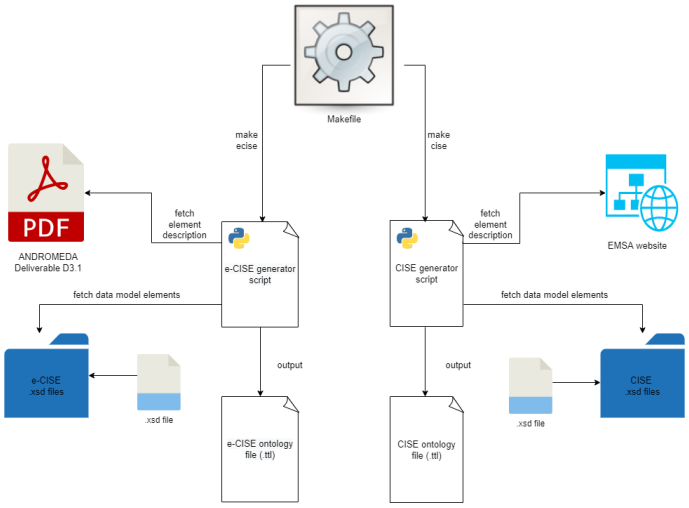


FIG. 2 – *Schéma du processus de génération des modèles OWL e-CISE et CISE à partir de leurs schémas XSD et d'autres sources externes.*

L'outil de transformation nécessite une connexion Internet pour récupérer la documentation du modèle de données CISE sur le site de l'EMSA (*European Maritime Safety Agency*). L'extraction du texte est basée sur l'analyse des patterns, incluant la structure DOM de la page web et l'ordre des paragraphes. Par conséquent, le résultat de l'extraction dépend de la typographie et de la dénomination des titres, sous-titres et de l'URL des pages. Le livrable D3.1 d'Andromeda (Antonopoulos et al., 2020) (au format PDF) est quant à lui nécessaire pour la documentation du modèle de données e-CISE. L'outil utilise la librairie PDFReader et effectue une lecture page à page ; il stocke le contenu des pages et permet de retrouver les descriptions des éléments traités. Les résultats obtenus suite au traitement des éléments XSD et à la lecture des sources Web ou PDF conduisent à la création d'un fichier au format turtle contenant les triplets RDF de l'ontologie générée.

4.2 Règles de transformation

La plupart des règles de transformation ont été inspirées de celles de l'outil Ontmalizer qui a été réutilisé aussi dans le projet Shift2Rail [8] (SprintProject, 2020). Concernant les classes d'association et les énumérations, nous avons repris la proposition de (Antonopoulos et al., 2020). Plus précisément, les classes d'association sont représentées comme des sous-classes d'une nouvelle classe appelée `AssociationClass`. Pour les énumérations, une nouvelle classe `EnumerationType` est également introduite, et les valeurs des énumérations sont des instances (individus) de cette classe. Des exemples sont donnés en Section 4.3.1.

8. https ://shift2rail.org/

Element XSD	Définitions OWL
xs :simpleType	rdfs :Datatype
xs :simpleType	rdfs :Datatype
xs :enumeration	owl :Class et owl :Individual
xs :complexType over xs :complexContent	owl :Class
xs :complexType over xs :simpleContent	owl :Class
xs :element (global) with complex type	owl :Class and rdfs :subclassOf
xs :element (global) with simple type	owl :Datatype
xs :element (local to the a type)	owl :DatatypeProperty or owl :ObjectProperty, and OWL Restrictions
xs :group	owl :Class
xs :attributeGroup	owl :Class

TAB. 2 – *Règles de conversion XSD en OWL.*

4.3 e-CISE en OWL

La Figure 3 présente un extrait de l'ontologie générée, où les classes centrales du modèle e-CISE (Figure 1) sont représentées. La Figure 4 présente un extrait de l'ontologie obtenue visualisée à l'aide du logiciel GraphBD. L'extrait concerne l'entité `Location`, une des entités centrales du modèle représentant la dimension spatiale.

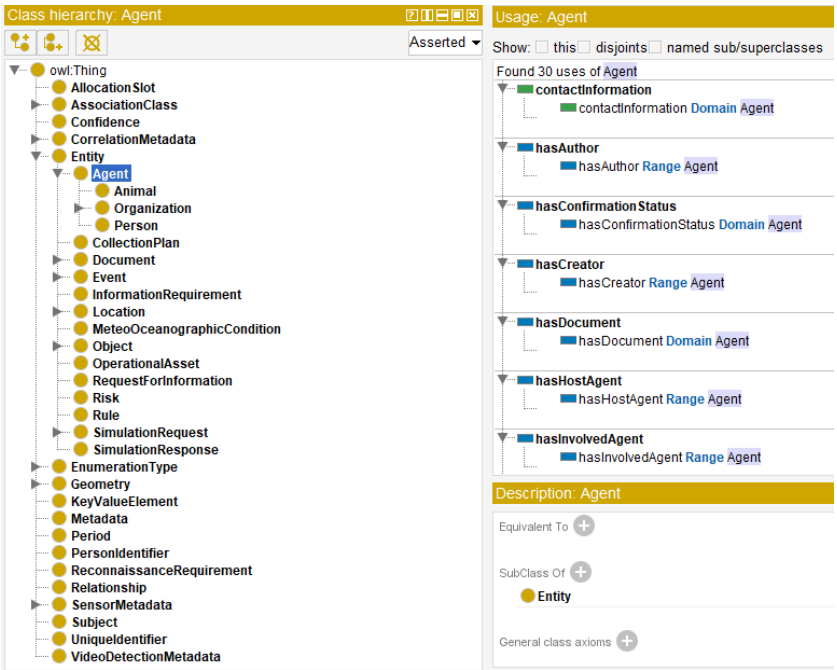


FIG. 3 – *Visualisation de l'ontologie sur Protégé.*

Sémantisation du modèle e-CISE

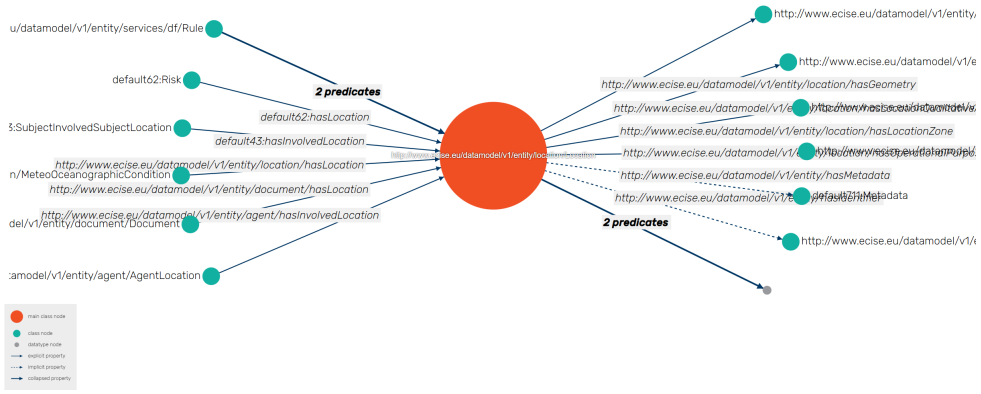


FIG. 4 – *Visualisation dans GraphDB des triplets impliquant la classe* `Location`.

Dans les sections qui suivent, des exemples de classes d'association, d'énumérations en XSD et en Turtle sont présentés.

4.3.1 Classes d'associations

Comme introduit plutôt dans l'article, le modèle e-CISE est composé d'un grand nombre de classes d'association (exprimées en XSD à l'aide du constructeur `xs:complexContent`). Nous illustrons comment ces classes sont gérées et transformées par le processus de conversion. Dans l'exemple qui suit, `InvolvedSubjectLocation` est une classe d'association dont le label est `SubjectLocation`. Cette classe implique les classes `Subject` et `Location` via deux `ObjectProperty` : `hasInvolvedSubject` et `hasInvolvedLocation`. Cette classe d'association est également composée d'attributs permettant de qualifier cette association : `LocationType`, `ThreatCertainty`, `IsThreat`, `InvolvedInIncidentRel`.

```
<xs:element name="InvolvedSubjectLocation"  minOccurs="0" maxOccurs="unbounded">
<xs:complexType>
<xs:complexContent>
<xs:extension base="rel:Relationship">
<xs:sequence>
<xs:element name="Location"
            type="location:Location"
            minOccurs="0"/>
<xs:element name="LocationType"
            type="location:LocationZoneType"
            minOccurs="0" />
<xs:element name="ThreatCertainty"
            type="utils:Confidence"
            minOccurs="0" />
<xs:element name="IsThreat"
            type="xs:boolean"
            minOccurs="0" />
<xs:element name="InvolvedInIncidentRel"
            type="incident:Incident" minOccurs="1"
            maxOccurs="unbounded" />
```

```
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>

:SubjectInvolvedSubjectLocation a owl:Class ;
    rdfs:label "SubjectLocation" ;
    rdfs:subClassOf <http://melodi.irit.fr/ontologies/ecise#AssociationClass> .

:hasInvolvedSubject a owl:ObjectProperty ;
    rdfs:domain :SubjectInvolvedSubjectAgent,
        :SubjectInvolvedSubjectLocation,
        :SubjectInvolvedSubjectObject ;
    rdfs:range :Subject .

:hasInvolvedLocation a owl:ObjectProperty ;
    rdfs:comment "Association relation" ;
    rdfs:domain :SubjectInvolvedSubjectLocation ;
    rdfs:range location:Location .

:hasLocationType a owl:ObjectProperty ;
    rdfs:domain :SubjectInvolvedSubjectLocation ;
    rdfs:range location:LocationZoneType .

:isThreat a owl:DatatypeProperty ;
    rdfs:domain :Subject,
        :SubjectInvolvedSubjectLocation ;
    rdfs:range xsd:boolean .

:hasInvolvedInIncidentRel a owl:ObjectProperty ;
    rdfs:domain :SubjectInvolvedSubjectAgent,
        :SubjectInvolvedSubjectLocation,
        :SubjectInvolvedSubjectObject ;
    rdfs:range incident:Incident .

:hasThreatCertainty a owl:ObjectProperty ;
    rdfs:domain :SubjectInvolvedSubjectAgent,
        :SubjectInvolvedSubjectLocation,
        :SubjectInvolvedSubjectObject ;
    rdfs:range utils:Confidence .
```

4.3.2 Enumérations

Le modèle e-CISE comprend un nombre important de classes d'énumération exprimées en XSD à l'aide du constructeur `xs:simpleType`. L'exemple qui suit correspond à la transformation de la classe d'énumération `ConfirmationStatusType`. Celle-ci comprend plusieurs valeurs d'énumération converties en instances (`individuals`) de la classe d'énumération. La classe d'énumération est quant à elle une sous-classe de la classe abstraite `EnumerationType`.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:entity="http://www.ecise.eu/datamodel/v1/entity/"
targetNamespace="http://www.ecise.eu/datamodel/v1/entity/">
<xs:simpleType name="ConfirmationStatusType">
CISE Legacy System. It can be local / incoming / transient
```

Sémantisation du modèle e-CISE

```
<xs:restriction base="xs:string">
<xs:enumeration value="Pending">
</xs:enumeration>
<xs:enumeration value="Failed">
DESCRIPTION: Classification of the entity is AssumeFriend
</xs:enumeration>
<xs:enumeration value="Confirmed">
DESCRIPTION: Classification of the entity is AssumeFriend
</xs:enumeration>
<xs:enumeration value="VisuallyConfirmed"></xs:enumeration>
<xs:enumeration value="Unconfirmed"></xs:enumeration>
<xs:enumeration value="Other"></xs:enumeration>
<xs:enumeration value="NonSpecified">
DESCRIPTION: Classification of the entity is not Specified
</xs:enumeration>
</xs:restriction>
</xs:simpleType>
</xs:schema>
```

```
@prefix : <http://www.ecise.eu/datamodel/v1/entity/> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

<http://melodi.irit.fr/ontologies/ecise#AssociationClass> rdfs:subClassOf owl:Thing .

:ConfirmationStatusType_Confirmed a :ConfirmationStatusType,
        owl:NamedIndividual ;
    rdfs:label "Confirmed" .

:ConfirmationStatusType_Failed a :ConfirmationStatusType,
        owl:NamedIndividual ;
    rdfs:label "Failed" .

:ConfirmationStatusType_NonSpecified a :ConfirmationStatusType,
        owl:NamedIndividual ;
    rdfs:label "NonSpecified" .

:ConfirmationStatusType_Other a :ConfirmationStatusType,
        owl:NamedIndividual ;
    rdfs:label "Other" .

:ConfirmationStatusType_Pending a :ConfirmationStatusType,
        owl:NamedIndividual ;
    rdfs:label "Pending" .

:ConfirmationStatusType_Unconfirmed a :ConfirmationStatusType,
        owl:NamedIndividual ;
    rdfs:label "Unconfirmed" .

:ConfirmationStatusType_VisuallyConfirmed a :ConfirmationStatusType,
        owl:NamedIndividual ;
    rdfs:label "VisuallyConfirmed" .

<http://melodi.irit.fr/ontologies/ecise#EnumerationType> rdfs:subClassOf owl:Thing .

:ConfirmationStatusType a owl:Class ;
    rdfs:subClassOf <http://melodi.irit.fr/ontologies/ecise#EnumerationType> .
```

4.4 Discussion

La construction des ontologies via l'utilisation de fichier sources au format .xsd soulève des difficultés de conversion particulières. Dans le cas du modèle CISE, ces difficultés sont notamment dues au nombre important d'énumérations à convertir, au manque d'information quant au nommage des classes d'association et à la conversion des contraintes de cardinalité.

Les énumérations représentent la grande majorité des conversions à traiter. Les requêtes effectuées depuis le point d'accès SPARQL au sein de GraphDB sur l'ontologie e-CISE permettent de chiffrer la proportion de classes d'énumération et d'individus peuplant celles-ci. Les classes d'association représentent un total de 141 classes sur les 269 de l'ontologie. Ces classes sont peuplées par des individus au nombre de 16 648. La plupart des axiomes de l'ontologie proviennent des énumérations du modèle de données. Contrairement aux propositions de conversion des outils existants, impliquant l'utilisation du constructeur `owl:oneOf` sur les valeurs possibles, une solution plus élégante consiste à définir une classe `EnumerationType` dont les valeurs possibles énumérées sont des instances.

Une autre difficulté lors de la conversion de ces modèles concerne les classes d'association. En effet, le nommage de ces classes n'est pas spécifié dans les fichiers sources .xsd. Deux solutions sont alors possibles pour obtenir un résultat de conversion en accord avec les spécifications : reconstruire le nommage pour qu'il corresponde aux spécifications du modèle de données, en spécifiant dans un fichier le nom de la classe d'association pour deux classes données ; ou alors effectuer la conversion en s'appuyant uniquement sur le nommage spécifié dans les fichiers .xsd. Le premier cas implique une programmation spécifique aux cas des modèles de données CISE et e-CISE. Dans le second cas, il est possible de proposer une solution plus générique qui serait utilisable pour la génération d'ontologies à partir d'autres modèles de données. Ces problèmes de nommage ont été identifiés lors d'une validation manuelle des modèles générés, par la vérification de leur conformité à leur description dans les documents PDF.

Enfin, la conversion des contraintes de cardinalité des modèles de données a soulevé des points de réflexion quant à la pertinence de leur retranscription dans un modèle sémantique géré par l'hypothèse du monde ouvert. En effet, à cause de cette hypothèse, les contraintes de cardinalités ne peuvent indiquer que des maximums ou minimums possibles, mais en aucun cas on ne peut garantir qu'un propriété soit présente avec une cardinalité n. Suivant l'utilisation prévue de l'ontologie générée, les contraintes sont gérées par des systèmes annexes à l'ontologie (à la réception d'un message CISE ou e-CISE par exemple) et avant l'utilisation de cette ontologie. Il est intéressant d'ajouter une option permettant de choisir si les contraintes de cardinalité doivent être retranscrites ou non lors de la génération des ontologies. Les résultats des conversions de ces contraintes sont représentés par des restrictions OWL sur les éléments de type `owl:ObjectProperty` et `owl:DataProperty`.

Finalement, la conversion soulève un dernier problème que nous n'avons pas traité et qui reste à améliorer. Il s'agit du traitement de sources externes de données servant à enrichir la couche terminologique des ontologies (noms des classes et propriétés, labels et commentaires). Pour certaines classes d'association et pour certaines valeurs d'énumérations, l'extraction de termes des tableaux présents dans ces sources exigerait l'utilisation d'outils d'extraction d'information plus sophistiqués que ceux que nous avons retenus.

5 Conclusions

Cet article a présenté un processus de conversion de fichiers XSD du modèle de données e-CISE en langage OWL. Nous avons discuté les difficultés rencontrées et les pistes pour les améliorer. Comme travaux futurs, nous envisageons de poursuivre plusieurs pistes : gérer les versions des ontologies en utilisant des vocabulaires dédiés ; rendre le code générique et modulaire ; générer des alignements entre les versions ontologiques des modèles CISE et e-CISE ; rendre le code et les ontologies publiquement disponibles.

Remerciements

Ces travaux sont financés par le programme Horizon 2020 d'innovation et de recherche de l'Union Européenne avec l'accord de financement No. 883374. Ce document ne reflète que la vision des auteurs et la REA (Research Executive Agency) ainsi que la Commission Européenne ne peuvent être tenue responsables pour tout usage de l'information qu'il contient.

Références

Antonopoulos, S., M. Tsogas, M. Moutzouris, A. Kostaridis, A. Aggelis, et L. Perlepes (2020). D.3.1 e-CISE Data Model description. Project Deliverable D3.1, EU.

Aryan, R. (2013). Converting xml schema to owl in python.

Bedini, I., C. Matheus, P. Patel-Schneider, A. Boran, et B. Nguyen (2011). Transforming xml schema to owl using patterns. In *2011 IEEE Fifth International Conference on Semantic Computing*, pp. 102–109.

EUCISE2020 (2020). Eucise2020 : Technical specifications. deliverable 4.3, revision 1, annex b. Technical report, EUCISE Data Model.

Hacherouf, M., S. N. Bahloul, et C. Cruz (2017). Transforming `XML` schemas into `OWL` ontologies using formal concept analysis. *Software & Systems Modeling 18*, 2093–2110.

Minutolo, A., A. Esposito, M. Ciampi, M. Esposito, et G. Cassetti (2014). An automatic method for deriving OWL ontologies from XML documents. In *2014 Ninth International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, Guangdong, China, November 8-10, 2014*, pp. 426–431. IEEE Computer Society.

Riga, M., E. Kontopoulos, K. Ioannidis, S. Kintzios, S. Vrochidis, et I. Kompatsiaris (2021). EUCISE-OWL : an ontology-based representation of the common information sharing environment (CISE) for the maritime domain. *Semantic Web 12*(4), 603–615.

SprintProject (2020). D4.2 A lightweight solution to automate the generation of ontologies, mappings and annotations (C-REL). Project Deliverable D4.2, EU.

Vinasco-Alvarez, D., J. S. Samuel, S. Servigne, et G. Gesquière (2020). From CityGML to OWL. Technical report, LIRIS UMR 5205.

Yüksel, M. (2013). A semantic interoperability framework for reinforcing post market safety studies. Technical report, Middle East Technical University.

Summary

Close cooperation across borders is essential for border surveillance, and in particulier during in case of maritime crises, which requires cross-border information sharing. Existing efforts from the Common Information Sharing Environment (CISE) constitutes a collaborative initiative for promoting automated information sharing between maritime monitoring authorities. The adoption of CISE and its different versions is however limited by the existing serialisation using only XML Schema. This format fails to deliver richer semantics, semantic interoperability and semantic reasoning capabilities. We discuss here the process of converting the most recent CISE model (e-CISE) in the Ontology Web Langage (OWL) stadard format, thus contributing to improve previous work on transforming XSD schemas into OWL ontologies.

Impact de la Pollution de l'Air sur la Mortalité : État des Lieux et Approches

Hana Sebia*, Tarik Boumaza**
Marie Le Guilly***, Mohand-Saïd Hacid****, Delphine Maucort-Boulch♯

*hana.sebia@etu.univ-lyon1.fr,
Université Lyon 1
**tarik.boumaza@etu.univ-lyon1.fr,
Université Lyon 1
***marie.le-guilly@univ-lyon1.fr,
LIRIS UMR 5205 CNRS, Université Lyon 1
****mohand-said.hacid@univ-lyon1.fr,
LIRIS UMR 5205 CNRS, Université Lyon 1
♯delphine.boulch@chu-lyon.fr
CNRS, UMR5558, Laboratoire de Biométrie et Biologie Évolutive
France

Résumé. L'OMS estime que la pollution de l'air contribue à 7 millions de décès prématurés par an. Notre recherche a pour objectif l'évaluation de ses effets sur la mortalité dans le département du Rhône (France). L'ensemble des données sur la mortalité disponibles entre 2007 et 2019 ont été analysées. En mesurant l'exposition à la pollution de la population, en nous intéressant aux polluants considérés comme les plus dangereux pour l'Homme d'après l'OMS (NO, NO_2, O_3, $PM_{2.5}$ et PM_{10}), et en utilisant des méthodes d'analyse linéaires et non linéaires, on démontre une corrélation significative. La liaison entre les deux phénomènes suit une tendance linéaire positive (particulièrement pour les PM et NO_2). On aboutit à un $\rho^2 = 0.44$ pour le NO_2, signifiant que 44% de la mortalité est liée à ce polluant. Nous avons également montré qu'une augmentation de 1% de ce polluant induisait une hausse de mortalité de 0.87%.

1 Introduction

D'après l'Organisation Mondiale de la Santé, la pollution de l'air est aujourd'hui le principal risque environnemental pour la santé dans le monde. L'OMS estime qu'en cumulant pollution intérieure et pollution extérieure, plus de 7 millions de décès survenus en 2012 sont liés à la pollution de l'air (dont 2.6 millions pour la pollution extérieure)OMS (2014). Par ailleurs, les flux de patients sont en constante progression dans les hôpitaux. De plus, la variation de ceux-ci peut induire une surcharge, du fait des contraintes matérielles (en particulier des lits de réanimations) et humaines.

Nous nous sommes alors fixés pour objectif l'étude des deux phénomènes. On se pose ainsi les questions suivantes : existe-t-il une corrélation entre la pollution extérieure et la mortalité dans le département du Rhône ? Peut-on prédire l'évolution de la mortalité en fonction de la pollution de l'air observée ?

Pour cela, nous avons d'abord effectué un travail sur les données de pollution ainsi que sur les chiffres de la mortalité. Ensuite, nous avons étudié les méthodes les plus pertinentes qui nous permettraient de démontrer un lien entre les deux phénomènes étudiés.

2 État de l'art

À ce jour, de nombreuses études ont démontré les effets néfastes de la pollution sur la mortalité.

Un programme nommé Erpurs (Campagna et al., 2003) a été mis en place en Île-de-France (France) suite à l'épisode de pollution de janvier-février 1989 afin d'évaluer la corrélation entre les variations temporelles des niveaux de pollution et le nombre quotidien de décès disponible de 1987 à 1998. Les polluants étudiés sont le NO_2, les FN (fumées noires), les PM_{13} et le SO_2. L'indicateur de mortalité considère les causes principales et immédiates du décès. L'humidité et la température, les niveaux de pollens ainsi que les épidémies de grippe ont été pris en compte dans les méthodes d'analyse utilisées - GAM (Hastie et Tibshirani, 1987) et GLM(Nelder et Wedderburn, 1972) -. Parmi les résultats montrés par l'étude, on peut observer une augmentation pouvant aller jusqu'à 4.7% pour la mortalité pour causes respiratoires, en rapport avec les particules fines lorsque la pollution passe d'un niveau faible à médian.

D'autres recherches évaluant les effets des polluants atmosphériques ont montré que ceux-ci étaient d'importants facteurs contribuant à l'augmentation des maladies respiratoires et à la mortalité prématurée (Hamra et al., 2014). Les effets néfastes sur la santé comprennent également une augmentation des hospitalisations pour maladies pulmonaires chroniques obstructives (Moolgavkar, 2000) ainsi qu'un risque accru de maladies cardio-vasculaires et de cancer du poumon (Valavanidis et al., 2014).

3 Méthode

3.1 Données de pollution

3.1.1 Recueil des mesures pour chaque polluant

Les données de pollution exploitées sont en libre accès et disponibles sur l'API ATMO Auvergne-Rhône Alpes rec. Les polluants suivants, jugés comme les plus dangereux pour la santé Valavanidis et al. (2016), ont été sélectionnés : CO : monoxyde de carbonne ; NO : monoxyde d'azote ; NO_2 : dioxyde d'azote ; O_3 : trioxygène (ozone) ; SO_2 : dioxyde de souffre ; $PM_{2.5}$: particules fines de diamètre inférieur à 2,5 μm ; PM_{10} : particules fines de diamètre inférieur à 10 μm.

Pour chaque polluant, nous avons recueilli les mesures effectuées par toutes les stations situées dans le département du Rhône de 2007 à 2019 (données mensuelles) et de 2018 à 2019 (données quotidiennes). Les mesures des CO et SO_2 étant trop incohérentes (valeurs négatives expliquées par l'appareillage ou les mauvaises conditions météorologiques d'après l'ATMO), ou incomplètes, nous avons retenu les polluants suivants : NO, NO_2, O_3, $PM_{2.5}$, PM_{10}.

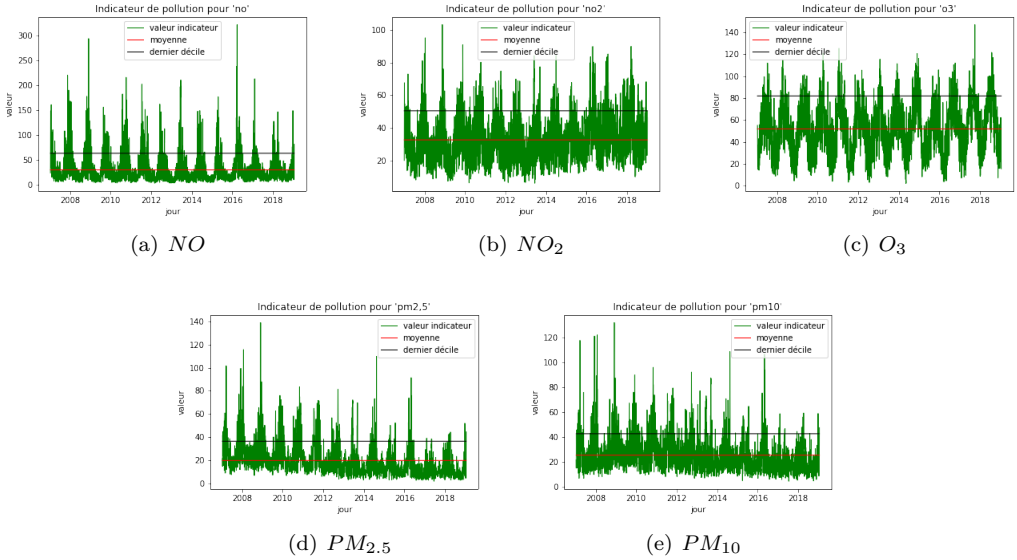
3.1.2 Données démographiques

Nous avons ensuite recueilli les données démographiques des communes où se trouvent les stations. Elles se basent sur les populations légales des communes en vigueur à compter du 1er janvier 2021 (date de référence statistique : 1er janvier 2018) INSEE (2020b). En prenant comme hypothèse que la population de chaque commune est stable relativement à la population totale

de la métropole, on peut construire un indicateur de pollution pour le département du Rhône pour n stations :

$$indicateur = \frac{\sum_{i=1}^{n} mesure_{station_i} * population\_commune_{station_i}}{population\_totale}$$

Cet indicateur permet de représenter l'exposition à la pollution de la population. En effet, chacune des stations mesure la pollution extérieure mais ne prend pas en compte l'absorption de celle-ci. Cet indicateur, bien qu'imparfait car il considère que la population d'une commune est exposée à la même quantité de pollution à un instant donné, fournit une vision globale de l'exposition. Lévolution de cet indicateur est représenté sur la figure 1.



(a) NO (b) NO_2 (c) O_3

(d) $PM_{2.5}$ (e) PM_{10}

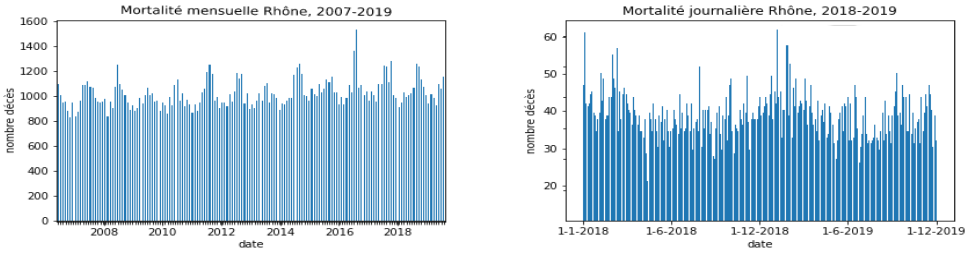
FIG. 1 – *Évolution de la valeur de l'indicateur entre 2007 et 2019*

3.2 Données de mortalité

Pour les chiffres de la mortalité, nous avons étudié les bases de données de l'INSEE pour les années de 2007 à 2019 (incluses) INSEE (2020a). Nous avons alors collecté les chiffres des décès quotidiens pour les années 2018 et 2019 (seuls chiffres quotidiens disponibles), ainsi que les décès mensuels de 2007 à 2019 (pour l'étude d'une période plus longue). Ces données sont représentées sur la figure 2. Ensuite, nous n'avons retenu que les décès des personnes ayant pour lieu de résidence une commune du département du Rhône (code INSEE 69), puisque c'est le département où l'on mesure la pollution extérieure.

3.3 Étude des corrélations

Afin de quantifier la relation entre nos deux variables (*i.e.*, la mortalité et la pollution) de manière à mettre en évidence le sens de la liaison et son intensité, nous explorons deux pistes : la linéarité et la non linéarité de la relation.



(a) *mensuelle* (2007-2019) (b) *quotidien* (2018-2019)

FIG. 2 – *Chiffres de la mortalité*

Le coefficient de corrélation de Pearson constitue une mesure de l'intensité de liaison linéaire et monotone entre deux variables. Il représente une normalisation de la covariance mesurant la tendance de deux variables à être simultanément au dessus ou en dessous de leurs espérances respectives. Or, la non-normalité des données de pollution et de mortalité ne nous permet pas de tirer de résultats fiables avec ce test paramétrique. En effet, les deux tests probabilistes (*Shapiro-Wilk* et *Anderson Darling*) effectués sur les données indiquent que les distributions ne suivent pas une loi normale (respectivement $p\text{-}value < 10^{-6} < 0.05$ et $p\text{-}value < 10^{-6} < 0.05$ pour la mortalité mensuelle ; $p\text{-}value < 10^{-4} < 0.05$ et $p\text{-}value < 10^{-3} < 0.05$ pour la mortalité quotidienne). Nous nous tournons donc vers une variante non paramétrique de ce dernier : le coefficient de corrélation de Spearman.

3.3.1 Corrélation de Spearman

Le coefficient de Spearman est calculé à partir des substitutions des valeurs observées des variables originelles par leurs rangs :

$$\rho = \frac{cov(R_X, R_Y)}{\sigma_{R_X} * \sigma_{R_Y}}$$

où $cov(R_X, R_Y)$ représente la covariance des variables de rang et σ_{R_X} et σ_{R_Y} représentent les écarts-types.

La mesure est normalisée et varie entre -1 (forte liaison négative) et 1 (forte liaison positive), 0 indiquant une absence de liaison entre les deux variables.

Ce coefficient est calculé pour les données mensuelles et quotidiennes de pollution et de mortalité.

On cherche également d'autres corrélations possibles, notamment par la sélection des données. En effet, l'impact de l'évolution du niveau de pollution n'est peut-être pas le même en considérant un niveau élevé relativement à la moyenne de celui-ci. On étudie alors deux méthodes. Dans un premier temps, on calcule la moyenne glissante de la mortalité quotidienne et des mesures de pollution sur N jours. Dans un second temps, on étudie la sélection par seuil : on ne sélectionne que les jours dont la mesure de pollution est supérieure à un seuil donné.

3.3.2 Modèles Additifs Généralisés

Une modélisation de la relation entre les indicateurs de mortalité et de pollution a également été réalisée via le développement de modèles additifs généralisés (GAM) Hastie et Tibshirani (1987). Contrairement à la méthode de corrélation de Spearman (qui mesure la monotonie de la

liaison), ces modèles permettent une plus grande flexibilité quant à la relation non linéaire que peuvent avoir certaines covariables explicatives avec une variable dépendante. Pour n variables, un modèle additif généralisé a la forme :

$$g[E(Y)] = \sum_{i=1}^{n} S_i(X_i)$$

oú E désigne l'éspérance, g est une fonction de lien et les S_i sont des fonctions de lissage. Cette modélisation prend en compte les indicateurs de pollution atmosphérique uniquement (pris individuellement).

Afin d'estimer un coefficient attribuable à chaque polluant étudié, nous nous appuyons sur le modèle statistique GLM : Generalized Linear model Nelder et Wedderburn (1972) :

$$h[E(Y)] = X\beta$$

oú h est une fonction quelconque, $E(Y)$ est l'espérance de la variable à prédire , X est la matrice des variables explicatives et β le vecteur des paramètres à estimer.
Le risque relatif (RR) est calculé à partir de l'exponentielle des éléments de β et le pourcentage de variation du risque de mortalité anticipé (MA) est calculé à partir du risque relatif : $(RR - 1) \cdot 100$. Ainsi, un pourcentage positif indique une augmentation du risque tandis qu'un pourcentage négatif indique une diminution de celui-ci. Les deux modèles ont été implémentés avec R (package `'mgcv'` [1]), le code est disponible en ligne [2]

4 Résultats

4.1 Données mensuelles

On détermine la corrélation de Spearman entre la mortalité et chacun des polluants étudiés sur la période 2007-2019. Les résultats obtenus démontrent que celles-ci ne sont pas négligeables : $\rho > 0.52$ pour NO et $\rho > 0.66$ pour NO_2 notamment (*cf.* table 1).

ρ^2 s'interprète comme une proportion de variance expliquée. Ceci permet d'affirmer alors que la pollution en NO_2 est liée à 44% des décès ($\rho^2 \approx 0.44$). Notons que la certitude p-value est inférieure à 10^{-3} pour tous les résultats présentés, ce qui indique que la probabilité que la valeur nulle soit correcte est très faible (et que les résultats sont aléatoires).

	NO	NO_2	O_3	$PM_{2.5}$	PM_{10}
mortalité	0.527262	0.665121	-0.478647	0.238333	0.238671

TAB. 1 – *Coefficients de corrélation de Spearman entre les polluants et la mortalité mensuelle 2007-2019*

Le développement des modèles additifs généralisés pour les données mensuelles montre une liaison significative (*edf* avec *p-value* $< 10^{-3}$) entre la mortalité et l'indicateur de pollution pour le NO_2 . La courbe suit une tendance linéaire positive : $edf \approx 1$ (*cf.* figure 3).

Les autres courbes indiquent des corrélations non linéaires mais monotones ($edf > 2$), à l'exception du polluant O_3 pour lequel on n'observe pas de corrélation significative. Un modèle

1. `https://cran.r-project.org/web/packages/mgcv/mgcv.pdf`.
2. `https://forge.univ-lyon1.fr/m1-grp26/impact-pollution-mortalite`

Impact de la pollution de l'air sur la mortalité

	NO	NO_2	O_3	$PM_{2.5}$	PM_{10}
mortalité	0.188265	0.197158	-0.219550	0.150273	0.106504

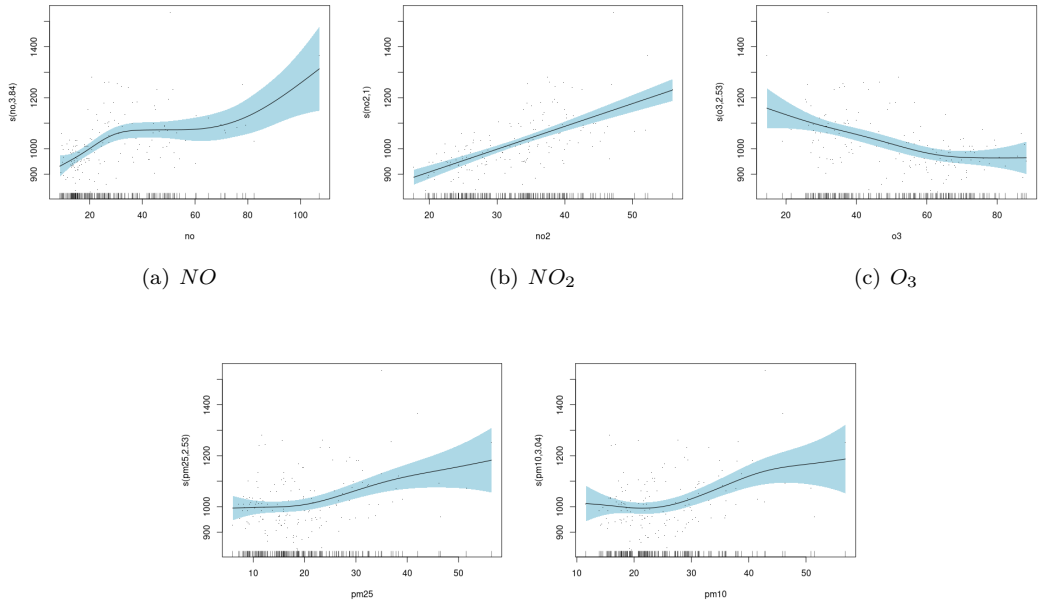
TAB. 2 – *Coefficients de corrélation de Spearman entre les polluants et la mortalité quotidienne 2018-2019*

polluant	r	p-value	seuil	N jours	L données	méthode
NO	0.53354	$< 10^{-5}$	0	7	730	moyenne glissante
NO_2	0.51777	$< 10^{-5}$	0	7	730	moyenne glissante
O_3	0.22844	0.039	87	5	82	sélection par seuil
$PM_{2.5}$	0.46977	$< 10^{-5}$	0	7	730	moyenne glissante
PM_{10}	0.44715	$2 \cdot 10^{-5}$	29	7	86	sélection par seuil

TAB. 3 – *Coefficients de corrélation de Spearman entre les polluants et la mortalité quotidienne 2018-2019 et méthode associée*

linéaire généralisé appliqué au polluant NO_2, paramétré par une loi de Poisson, indique un coefficient 0.0086629. D'où :

$$MA_{NO_2} = (RR_{NO_2} - 1) \cdot 100 = (e^{0.0086629} - 1) \cdot 100 \approx 0.87\%$$



(a) NO (b) NO_2 (c) O_3

(d) $PM_{2.5}$ (e) PM_{10}

FIG. 3 – *Modèle GAM pour la mortalité mensuelle et chacun des polluants étudiés*

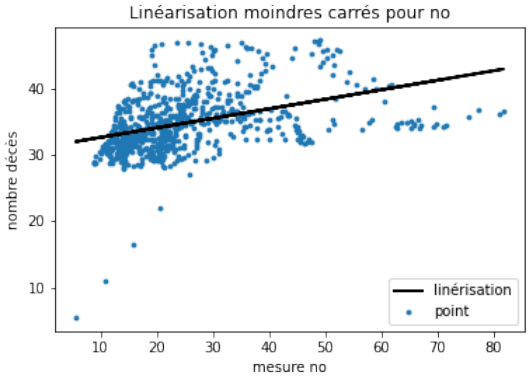
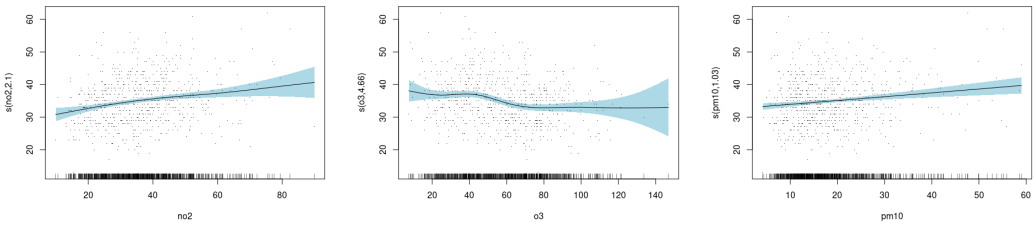


FIG. 4 – *Corrélation entre NO et mortalité : droite approchant le nuage de points par la méthode des moindres carrés*



(a) NO_2 (b) O_3 (c) PM_{10}

FIG. 5 – *Modèle GAM pour la mortalité quotidienne et chacun des polluants étudiés*

4.2 Données quotidiennes

On détermine la corrélation de Spearman entre la mortalité et chacun des polluants étudiés sur la période 2018-2019. Les résultats obtenus ne sont pas concluants : ($\rho_i^2 < 0.05$, négligeable) (*cf.* table 2).

En utilisant les méthodes décrites précédemment (sélection par seuil et moyenne glissante), on démontre une corrélation entre certains polluants et la mortalité quotidienne (*cf.* Table 3).

En effet, on obtient un $\rho^2 \in [0.2, 0.28]$ pour le NO, O_3, $PM_{2.5}$, PM_{10}, avec des paramètres particuliers. Cette relation permet notamment, via la méthode des moindres carrés, d'approcher la droite d'équation $y = ax + b$ déterminée par le nuage de points : $a = 0.143$, $b = 31.220$ (*cf.* figure 4).

Le développement des modèles additifs généralisés pour les données quotidiennes montre une faible liaison linéaire positive entre la mortalité et les polluants $PM_{2.5}$ et $PM10$ ($edf_{PM_{2.5}} \approx 1.001$ et $edf_{PM_{10}} \approx 1.027$), une liaison quadratique pour les polluants NO et NO_2 ($edf_{NO} \approx 2.779$ et $edf_{NO_2} \approx 2.101$) et une absence de liaison pour le polluant O_3 (approchant des fonctions constantes) (*cf.* table 5).

$$MA_{PM_{2.5}} = (RR_{PM_{2.5}} - 1) \cdot 100 = (e^{0.004589} - 1) \cdot 100 \approx 0.46\%$$

Impact de la pollution de l'air sur la mortalité

$$MA_{PM_{10}} = (RR_{PM_{10}} - 1) \cdot 100 = (e^{0.0030908} - 1) \cdot 100 \approx 0.31\%$$

5 Discussion

Les résultats explicités dans la section précédente montrent une corrélation entre pollution extérieure et mortalité. On retrouve des coefficients de corrélation de Spearman significatifs pour l'analyse des données mensuelles comme quotidiennes (en particulier pour les polluants NO et NO_2).

De plus, les modèles développés montrent une hausse de la mortalité due à l'augmentation de la pollution. D'après l'analyse des données mensuelles, une augmentation d'un pourcent de la pollution en NO_2 conduit à une hausse de la mortalité de 0.87%.

Ces résultats sont cohérents avec la littérature disponible (dont l'étude décrite dans la section 2). Cependant, les corrélations sont ici moins puissantes. En effet, l'étude démontrant elle aussi le lien entre les deux phénomènes étudiés Campagna et al. (2003) formule une conclusion plus forte, puisqu'elle aboutit à des pourcentages plus importants. Plusieurs facteurs peuvent l'expliquer, tels que l'exclusion des décès accidentels et intérêt particulier porté à la mortalité pour causes respiratoires ou cardio-vasculaires et la prise en compte d'autres facteurs environnementaux ainsi que les variations à court et à long terme. Il aurait été également intéressant de pouvoir effectuer un travail équivalent pour les polluants CO et SO_2 considérés parmi les polluants les plus dangereux sur la santé d'après l'OMS. Aussi, la distinction des patients par âge, sexe, pathologie et antécédents (informations qui ne sont pas fournies par l'INSEE et soumises au secret médical) pourraient produire des résultats plus pertinents. Enfin, l'hypothèse non réaliste d'uniformité des durées d'exposition aux polluants étudiés pourrait constituer un biais important.

6 Conclusion

Les résultats obtenus démontrent une corrélation entre les niveaux de pollution atmosphérique et la mortalité dans le département du Rhône entre 2007 et 2019 pour les polluants NO, NO_2, $PM_{2.5}$, et PM_{10}. Les méthodes d'analyse employées permettent de montrer qu'une augmentation de la mesure de certains polluants entraînait une hausse de la mortalité à court terme.
Cependant, ces conclusions ne sont pas suffisamment puissantes pour la conception d'un modèle permettant la prédiction de la mortalité. Une étude de la mortalité quotidienne sur une période plus importante, en disposant d'informations privilégiées sur les patients, de sorte à les distinguer par âge, sexe, durée d'exposition moyenne et antécédents médicaux pourraient néanmoins le permettre. Par la suite, on pourra également étudier le développement de modèles prédictifs pour tenter de prédire la mortalité et les hospitalisations en se basant sur les mesures de qualité de l'air.

Références

Api atmo auvergne-rhône-alpes.

Campagna, D., A. Lefranc, C. Nunes-Odasso, et R. Ferry (2003). Évaluation des risques de la pollution urbaine sur la santé en île-de-france (erpurs) : liens avec la mortalité 1987-1998. *VertigO-la revue électronique en sciences de l'environnement 4*(1).

Hamra, G. B., N. Guha, A. Cohen, F. Laden, O. Raaschou-Nielsen, J. M. Samet, P. Vineis, F. Forastiere, P. Saldiva, T. Yorifuji, et al. (2014). Outdoor particulate matter exposure and lung cancer : a systematic review and meta-analysis. *Environmental health perspectives*.

Hastie, T. et R. Tibshirani (1987). Generalized additive models : some applications. *Journal of the American Statistical Association 82*(398), 371–386.

INSEE (2020a). Base de données, mortalité.

INSEE (2020b). Recensement de la population.

Moolgavkar, S. H. (2000). Air pollution and hospital admissions for chronic obstructive pulmonary disease in three metropolitan areas in the united states. *Inhalation Toxicology 12*, 75–90.

Nelder, J. A. et R. W. Wedderburn (1972). Generalized linear models. *Journal of the Royal Statistical Society : Series A (General) 135*(3), 370–384.

OMS (2014). 7 millions de décès prématurés sont liés à la pollution de l'air chaque année.

Valavanidis, A., V. Thomais, et F. Konstantinos (2016). Air pollution as a significant cause of diseases and premature death. *Ambient Air Pollution in Urban Areas and Indoor Air Pollution are Associated with Adverse Health Effects and Premature Mortality*.

Valavanidis, A., T. Vlachogianni, et K. Fiotakis (2014). Airborne particulate matter in urban areas and risk for cardiopulmonary mortality and lung cancer : Dietary antioxidants and supplementation for prevention of adverse health effects. *Pharmakeftiki 26*(4), 139–156.

Summary

The WHO estimates that air pollution contributes to 7 million premature deaths per year. Our research aims to evaluate its effects on mortality in the Rhone department (France). Available mortality data between 2007 and 2019 were analyzed. By measuring the exposure to pollution of the population, focusing on the pollutants considered to be the most dangerous for humans according to the WHO (NO, NO_2, O_3, $PM_{2.5}$ and PM_{10}), and using linear and non-linear analysis methods, a significant correlation is demonstrated. The link between the two phenomena follows a positive linear trend (particularly for the PM and NO_2). We measured a $\rho^2 = 0.44$ for the NO_2, meaning that 44 % of the mortality is linked to this pollutant. We have also shown that an increase of 1% of this pollutant induces a 0.87% increase of mortality.

Visualisation spatio-temporelle de données de mobilité touristique extérieures

Maxime Masson*, Cécile Cayèré**, Marie-Noëlle Bessagnet*, Christian Sallaberry*, Philippe Roose*, Cyril Faucher**

*LIUPPA, E2S, Université de Pau et des Pays de l'Adour
https://liuppa.univ-pau.fr
**L3i, La Rochelle Université
https://l3i.univ-larochelle.fr

Résumé. Dans cet article, nous présentons la conception et le développement de composants de visualisation de données de mobilité intégrés dans un pipeline de traitement générique et modulaire. Ce travail a lieu dans le cadre du projet DA3T interdisciplinaire (à la fois géographie et informatique) de la région *Nouvelle-Aquitaine* visant à améliorer la valorisation et la gestion des villes touristiques. Nous discutons des travaux connexes sur les modes de visualisation spatiaux, temporels et spatio-temporels et faisons ensuite plusieurs propositions pour améliorer ces modes afin de répondre aux besoins d'analyse spécifique des villes touristiques. Cela inclut l'implémentation d'un cube de visualisation spatio-temporelle hautement personnalisable mais aussi la conception de cartes de chaleur multi-modes. L'expérience menée sur un jeu de données touristiques (données visiteurs de la ville de La Rochelle) pour prouver la pertinence de ces modes de visualisation est ensuite exposée.

1 Introduction

Grâce à l'évolution récente et rapide des technologies de capture mobile et des réseaux sociaux, les traces de mobilité humaine laissées volontairement ou involontairement sur le web sont de plus en plus nombreuses. Le tourisme est un domaine dans lequel la compréhension des traces de mobilité est importante pour améliorer la valorisation et le développement du territoire touristique. De nombreux travaux s'intéressent à l'analyse et à la compréhension de ces données (Girardin et al. (2009)). Cependant, peu d'outils et méthodes d'analyse génériques existent alors qu'ils présentent un grand intérêt pour les décideurs. A cette fin, un cadre d'analyse des traces numériques de mobilité a été proposé au sein d'un projet régional intitulé DA3T. DA3T (pour *Dispositif d'Analyse des Traces Numériques pour la Valorisation des Territoires Touristiques*) est un projet visant à améliorer la gestion, le développement et la valorisation des villes touristiques de la région *Nouvelle-Aquitaine* par l'analyse fine des pratiques touristiques dans ses villes, d'abord à l'échelle individuelle puis à l'échelle agrégée. Ce projet interdisciplinaire intègre à la fois des géographes et des informaticiens. Nous collectons des données GPS périodiquement via le téléphone de touristes volontaires grâce à l'application

mobile *GeoLuciole* développée dans le cadre du projet. Les données collectées sont appelées **traces de mobilité** et fournissent un aperçu des lieux visités par ces touristes au cours de leur séjour. Nous observons le déplacement des touristes de manière discrète via une série de positions géolocalisées et horodatées. Chaque position d'une trace est représentée par un tuple $p = (o, x, y, t, D)$ avec p la capture de l'objet mobile o (le visiteur), x et y les coordonnées spatiales (latitude et longitude), t l'horodatage de la capture, et D un ensemble de métadonnées supplémentaires collectées telles que la précision de la capture, la vitesse, etc (Cayèré et al. (2021)). Le touriste peut choisir la durée de l'acquisition des données et l'arrêter à tout moment. Ces traces sont discrètes en raison des limitations techniques (p. ex. le traitement, la capture, la sauvegarde de la batterie, etc.) et varient en précision selon le type de dispositif de capture. Elles sont purement quantitatives et au premier lancement de l'application, le visiteur est invité à fournir des informations générales (p. ex. la date du séjour, la connaissance ou non de la ville, le nombre de personnes avec lesquelles il voyage, etc.) En outre, des données publiques issues de plateformes Open Data peuvent être utilisées pour enrichir et compléter les traces (ex : points d'intérêt, restaurants, météo, etc.).

Afin d'analyser ces données collectées, une partie importante du projet DA3T consiste à concevoir une approche et des méthodes pour faciliter la réalisation d'applications d'exploitation de traces numériques nativement spatio-temporelles. A ce titre, une plateforme permettant de construire des chaînes de traitement pour l'analyse et l'exploitation d'informations provenant de sources complexes, hétérogènes et de granularité différente (ex : application mobile, réseaux sociaux, APIs web, crowdsourcing auprès du grand public, etc.) a été réalisée (Cayèré et al. (2021)). Elle est totalement modulaire et générique et permet aux utilisateurs de concevoir des flux de traitement personnalisés à partir de modules. Ces modules vont de l'extraction (modules de lecture ou d'importation de données), à la transformation (modules de nettoyage et d'analyse des données, par exemple par enrichissement sémantique) et enfin au chargement de ces données (modules de visualisation des données, par exemple via une carte, des diagrammes, des graphiques, etc.).

Cet article vise à présenter un module de visualisation des traces pouvant être appelé par la plateforme de chaîne de traitement. Ce module doit répondre aux exigences des géographes et du contexte du projet DA3T en matière de visualisation des données. Dans la section 2, nous présenterons les exigences du projet DA3T en matière de visualisation. Ensuite, nous présenterons les différentes méthodes de visualisation existantes sur lesquelles nous nous sommes basés et dont nous nous sommes inspirés pour construire nos deux contributions (c.f. section 3). Nous introduirons nos contributions (c.f. section 4) et présenterons les différents tests et expériences que nous avons réalisés sur un jeu de données touristiques afin de démontrer la pertinence des modules de visualisation que nous proposons (c.f. section 5). Enfin, nous conclurons en abordant les perspectives futures dans le cadre du projet DA3T.

2 Exigences du projet DA3T pour la visualisation de données de mobilité

Nous nous concentrerons sur les modules de visualisation dans un contexte strictement touristique. En collaboration avec les géographes du projet, nous avons identifié deux besoins principaux en matière de visualisation.

Le premier est la **visualisation des données de mobilité spatio-temporelle**, différents critères pour la conception du mode de visualisation ont été décidés :

- *Adaptabilité spatiale* : Les positions (captures GPS) sont capturées à intervalles réguliers par le dispositif de capture (p. ex. le smartphone) du touriste. Par conséquent, les traces individuelles peuvent varier fortement en taille en fonction de la durée de l'activité touristique et peuvent avoir un grain spatial totalement différent : à l'échelle d'un quartier, d'une ville ou même d'une région si la personne s'est beaucoup déplacée durant son séjour. Il est donc nécessaire de proposer un mode de visualisation qui soit **spatialement adaptatif**, et qui puisse montrer de manière pertinente un parcours court, moyen, ou long.
- *Performance* : Le nombre de touristes participant au projet DA3T est assez important, ils restent généralement plusieurs jours ce qui multiplie le nombre de captures GPS à analyser. Actuellement, la base de données des traces comprend déjà plus de 100 000 positions capturées et leurs méta-données. Il est donc nécessaire de pouvoir afficher toutes les traces de manière **performante**. Par performant, nous entendons que le mode de visualisation doit être utilisable sans ralentissement majeur avec une quantité de points affichés allant jusqu'à 110 000 au maximum. Il doit également être rapide, c'est-à-dire que son temps de chargement initial (initialisation) ne doit pas dépasser un temps jugé acceptable par l'utilisateur : nous l'avons fixé à 15 secondes après discussion avec les utilisateurs finaux.
- *Accessibilité* : Le logiciel doit être facile à utiliser et à lire par des non-informaticiens. Un utilisateur non spécialiste souhaite obtenir une représentation des données spatiales et temporelles sur un même support afin de faciliter l'analyse.
- *Échelle temporelle multi-modes* : L'analyse des données doit se faire à différentes échelles temporelles : 24 heures, semaine, mois ou même saison. Par exemple, nous voulons explorer le comportement des touristes sur 24 heures, tout au long de l'année.
- *Extensibilité* : Il devra également être **extensible**. Nous définissons ce terme de deux manières : (a) la capacité à s'adapter à d'autres ensembles de données et à d'autres cas d'utilisation en dehors du domaine du tourisme ; (b) la capacité d'être étendu avec de nouvelles fonctionnalités.
- *Portabilité* : Il doit être **portable**, donc simple à transférer entre différents utilisateurs mais surtout fonctionnel sans utilisation d'un logiciel particulier installé sur la machine cible. Nous excluons l'utilisation de systèmes d'information géographique lourds qui nécessitent une installation préalable.

Le deuxième besoin de visualisation consiste à **mettre en évidence des *points chauds* multi-dimensionnels dans des villes touristiques**. Un point chaud est une zone géographique, une période temporelle (ou les deux) d'activité intense. Dans notre cas, nous mesurons l'activité des touristes. Ces points chauds peuvent être de différents types :

- *Points chauds spatiaux* : Ils peuvent être strictement spatiaux. Il s'agit donc de zones géographiques d'une ville où la concentration de traces de mobilité est élevée. La connaissance de ces zones permet aux décideurs de savoir quels sont les quartiers ou les points d'intérêt les plus populaires auprès des touristes et donc de prévoir éventuellement des aménagements supplémentaires.
- *Points chauds temporels* : Les points chauds peuvent également être strictement temporels : il s'agit de périodes de la journée (ou de la semaine, du mois, de l'année, etc.)

où les touristes sont les plus actifs dans leur pratique.

— *Points chauds spatio-temporels* : Enfin, en mixant les deux critères, nous obtenons les points chauds spatio-temporels, c'est-à-dire les zones de la ville où les touristes sont présents au même moment de la journée. Par exemple, une zone où les touristes sont nombreux mais répartis sur la journée n'est pas considérée comme un point chaud spatio-temporel : la zone est massivement visitée mais pas au même moment de la journée.

3 Travaux connexes

Présentons des travaux connexes sur les modes de visualisation qui peuvent être pertinents pour les préoccupations et les défis du projet DA3T. Nous avons divisé les modes de visualisation en trois catégories, (1) spatial et thématique, (2) temporel et thématique et (3) spatio-temporel. Nous appelons visualisation thématique toutes visualisations incluant des données d'enrichissement en plus des données de mobilité.

3.1 Visualisations spatiales et thématiques

Les modes de visualisation spatiaux les plus évidents sont les visualisations de type carte. Il existe de nombreux types de cartes et cette catégorie s'étend bien au-delà des cartes "classiques". Explorons les différents types de cartes.

Les cartes choroplèthes peuvent être utilisées pour visualiser un ensemble de données discrètes simples (p. ex. des points sur un planisphère) mais aussi des données complexes agrégées en statistiques. Il s'agit d'un modèle de carte thématique où certaines zones sont colorées en fonction des valeurs statistiques des données relatives à cette zone.

Les cartes hexagonales divisent l'espace en une grille d'hexagones qui sont ensuite colorés de la même manière que la carte choroplèthe, en tenant compte de la valeur statistique des données pour cette zone.

Les cartes de chaleur spatiales montrent l'ampleur d'un phénomène spécifique à travers la variation de l'intensité ou de la teinte d'un gradient de couleur en 2 dimensions. L'espace géographique est divisé en une matrice (p. ex. matrice de pixels) de dimensions x et y et chaque pixel est associé à la couleur appropriée en fonction des données à visualiser (Liu et al. (2011)).

Les cartes à distribution des points sont une alternative aux cartes de chaleur. Ces cartes montrent la distribution géographique d'un phénomène discret à l'aide de symboles, le plus souvent des points. Ainsi, on peut comprendre la distribution globale d'un phénomène et comparer cette distribution en fonction des différentes régions d'étude (Roth (2010)).

Les cartes à bulles sont assez similaires aux cartes en grappes de points, sauf que chaque bulle est associée à une zone géographique choisie et que deux variables peuvent être représentées à la fois par la taille de la bulle et sa couleur.

Enfin, les **cartogrammes** ou *cartes anamorphiques* sont des représentations combinant des informations statistiques et géographiques où la géométrie des zones est déformée en fonction d'une valeur statistique (Nusrat et Kobourov (2016)). Plusieurs types existent, basés sur la

surface (*ex : taille des pays en fonction de leur population*) ou la distance (*ex : représentation des temps de trajet*).

3.2 Visualisations temporelles et thématiques

L'actogramme montre différents individus étudiés sur une période de temps (p. ex. une journée). Ces individus sont disposés en ligne sur le côté gauche de la représentation. L'objectif est de montrer le type d'activité réalisée en fonction du temps. Chaque individu est associé à une barre montrant l'ordre chronologique de ses activités (aussi appelé modalité) pendant la période d'étude. La couleur de chaque barre évolue en fonction de l'activité en cours de réalisation.

Le chronogramme est un diagramme à barres empilées avec le temps sur l'axe des x et la proportion de personnes par catégorie d'activité sur l'axe des y. Le chronogramme représente la distribution des individus, répartis selon plusieurs modalités, à chaque pas de temps de la période d'étude (Pistre et al. (2015)). Il permet de trouver des tendances dans les activités des individus pendant la période d'étude (Menin et al. (2020)).

La visualisation radiale ou "roue de mobilité" est un diagramme circulaire textuel simulant une horloge de 24 heures et contenant 2 anneaux de 24 rectangles, un pour chaque heure. Le cercle extérieur indique le rapport entre le nombre d'individus en mouvement et la population de la zone pour une période donnée. Le cercle intérieur représente la distribution des individus mobiles sur la période de temps donnée selon le type de mobilité. Au centre se trouve la distribution temporelle des modes de transport (Menin et al. (2020)).

3.3 Visualisations spatio-temporelles

Le **cube spatio-temporel** est un mode de visualisation tridimensionnel visant à montrer les comportements et les interactions à travers l'espace et le temps. Il est basé sur un objet cube dans un espace euclidien (Bach et al. (2017)). Les dimensions largeur et longueur (x et y) représentent les coordonnées spatiales (c'est-à-dire la latitude (y) et la longitude (x)) et la dimension hauteur (z) : le temps. Ce mode de visualisation a l'avantage d'être interactif en permettant à l'utilisateur de changer l'échelle temporelle (Kraak (2003a)). Les trajectoires (aussi appelées *trajets espace-temps*) sont affichées dans le cube (Kraak (2003b)). De nombreuses interactions supplémentaires peuvent être mises en œuvre à l'aide de ce type de représentation : on peut imaginer la possibilité d'extraire des intervalles temporels ou des zones spatiales (Bach et al. (2017)).

Les cartes animées sont un autre moyen, cette fois en 2D, de visualiser des ensembles de données spatio-temporelles. L'utilisateur a la possibilité de démarrer et d'arrêter une animation appliquée à une séquence d'éléments cartographiques mais aussi de contrôler sa vitesse d'exécution. Une autre technique consiste à afficher très rapidement plusieurs cartes représentant un phénomène à différents moments afin de montrer l'évolution et la dynamique d'un phénomène (Kaddouri et al. (2014)). L'objet cartographique est alors synchronisé avec le temps, qui est souvent représenté sous la forme d'une barre de temps afin de permettre à l'utilisateur d'identifier le moment affiché.

Pour la **visualisation des données de mobilité spatio-temporelle**, le cube spatio-temporel et la carte animée peuvent être utilisés à cette fin (c.f. section 3.3). Mais le cube a l'avantage

de faciliter la mise en œuvre d'une mise à l'échelle temporelle multi-modes et de permettre aux utilisateurs finaux de visualiser instantanément la dimension temporelle des données sans avoir à attendre la fin d'une animation. C'est un moyen robuste et éprouvé d'afficher des données ayant à la fois une dimension spatiale et temporelle. Le principal inconvénient du cube spatio-temporel dans les implémentations existantes est qu'ils sont souvent intégrés dans des logiciels SIG lourds et, en tant que tels, ne conviennent pas pour être intégrés dans un pipeline de traitement comme le nôtre. De plus, ces implémentations manquent généralement d'options de personnalisation avancées, sont dépendantes de logiciels externes et ne sont pas facilement extensibles. En termes de **mise en évidence des points chauds spatiaux**, nous pouvons constater que la plupart des cartes proposées (c.f. section 3.1) sont appropriées. Cependant, on peut distinguer deux catégories différentes. Dans la **première catégorie** (comprenant, entre autres, les cartes choroplèthes, les cartogrammes, etc.) : les points chauds spatiaux doivent être associés à des zones géographiques précises avant d'être calculés. Le choix de la granularité des zones est donc très important (quartiers, communes, etc.). La **deuxième catégorie** (cartes de chaleur spatiales, cartes à distribution de points) permet en revanche de visualiser les points chauds spatiaux de manière totalement dynamique. Ainsi, il n'est pas nécessaire de définir au préalable des zones où les données seraient agrégées. La première catégorie est cependant utile pour afficher des aspects thématiques par zone géographique (p. ex. le nombre de visites par quartier).

4 Contributions

Nos contributions sont divisées en 2 sections. Premièrement, notre base de travail est une **nouvelle implémentation** personnalisable et efficace du concept de **cube spatio-temporel** (c.f. section 4.1) qui s'intégre sous forme de module à la plateforme de chaîne de traitement DA3T. De plus, nous intégrons des cartes de chaleur multi-modes à ce cube pour l'identification de points chauds (c.f. section 4.2).

4.1 Visualisation des données de mobilité spatio-temporelle : *une nouvelle implémentation du cube spatio-temporel*

Nous proposons un module de visualisation de traces GPS horodatées utilisant une représentation en cube spatio-temporel, pertinent pour des données à la fois spatiale et temporelle.

4.1.1 Objectif

Pour rappel, notre besoin est un mode de visualisation supportant les données de mobilité à la fois spatiale et temporelle, facile à manipuler pour les non-spécialistes et adaptatif (c.f. section 2). Les cartes 2D simples mettent exclusivement en évidence l'aspect spatial des données et les représentations chronologiques négligent souvent l'aspect spatial des données offrant donc une visualisation très limitée de celles-ci. Le cube spatio-temporel remédie à ces problèmes en matérialisant ces deux dimensions de la même manière (décalage sur un axe) et permet de les visualiser simultanément. De plus, en raison de la nature de la composante spatiale intrinsèquement caractérisée par deux valeurs ($latitude$ et $longitude$) contre une seule pour le temps ($timestamp$), le côté spatial conserve une importance légèrement supérieure au

sein du support de visualisation. Ce problème est mineur pour notre étude de cas puisque la géographie des points doit primer sur l'aspect purement temporel.

Comme mentionné précédemment, la notion de cube spatio-temporel existe dans la littérature Kraak (2003b). Au-delà du modèle conceptuel, il existe cependant un nombre réduit d'implémentations, et les existantes sont souvent intégrées sous forme de plugins dans des logiciels SIG (*Système d'Information Géographique*). Elles ne sont donc pas utilisables dans une chaîne de traitement automatisée comme celle du projet DA3T, ni portables. De plus, ces implémentations sont souvent limitées en termes de possibilités (faible personnalisation, granularité temporelle non modifiable, intégration avec des logiciels SIG lourds, etc). Or, dans le cadre du projet DA3T, les géographes souhaitent que des outils de visualisation supplémentaires soient intégrés au cube et puissent être activés à volonté en fonction de leurs besoins d'analyse. Pour répondre précisément à ces besoins, nous avons basé notre module de visualisation sur le concept du cube spatio-temporel et l'avons étendu pour proposer une implémentation adaptative et puissante fonctionnant sur de grands ensembles de données et offrant de nombreuses possibilités de personnalisation pour les utilisateurs finaux non spécialistes (c.f. figure 1).

4.1.2 Implémentation

Le cube a trois axes : x (latitude), y (longitude) et z (temps) (c.f. figure 1, 1). À la base du cube (c.f. figure 1, 2), la carte de la zone géographique concernée est affichée et superposée de sorte qu'en vue de dessus, les points coïncident avec leur position réelle sur la carte. Cette carte adapte son échelle spatiale à la couverture géographique des points à visualiser : quartier, ville, département, pays, etc. Quatre vues sont proposées par défaut dans le cube : la vue libre où l'utilisateur peut zoomer, tourner et faire un panoramique librement, la vue de haut en bas qui affiche strictement les dimensions spatiales (latitude/longitude), et enfin deux vues horizontales (latitude/temps, et longitude/temps). Elles sont accessibles via des boutons d'icône (c.f. figure 1, 4).

Concernant l'axe temporel, plusieurs échelles sont disponibles (c.f. figure 1, 5) pour visualiser les traces : l'échelle *Complète* est définie par l'intervalle temporel englobant des traces visualisées (p. ex. deux traces se passant à une semaine d'intervalle sont visualisées sur un axe représentant cette semaine entière), l'échelle *1 an* représente les données sur un axe d'une année (l'année précise de la capture d'une position est ignorée dans ce mode), l'échelle *1 mois* représente les données sur un axe d'un mois, l'échelle *1 semaine* représente les données sur un axe d'une semaine et l'échelle *24h* représente les données sur un axe d'un jour divisé en 24 heures. Chacune de ces échelles permet d'avoir un degré d'analyse différent. Le choix de l'échelle se fait en fonction de la granularité temporelle des données sources. En fonction de l'échelle sélectionnée, le code couleur de la légende à gauche (c.f. figure 1, 6) change. Dans le cas où l'échelle *1 an* est sélectionnée, il y aura une couleur différente pour chaque trajectoire représentant le déplacement d'un visiteur durant une année particulière (p. ex. *visiteur 1 - année 1*, *visiteur 1 - année 2*, *visiteur 2 - année 1*, etc.) .

Les points insérés dans le cube peuvent avoir des propriétés supplémentaires sous forme de clés-valeurs. Il s'agit d'informations sur les points (p. ex. l'altitude, la précision GPS, le type de dispositif de capture, etc.). En survolant un point du cube avec la souris, les propriétés de ce point s'affichent en haut à gauche de l'écran (c.f. figure 1, 3). Parmi ces propriétés, l'une est

l'identifiant du visiteur. Avec lui et la date, le cube attribue une couleur pour chaque touriste et pour chaque jour, mois, année, etc. (selon le type de segmentation choisi, voir ci-dessus). Par exemple, pour une segmentation en 24h, toutes les traces de l'utilisateur x à la date y auront la même coloration. Cette couleur est affichée à deux endroits : d'abord directement sur les points dans l'espace tridimensionnel du cube (c.f. figure 1, 1), mais aussi dans la légende à gauche (c.f. figure 1, 6) où l'utilisateur peut observer à quel jour et à quel touriste correspond chaque couleur.

A partir de cette même légende, l'utilisateur peut décider de cacher ou d'afficher une trace, cette fonctionnalité est particulièrement utile dans le cas d'un très grand jeu de données affiché dans le cube afin de visualiser plus facilement une ou plusieurs traces. Les mouvements sur ces traces sont animés en temps réel dans le cube (c.f. figure 1, 1), permettant à l'utilisateur de voir rapidement à quel moment de la journée les touristes se croisent. Cette animation peut être mise en pause et sa vitesse peut être réglée via le menu latéral de droite (c.f. figure 1, 7). D'un point de vue purement visuel, un thème clair et un thème foncé sont proposés, ainsi que la possibilité de relier les points entre eux par des lignes (c.f. figure 1, 7). Cela peut être utile pour visualiser de grandes quantités de données dans des environnements géographiquement complexes. Enfin, un menu est dédié à l'édition des gradients de couleur utilisés par les différentes cartes de chaleur qui peuvent être activées dans le cube (c.f. figure 1, 7), chacune d'entre elles étant dédiée à un type spécifique d'analyse des pratiques touristiques. Nous aborderons ce sujet dans les prochaines sections (c.f. section 4.2).

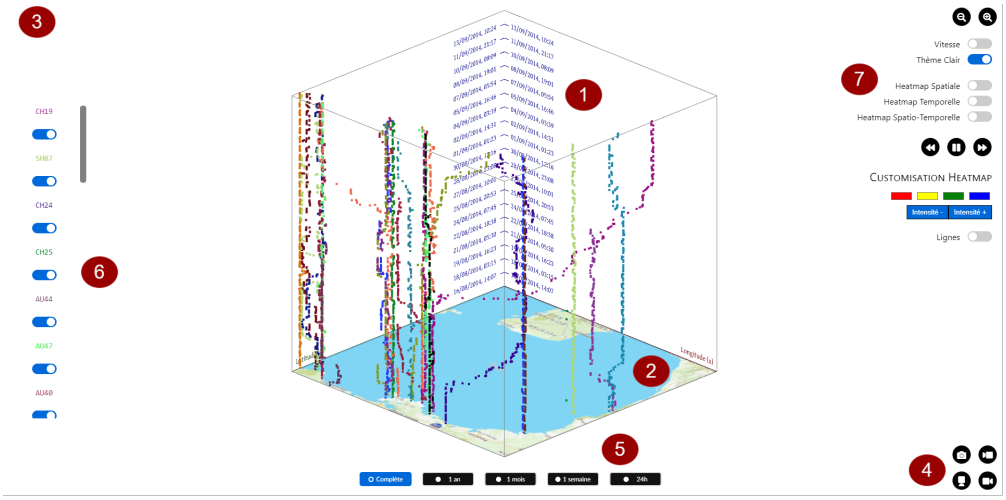


FIG. 1 – *Notre implémentation du cube spatio-temporel*

4.1.3 Développement

Le développement de ce prototype de cube a été réalisé en $JavaScript$ en utilisant le moteur de rendu $WebGL$. Nous avons utilisé la bibliothèque $Three.JS$ pour la création de la scène 3D. Un soin particulier a été apporté à l'optimisation de la scène afin que la visualisation de milliers de points ne pose pas de problème particulier. Un contrôleur web $Python$ est chargé

de recevoir les requêtes pour ce module, de générer le fichier HTML avec les bons paramètres (liste de points à visualiser, méta-données, etc.) puis de le renvoyer à l'utilisateur. L'utilisateur peut alors l'ouvrir dans son navigateur web ou l'envoyer à une autre personne très facilement : pas besoin de logiciel SIG, ni d'installation de bibliothèque logicielle.

Cette implémentation efficace et personnalisable du concept de cube spatio-temporel permet donc de visualiser finement une ou plusieurs trajectoires, les rendant ainsi plus compréhensibles et analysables par les géographes. Cependant, des fonctionnalités de visualisation plus avancées ont également été intégrées : les cartes de chaleur que nous décrivons ci-après.

4.2 Mise en évidence de points chauds : *cartes de chaleur multi-modes*

Comme mentionné précédemment, le cube est une base pour des outils de visualisation plus avancés. Le premier outil que nous proposons est un ensemble de cartes de chaleur. Chacune d'entre elles a un objectif distinct. La carte de chaleur spatiale pour l'identification des points chauds strictement spatiaux (i), la carte de chaleur temporelle pour l'identification des points chauds strictement temporels (ii) et la carte de chaleur spatio-temporelle pour l'identification des points chauds spatiaux et temporel (iii).

4.2.1 Objectif

L'un des principaux défis du projet DA3T est l'identification des points chauds touristiques. Ceux-ci sont de différents types : (i) les zones géographiques où les touristes sont les plus actifs et qui ont donc un plus grand nombre de captures GPS associées (points chauds spatiaux) ; (ii) les points chauds temporels que nous définissons comme une période (intervalle de temps) où les touristes sont les plus actifs (la quantité de points GPS relevés est la plus importante) ; (iii) les points chauds à la fois spatiaux et temporels, i.e. des zones géographiques où les sujets de l'étude (p. ex. les touristes) se trouvent au même endroit au même moment.

Les cartes de chaleur sont des outils de visualisation qui regroupent des données. Elles associent ensuite ces valeurs statistiques à un dégradé de couleurs pour les présenter visuellement. Dans le domaine du traitement des données géographiques, les cartes de chaleur sont particulièrement utiles pour mettre en évidence les zones géographiques riches en éléments présentant une caractéristique spécifique (p. ex. les zones à forte activité touristique, également appelées *points chauds*). Pour les points chauds spatiaux, nous avons utilisé une carte de chaleur classique en deux dimensions agrégeant les latitudes et les longitudes, qui, lorsqu'elles sont superposées à un plan de ville, permettent d'identifier les zones les plus fréquentées en un clin d'œil. Nous avons basé notre développement sur des algorithmes existants. L'implémentation de cette carte de chaleur a été réalisée à l'aide d'une bibliothèque $JavaScript$ existante (*heatmap.js*) et n'est donc pas abordée plus dans cet article.

La connaissance de la période de la journée où les touristes sont les plus actifs est particulièrement importante pour les décideurs des villes touristiques, et le concept de cube spatio-temporel tel que défini dans la littérature ne propose pas de mode de visualisation répondant à cette problématique. Les captures GPS des traces sont visibles en fonction de l'heure de la journée, mais aucun système ne permet de détecter et de mettre en évidence automatiquement les périodes d'activité intense afin d'obtenir une visualisation ergonomique et présentable aux décideurs. Nous proposons donc une carte de chaleur unidimensionnelle appliquée à l'axe du

temps. Nous nous sommes ensuite inspirés des cartes de chaleur bidimensionnelles (spatiales) et unidimensionnelles (temporelles) pour proposer une généralisation tridimensionnelle pour l'identification des points chauds spatio-temporels. Ici, l'espace n'est pas divisé en pixels ni en unités temporelles mais en sous-cubes d'unités fixes que l'on peut assimiler à des voxels, un voxel étant l'équivalent d'un pixel mais en 3 dimensions. L'ensemble des calculs se fait donc sur cette matrice de voxels.

4.2.2 Implémentation

Carte de chaleur temporelle Dans un premier temps, nous proposons une carte de chaleur purement temporelle prenant la forme d'un cylindre sur l'axe du temps. Cette carte de chaleur est mono-dimensionnelle (contrairement aux cartes de chaleur spatiales qui sont bidimensionnelles). Plus une période de temps est riche en captures GPS (points de trace), plus le gradient tendra vers le rouge. Moins il y a de captures pour la période, plus le gradient tendra vers le bleu. Dans l'exemple figure 3, nous l'appliquons à trois traces (deux à gauche et une à droite) sur une période de 24 heures, mais étant calculé dynamiquement, il est utilisable avec n'importe quelle période de temps. Le gradient de couleur et l'intensité de la carte de chaleur sont entièrement personnalisables.

Carte de chaleur spatio-temporelle Nous proposons ensuite la carte de chaleur spatio-temporelle (c.f. figure 2). Les zones pour lesquelles un nombre significatif de points de trace ont une proximité à la fois spatiale et temporelle sont colorées en rouge et sont au nombre de 3 sur l'exemple figure 2. De la même manière que la carte de chaleur temporelle, la carte de chaleur spatio-temporelle est également entièrement personnalisable avec les mêmes outils (choix du gradient et de l'intensité). Ce type de carte de chaleur permet de visualiser les zones d'un espace géographique où les touristes sont présents au même moment.

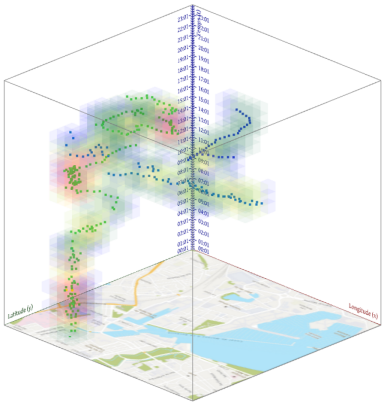


FIG. 2 – *La carte de chaleur spatio-temporelle en vue libre*

4.2.3 Développement

Dans le cas de la carte de chaleur temporelle, le moteur de génération va échantillonner l'axe temporel en un nombre fixe de sous-axes d'une unité de longueur calculée (p. ex. 5min, 1h, etc.), cette valeur dépendant de la plage de données à afficher. Pour chaque sous-axe, il comptera les points ayant une valeur temporelle incluse ou proche de celle-ci et attribuera ainsi une couleur pour chaque bande horizontale. Pour la carte de chaleur spatio-temporelle, le cube de dimension 100 x 100 x 100 est subdivisé en sous-cubes (pseudo voxel) de 10 unités par 10 unités pour un total de 1000 sous-cubes. Le choix de cette valeur de 10 unités a fait l'objet de plusieurs expériences, en effet les cubes ayant un trop petit volume entraînent des temps de

chargement et d'initialisation de la carte de chaleur très longs, tandis que les cubes trop grands sont synonymes de perte de précision. Pour chaque sous-cube, les points inclus ou proches sont comptés afin d'associer une couleur du gradient spécifié. Nous avons également utilisé la bibliothèque $JavaScript$ ($WebGL$) $Three.JS$ pour implémenter ces cartes de chaleur. Grâce à ces représentations en cartes de chaleur, l'identification directe des points chauds spatiaux, temporels et spatio-temporels a été facilitée. Les 3 modes de cartes de chaleur peuvent être activés et désactivés à volonté via un menu de contrôle.

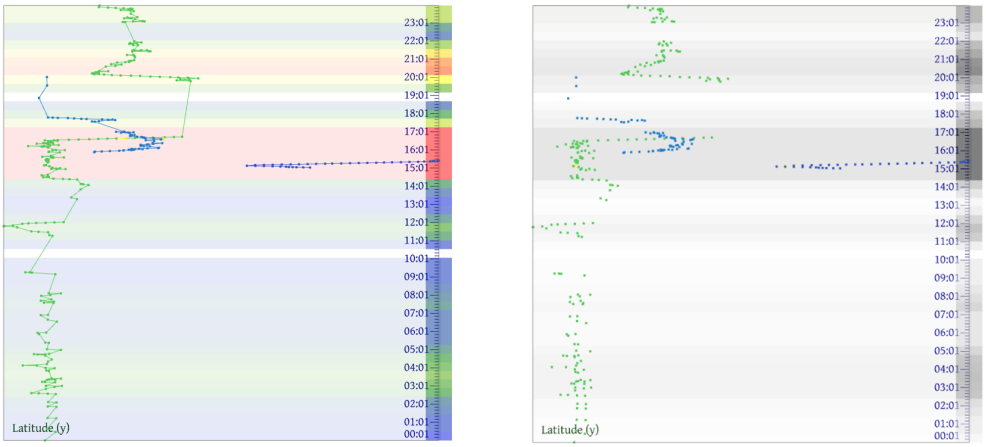


FIG. 3 – *La carte de chaleur temporelle appliquée à 3 traces sur une période de 24 heures (avec deux exemples de gradient de couleur) en vue horizontale (latitude/temps).*

5 Expérimentations

Nous avons mis en place plusieurs expériences avec divers jeux de données afin de valider les critères que nous avons définis dans la section 2 (*Exigences du projet*). L'une porte sur la visualisation spatio-temporelle de traces de migration d'oiseaux : pour valider les critères liés à la **visualisation des données de mobilité spatio-temporelle** (*adaptabilité spatiale, performance, accessibilité, échelle temporelle multi-modes, extensibilité et portabilité*). L'autre sur la visualisation des points chauds spatio-temporels appliquée aux données d'un réseau social : pour valider la mise en évidence des **points chauds spatio-temporels**. Pour plus de détails sur ces expériences, voir Cayèré et al. (2021). Dans cet article, nous présentons la visualisation des points chauds temporels appliquée à des données de mobilité touristique (afin de valider la mise en évidence des points chauds temporels).

5.1 Objectif

L'objectif de cette expérience est de montrer la capacité de la carte de chaleur temporelle à mettre en évidence les points chauds temporels, c'est-à-dire les moments de la journée où les

sujets de l'étude (c.-à-d. touristes) sont les plus actifs. Nous tentons de valider les principaux critères et exigences que nous avons établis pour la carte de chaleur temporelle (c.f. section 1).

5.2 Jeu de données

Nous utilisons les traces de mobilité des touristes collectées via l'application GeoLuciole dans la ville de La Rochelle (France) dans le cadre du projet DA3T. Ce jeu de données comprend environ 100 000 positions GPS capturées lors des déplacements de 82 touristes. Elles sont réparties sur un peu moins d'un an (2020), principalement en période estivale. L'intervalle moyen de capture est d'environ 5 minutes lorsque l'application est active.

Afin de valider nos résultats et en raison de leur nature principalement visuelle, nous avons dû utiliser un indicateur qualitatif. Nous avons donc collaboré avec les géographes du projet à La Rochelle et organisé un entretien avec eux afin d'obtenir leur avis. Pour chaque point chaud temporel identifié, nous leur avons demandé s'il était pertinent ou non et s'ils étaient capables de l'utiliser pour construire une analyse des comportements touristiques temporels dans la ville à différentes échelles.

5.3 Mise en place

En utilisant le jeu de données *GeoLuciole* (mobilité touristique à La Rochelle), nous tentons de répondre aux questions :

— En semaine, à quelle heure de la journée les touristes sont-ils les plus actifs ?

— Pendant le week-end, à quelle heure de la journée les touristes sont-ils les plus actifs ?

Par exemple, pour répondre à la deuxième question, nous mettons en œuvre la chaîne de traitement suivante grâce à la plateforme DA3T (c.f. figure 4) :

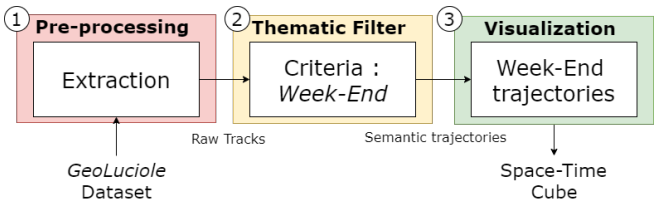


FIG. 4 – *Chaine de traitement appliqué au jeu de données GeoLuciole*

Tout d'abord, nous effectuons une extraction de données *GeoLuciole*, puis nous filtrons la trace pour n'avoir que les positions capturées le samedi et le dimanche. Enfin, nous envoyons ces données filtrées au cube spatio-temporel pour la visualisation.

5.4 Résultats

Nous obtenons les cartes de chaleur temporelles présentées ci-dessous (c.f. figure 5). Pour les traces des jours de la semaine avec une segmentation sur 24 heures, nous obtenons un total de 137 trajectoires (environ 85 000 positions). On remarque sur l'axe du temps que la période d'activité se situe entre 10h et 23h (couleur jaune/vert). Il existe également une période encore

plus intense entre 14h et 20h (couleur orange/rouge). C'est également durant cette période que les touristes sont les plus mobiles (la longitude varie beaucoup). En revanche, le matin et le soir, les touristes ne sont pas très actifs. Ils sont statiques et bougent très peu. Pour les traces du week-end avec une segmentation sur 24 heures, nous obtenons un total de 29 trajectoires (environ 15 000 positions) avec 22 trajectoires le samedi et 7 le dimanche. La principale période d'activité se situe entre 10h et 20h (couleur jaune/vert) avec une période d'activité plus intense entre 17h et 20h (couleur orange/rouge).

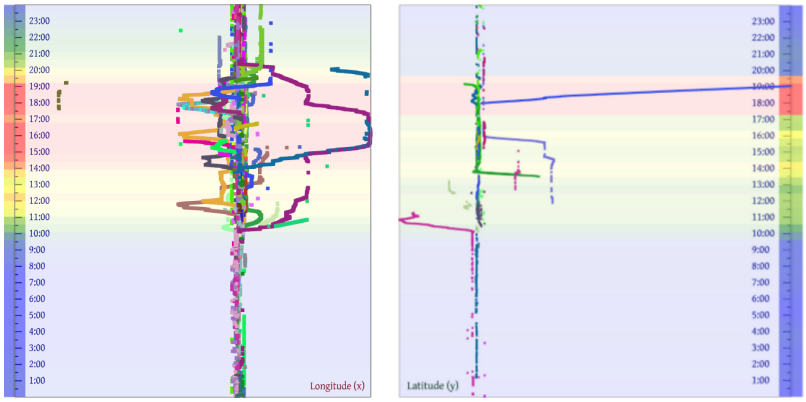


FIG. 5 – *La carte de chaleur temporelle appliquée aux traces des jours de semaine (à gauche) et aux traces du week-end avec segmentation en 24 heures (à droite)*

Nous avons mis en place un entretien qualitatif avec les géographes du projet, leur avons présenté les résultats, posé plusieurs questions et avons obtenu la confirmation de la pertinence de nos résultats. De plus, il semble logique que les touristes restent chez eux ou dans un restaurant le soir, ce qui explique le manque de mobilité pendant cette période, ainsi que la nuit lorsqu'ils dorment. La carte de chaleur temporelle nous a donc permis d'identifier très facilement ces points chauds via un affichage visuel.

Géographe 1 *: La carte de chaleur temporelle nous a permis d'identifier facilement les périodes de forte activité au cours d'une journée touristique typique à La Rochelle.*
Géographe 2 *: Avec cet outil, nous pouvons observer visuellement les points chauds sur un intervalle de temps spécifique très rapidement. Il est très pratique de pouvoir passer d'une vue spatiale à une vue temporelle presque instantanément.*

6 Conclusion

Dans cet article, nous avons proposé deux contributions distinctes. Elles s'inscrivent dans un projet régional visant la valorisation des villes touristiques par l'analyse fine des pratiques des visiteurs : **le projet DA3T**. Tout d'abord, afin de répondre aux besoins de visualisation liés aux traces nativement spatio-temporelles collectées auprès de dizaines de touristes volontaires via une application mobile, nous proposons une nouvelle implémentation du concept de cube spatio-temporel. Ce dernier est spatialement adaptatif et à une échelle temporelle multi-modes

afin de répondre à de nombreux besoins d'analyse sur des données de différentes granularités. De plus, nous avons conçu cette solution pour qu'elle soit accessible aux non-informaticiens, elle est donc à la fois facile à utiliser et portable, c'est-à-dire qu'aucune installation logicielle n'est nécessaire pour lancer la visualisation. Enfin, nous avons voulu que cette implémentation soit efficace et rapide sur de grands ensembles de données.

Le cube spatio-temporel est également extensible, il sert de base à des fonctionnalités de visualisation plus avancées. À ce titre, notre deuxième proposition consiste à lui ajouter des cartes de chaleur multi-modes. Celles-ci sont de 3 types. Tout d'abord, nous avons la carte de chaleur spatiale, qui est largement utilisée de nos jours. Elle permet d'identifier visuellement les points chauds spatiaux d'un territoire, c'est-à-dire les zones géographiques où les touristes sont les plus présents. Mais nous proposons également 2 modes de cartes de chaleur différents : **la carte de chaleur temporelle** pour l'identification des points chauds temporels, c'est-à-dire les moments de la journée (ou de toute autre période) où l'activité touristique est la plus intense. Enfin, nous proposons une carte de chaleur tridimensionnelle ou **cartes de chaleur spatio-temporelles** (*spatial et temporel*) pour visualiser les zones où les touristes se trouvent au même endroit au même moment donné ou à la même période temporelle. Nous avons ensuite validé ces 3 propositions en mettant en place des expérimentations.

Le projet DA3T prendra fin en 2022. D'ici là, il est prévu de continuer à améliorer l'implémentation de notre cube spatio-temporel en ajoutant de nouvelles fonctionnalités. Par exemple, la possibilité d'afficher des données d'enrichissement thématiques directement dans celui-ci : points d'intérêt (POI), quartiers, espaces verts. Nous allons également implémenter une superposition sur l'axe z pour afficher des données thématiques avec une dimension temporelle pour la période sélectionnée (p. ex. météo historique, lever et coucher du soleil, etc.) Pour les cartes de chaleur multi-modes, nous pensons les rendre encore plus personnalisables en permettant, par exemple, la personnalisation des dimensions des sous-cubes dans la carte de chaleur spatio-temporelle (p. ex. 1h/1km^2). Nous souhaitons également améliorer les performances de la carte de chaleur spatio-temporelle afin que sa résolution puisse être augmentée. Enfin, une expérimentation plus globale de la plateforme est en cours de réalisation ; elle intégrera des expérimentations sur chaque module développé dont le cube spatio-temporel. Le code de ces travaux sera mis en ligne à la fin du projet DA3T.

Références

Bach, B., P. Dragicevic, D. Archambault, C. Hurter, et S. Carpendale (2017). A descriptive framework for temporal data visualizations based on generalized space-time cubes. In *Computer Graphics Forum*, Volume 36, pp. 36–61. Wiley Online Library.

Cayèré, C., C. Sallaberry, C. Faucher, M.-N. Bessagnet, et P. Roose (2021). Proposition d'un modèle de trajectoires multi-aspects et multi-niveaux appliqué au tourisme. In M. Lefrançois (Ed.), *Journées Francophones d'Ingénierie des Connaissances (IC) Plate-Forme Intelligence Artificielle (PFIA'21)*, Bordeaux, France, pp. pp 56–64.

Cayèré, C., C. Sallaberry, C. Faucher, M.-N. Bessagnet, P. Roose, et M. Masson (2021). Multilevel and multiple aspect semantic trajectory model : application to the tourism domain. *ISPRS International Journal of Geo-Information*.

Girardin, F., F. Calabrese, F. Dal Fiore, C. Ratti, et J. Blat (2009). Digital Footprinting : Uncovering Tourists with User-Generated Content. *Pervasive Computing, IEEE 7*, 36–43.

Kaddouri, L., J.-Y. Blaise, P.-A. Davoine, H. Mathian, et C. Saint-Marc (2014). *État des lieux des représentations dynamiques des temporalités des territoires*. Ph. D. thesis, UMR 7300 ESPACE ; UMR 3495 MAP-GAMSAU ; Laboratoire d'Informatique de Grenoble.

Kraak, M.-J. (2003a). Geovisualization illustrated. *ISPRS journal of photogrammetry and remote sensing 57*(5-6), 390–399.

Kraak, M.-J. (2003b). The space-time cube revisited from a geovisualization perspective. In *Proc. 21st International Cartographic Conference*, pp. 1988–1996. Citeseer.

Liu, H., Y. Gao, L. Lu, S. Liu, H. Qu, et L. M. Ni (2011). Visual analysis of route diversity. In *2011 IEEE conference on visual analytics science and technology (VAST)*, pp. 171–180. IEEE.

Menin, A., S. Chardonnel, P.-A. Davoine, M. Ortega, E. Duble, et L. Nedel (2020). estime : une approche visuelle, interactive et modulable pour l'analyse multi-points de vue des mobilités quotidiennes. *Geomatica 74*(3), 65–86.

Nusrat, S. et S. Kobourov (2016). The state of the art in cartograms. In *Computer Graphics Forum*, Volume 35, pp. 619–642. Wiley Online Library.

Pistre, P., H. Commenges, D. Guerrero, et L. Proulhac (2015). Operational definitions of time for longitudinal data analysis : Illustration in the field of spatial mobilities.

Roth, R. (2010). *Dot density maps*, pp. 787–790.

Summary

In this paper, we present the design and development of mobility data visualization components integrated into a generic and modular processing pipeline. This work took place within the framework of the DA3T interdisciplinary project (both geography and computer science) of the *Nouvelle-Aquitaine* region (*France*) aiming at the improvement of the valorization and the management of tourist cities. We discuss related works on spatial, temporal and spatio-temporal visualization modes and then make several proposals to improve these modes in order to meet the specific analysis requirements of tourist cities. This includes the implementation of a highly customizable spatio-temporal visualization cube but also the design of multi-mode heat maps. The experiment conducted on a touristic dataset (City of La Rochelle visitors' data) to prove the relevance of these visualization modes is then exposed.

Fouille de séquences temporelles avec l'AFC et l'outil GALACTIC

Salah Eddine Boukhetta, Christophe Demko, Karell Bertet, Jérémy Richard et Cécile Cayèré

*Laboratory L3i, La Rochelle University, La Rochelle, France

Résumé. Dans cet article, nous nous intéressons à l'analyse de données séquentielles temporelles en utilisant GALACTIC, un nouveau outil basé sur l'Analyse Formelle de Concepts (AFC) pour calculer un treillis de concepts à partir de données hétérogènes et complexes. Inspiré de la théorie de patrons, GALACTIC exploite des données décrites par des prédicats et est composé d'un système d'extensions pour une intégration facile de nouvelles caractéristiques de données et de leurs descriptions. Dans ce travail, nous utilisons GALACTIC pour analyser des données séquentielles temporelles, où chaque élément x_i de la séquence est associé à un horodatage t_i: $s = \langle (t_i, x_i) \rangle_{i \leq n}$. Nous introduisons des descriptions et des stratégies dédiées aux séquences temporelles. Nous montrons sur certains jeux de données que la stratégie de sous-séquence avec contrainte de distance permet de générer des treillis de bonne qualité.

1 Introduction

Les séquences apparaissent dans de nombreux domaines : séquences de mots dans un texte, trajectoires, navigation sur internet ou achat de produits dans un supermarché. Une séquence est une succession $\langle x_i \rangle$ de symboles ou d'ensembles. L'exploration de séquences vise à trouver des motifs fréquents dans un ensemble de données de séquences.

L'algorithme GSP (Srikant et Agrawal, 1996), est le premier algorithme d'extraction de sous-séquences fréquentes. D'autres algorithmes ont ensuite été proposés pour améliorer GSP, comme PrefixSpan (Pei et al., 2001), SPADE (Zaki, 2001), etc. Ces algorithmes prennent en entrée un ensemble de séquences et un seuil de support minimum, et génèrent tous les motifs séquentiels fréquents. Les motifs fermés, qui représentent la même information sous une forme réduite, donnent lieu à des approches plus efficaces pour extraire des motifs séquentiels fermés, qui sont des sous-séquences maximales, comme CloSpan (Yan et al., 2003), BIDE (Wang et Han, 2004), etc. Les contraintes de motifs proposent d'extraire les motifs vérifiant des contraintes d'entrée telles que la taille maximale d'une sous-séquence, une expression régulière, un écart entre les éléments, etc (Bonchi et al., 2006; Ugarte et al., 2017).

Nous avons étudié les séquences temporelles $\langle (t_i, x_i) \rangle$ où l'élément x_i est associé à un horodatage t_i. L'information temporelle peut améliorer l'expressivité des motifs, et le défi est alors d'extraire des motifs séquentiels avec une information temporelle. Des

intervalles sont introduits, représentant un écart ou un retard entre les éléments. Par exemple $(a, [t_s, t_e], b)$ dénote un motif séquentiel (a, b) qui se produit fréquemment avec un retard ou un temps de transition dans l'intervalle $[t_s, t_e]$. Les chroniques ont été introduites par Dousson (Dousson et al., 1993), où l'intervalle représente un retard entre les items, et le problème est de découvrir toutes les contraintes de temps possibles. Une chronique peut évidemment être représentée par un graphe. Dans un état de l'art récent et complet, Guyet (Guyet, 2020) a identifié deux types de problèmes dans l'exploration de motifs temporels. Un premier problème est la découverte complète de relations temporelles, comme par exemple la découverte complète de chroniques (Cram et al., 2012; Sahuguède et al., 2018). Le deuxième vise à extraire des motifs où l'information temporelle est ajoutée aux motifs de sous-séquences extraits par des approches classiques de fouille de séquences, qui correspondent à des chroniques «linéaires», comme dans le système FACE (Dousson et Duong, 1999) et dans (Hirate et Yamana, 2006). Citons également (Giannotti et al., 2006; Yen et Lee, 2013) qui propose d'extraire les valeurs représentatives des transitions en utilisant des méthodes de clustering. Guyet propose également NegPSpan (Guyet et Quiniou, 2020) pour extraire les motifs séquentiels négatifs et QTempIntMiner (Guyet et Quiniou, 2008) pour extraire les informations temporelles.

Récemment, certaines approches de fouille de séquences se positionnent dans le cadre de l'analyse formelle de concepts (AFC), basée sur le formalisme des structures de motifs (Ganter et Kuznetsov, 2001; Kaytoue et al., 2015). L'AFC apparaît en 1982 (Wille, 1982), puis dans l'ouvrage de Ganter et Wille 1999 (Ganter et Wille, 1999), où la notion de concept, de treillis de concepts et des règles d'associations sont formellement définies et étudiées. L'AFC est une branche de la théorie des treillis qui apparaît dans le livre de Barbut et Monjardet en 1970 (Barbut et Monjardet, 1970), où l'opérateur de fermeture est identifié comme une notion centrale pour établir des liens entre les données binaires, les treillis et les bases de règles (Bertet et al., 2018). Le formalisme des structures de motifs (Ganter et Kuznetsov, 2001) et de la navigation conceptuelle abstraite (NCA) (Ferré, 2002) étendent l'AFC pour traiter des données non binaires, où les données sont décrites par des motifs communs. Dans (Casas-Garriga, 2005), les auteurs présentent les Ordres Partiels Fermés (OPC) comme un graphe pour les données séquentielles où chaque chemin dans le graphe représente un motif séquentiel. Certaines approches visent à découvrir et à construire des motifs OPC en utilisant l'AFC. Citons également RCA-Seq dans (Nica et al., 2020) qui étend l'analyse relationnelle des concepts (ARC) à l'exploration de données séquentielles. Nous pouvons citepr des travaux d'extraction de trajectoires de soins médicaux (Buzmakov et al., 2013), d'extraction de séquences pour découvrir des motifs rares (Codocedo et al., 2017), et des études sur des séquences démographiques (Gizdatullin et al., 2017).

Cependant, le nombre de motifs générés est énorme, souvent intraitables (Kaytoue, 2020), on parle de «deluge de motifs» et le besoin d'approches permettant de diriger la recherche vers les motifs les plus pertinents représente un défi scientifique actuel. En effet, avec deux décennies de recherche sur l'exploration de séquences et de la puissance de calcul, les algorithmes d'exploration de séquences sont maintenant plus efficaces pour traiter d'énormes bases de données de séquences. Le problème passe du déluge de données au déluge de motifs, et les études récentes se concentrent davan-

tage sur l'extraction de motifs les plus pertinents. Inspiré par les structures de motifs, l'algorithme NextPriorityConcept (Demko et al., 2020) propose une approche de *pattern mining* pilotée par l'utilisateur pour des données hétérogènes et complexes avec différentes stratégies d'exploration visant à réduire le nombre de concepts tout en conservant la propriété de treillis. Dans un travail récent (Boukhetta et al., 2020a,b), nous avons proposé une nouvelle approche d'exploration de séquences utilisant l'algorithme NextPriorityConcept, la description correspond aux sous-séquences maximales communes, et plusieurs stratégies sont proposées, de la stratégie naïve générant tous les motifs possibles, à des stratégies plus spécifiques réduisant le nombre de motifs.

Dans cet article, nous proposons une nouvelle approche de fouille de séquences temporelles, avec des descriptions et des stratégies dédiées aux séquences temporelles. Nous extrayons des motifs fermés composés de sous-séquences distantes communes maximales, qui correspondent à des chroniques linéaires, avec un intervalle représentant un retard possible entre deux éléments consécutifs. Nous proposons deux stratégies d'exploration de motifs, la stratégie *Naïve* et la stratégie *Milieu*. Alors que la stratégie *Naïve* génère tous les motifs possibles, La stratégie *Milieu* vise à réduire les motifs dans une approche de découverte de motifs. Afin d'évaluer le treillis de motifs obtenu, nous utilisons des mesures de qualité non supervisées : la *stabilité logarithmique globale*, la *représentabilité* et la *distinctivité* introduites par (Boukhetta et al., 2021). La Section 2 présente les définitions de base relatives aux séquences temporelles et distancielles. Une brève description de l'algorithme NextPriorityConcept est et donnée dans la Section 3. La Section 4 est consacrée à la définition des descriptions et des stratégies. Les résultats expérimentaux sont présentés dans la Section 5.

2 Séquences

Une **séquence** $s\langle x_i\rangle_{i\leq n}$ est une liste d'événements ordonnés appartenant à un alphabet Σ, Une **séquence temporelle** $s = \langle(t_i, x_i)\rangle_{i\leq n}$ est une séquence où chaque élément x_i est associé à un horodatage t_i. Pour éviter toute confusion lors du traitement de plusieurs séquences, nous allons introduire la notation $s = \langle(t_i^s, x_i^s)\rangle_{i\leq n_s}$. Une **séquence distancielle** $d(s)$ d'une séquence temporelle est définie par $d(s) = \langle(x_i, \lfloor d_i \rceil), (x_n)\rangle_{i<n}$ avec $d_i = t_{i+1} - t_i$. L'exemple de la Table 1 est une partie d'un ensemble de données représentant les actions quotidiennes des individus du laboratoire L3i [1], où le horodatage indique une heure dans la journée et les actions quotidiennes peuvent être : $\Sigma = \{Wakeup(Wa), Breakfast(B), Work(Wo), Dinner(D), Coffee(C), Lunch(L), Sports(Sp), Read(R), Sleep(Sl)\}$.

La séquence distancielle de $s1$ est $d(s_1) = \langle(Wa, 1), (B, 2), (L, 6), (D)\rangle$ ce qui signifie que l'individu 1, s'est réveillé, et après 1 heure il a pris son petit-déjeuner, puis après 2 heures il a déjeuné, et après 6 heures il a pris son dîner.

Les sous-séquences communes maximales sont une description concise et complète d'un ensemble de séquences (Boukhetta et al., 2020a) issues de la relation de sous-séquence, nous l'étendons aux sous-séquences distancielles. Une séquence $a = \langle x_i^a\rangle_{i\leq n_a}$ est sous-séquence d'une séquence $b = \langle x_i^b\rangle_{i\leq n_b}$ s'il existe des entiers $1 \leq i_0 < \cdots < i_{n_a} \leq n_b$ tels que $x_j^a = x_{i_j}^b$ pour $j \leq n_a$, et on écrit $a \sqsubseteq_s b$. Pour deux séquences temporelles $a = \langle(t_i^a, x_i^a)\rangle_{i\leq n_a}$, et $b = \langle(t_i^b, x_i^b)\rangle_{i\leq n_b}$ la séquence distancielle $d(a) =$

1. https ://l3i.univ-larochelle.fr/

Id	Séquences temporelles	Sous-séquence distancielle
s_1	$\langle(\mathbf{11},\mathbf{Wa}),(\mathbf{12},\mathbf{B}),(\mathbf{14},\mathbf{L}),(\mathbf{20},\mathbf{D})\rangle$	$\langle(Wa,1),(B,2),(L,6),(D)\rangle$
s_2	$\langle(\mathbf{10},\mathbf{Wa}),(\mathbf{11},\mathbf{B}),(\mathbf{14},\mathbf{L}),(20,Sp),(21,R),(\mathbf{22},\mathbf{D})\rangle$	$\langle(Wa,1),(B,3),(L,8),(D)\rangle$
s_3	$\langle(\mathbf{8},\mathbf{Wa}),(9,Wo),(10,C),(\mathbf{11},\mathbf{B}),(\mathbf{13},\mathbf{L}),(14,Sp),(\mathbf{21},\mathbf{D})\rangle$	$\langle(Wa,3),(B,2),(L,8),(D)\rangle$
s_4	$\langle(\mathbf{7},\mathbf{Wa}),(\mathbf{8},\mathbf{B}),(9,Wo),(\mathbf{13},\mathbf{L}),(\mathbf{20},\mathbf{D}),(22,Sl)\rangle$	$\langle(Wa,1),(B,5),(L,7),(D)\rangle$
s_5	$\langle(\mathbf{8},\mathbf{Wa}),(\mathbf{9},\mathbf{B}),(\mathbf{14},\mathbf{L}),(18,Sp),(\mathbf{21},\mathbf{D}),(23,Sl)\rangle$	$\langle(Wa,1),(B,5),(L,7),(D)\rangle$

TAB. 1 – *Exemples d'actions quotidiennes*

$\langle(x_i^a,\lfloor d_i^a\rceil),(x_{n_a}^a)\rangle_{i<n_a}$ est une sous-séquence distancielle de $d(b)=\langle(x_i^b,\lfloor d_i^b\rceil),(x_{n_b}^b)\rangle_{i<n_b}$ et on écrit $d(a)\sqsubseteq_d d(b)$ si $\langle x_i^a\rangle\sqsubseteq_s\langle x_i^b\rangle$ et $d_j^a=t_{i_{j+1}}^b-t_{i_j}^b$ avec $j<n_a$. Par exemple, $s_1'=\langle(9,Wa),(12,L),(18,D)\rangle$, avec $d(s_1')=\langle(Wa,\lfloor 3\rceil),(L,\lfloor 6\rceil),(D)\rangle$ est une sous-séquence distancielle de $d(s_1)$ (Table 1).

Pour un ensemble de séquences temporelles A, les sous-séquences distancielles exactes peuvent être rares car les éléments des sous-séquences doivent apparaître tous après le même laps de temps. Pour étendre cela aux **Sous-séquences Distancielles Communes** $SDC(A)$ nous introduisons l'intervalle $\lfloor d_i^{min},d_i^{max}\rceil$ de distances possibles :

$$SDC(A)=\{r=\langle(x_i^r,\lfloor d_i^{r,min},d_i^{r,max}\rceil),(x_{n_r})\rangle_{i<n_r}\}\text{ ou :}\quad(1)$$

— $r\sqsubseteq_s a\ \forall a\in A$
— $d_i^{r,min}=min(\{d_i^k:k\sqsubseteq_d a,\forall a\in A\})$
— $d_i^{r,max}=max(\{d_i^k:k\sqsubseteq_d a,\forall a\in A\})$

Dans la Table 1, $r=\langle(Wa,\lfloor 1,3\rceil),(B,\lfloor 2,5\rceil),(L,\lfloor 6,8\rceil),(D)\rangle$, est une sous-séquence distancielle commune de toutes les séquences, ce qui signifie que toutes les personnes prennent leur petit-déjeuner entre 1 et 3 heures après le réveil et qu'elles déjeunent entre 2 et 5 heures après.

3 L'algorithme NEXTPRIORITYCONCEPT

L'algorithme NEXTPRIORITYCONCEPT (Demko et al., 2020) calcule des concepts pour des données hétérogènes et complexes pour un ensemble d'objets G. Il s'inspire de l'algorithme de Bordat (Bordat, 1986), que l'on retrouve également dans les travaux de Linding (Linding, 2002), qui calcule de manière récursive les prédécesseurs immédiats d'un concept, en commençant par le concept du top $(G,\delta(G))$ contenant l'ensemble des objets, jusqu'à ce que plus aucun concept ne puisse être généré. L'utilisation d'une file de priorité assure que chaque concept est généré avant ses prédécesseurs et évite une récursivité inutile, et un mécanisme de propagation des contraintes assure que les infimum à deux concepts seront préservés NEXTPRIORITYCONCEPT calcule un treillis de concepts et se positionne donc dans le cadre de l'AFC. Ses principales caractéristiques sont :

Descriptions comme application générant des prédicats L'algorithme introduit la notion de *description* δ comme une application permettant de fournir des *prédicats* décrivant un ensemble d'objets $A\subseteq G$. Chaque concept $(A,\delta(A)$ est composé d'un sous-ensemble d'objets A et d'un ensemble de prédicats $\delta(A)$ les décrivant, correspondant à leur motif. Cette utilisation générique des prédicats permet de considérer en entrée des données hétérogènes, c'est-à-dire des données

numériques, discrètes ou plus complexes. Un concept $(A, \delta(A)$ peut être interprété comme une coque convexe généralisée, où chaque frontière correspond à un prédicat, et les éléments à l'intérieur de la coque correspondent aux objets qui vérifient tous les prédicats. Contrairement aux structures de motifs classiques, les prédicats ne sont pas calculés globalement dans une étape de pré-traitement, mais localement pour chaque concept.

Stratégies comme application générant des sélecteurs L'algorithme introduit également la notion de *stratégie* σ pour fournir des prédicats appelés *sélecteurs* décrivant les candidats pour la réduction d'objet d'un concept $(A, \delta(A)$ c'est-à-dire les prédécesseurs de $(A, \delta(A))$ dans le treillis de motifs. Un sélecteur propose une façon d'affiner la description pour un ensemble réduit $A' \subset A$ d'objets. Plusieurs stratégies sont possibles pour générer les prédécesseurs d'un concept, allant de la stratégie naïve classiquement utilisée dans l'AFC qui considère tous les prédécesseurs possibles, à des stratégies permettant d'obtenir peu de prédécesseurs et des treillis plus petits. Les sélecteurs ne sont utilisés que pour la génération des prédécesseurs, ils ne sont conservés ni dans la description ni dans l'ensemble final des prédicats.

Le résultat principal de (Demko et al., 2020) indique que l'algorithme NextPriorityConcept calcule le contexte formel $\langle G, P, I_P \rangle$ et son treillis de concepts (où P est l'ensemble des prédicats décrivant les objets dans G, et $I_P = \{(a, p),\ a \in G, p \in P\ :\ p(a)\}$ est la relation entre les objets et les prédicats) si la description δ vérifie $\delta(A) \sqsubseteq \delta(A')$ pour $A' \subseteq A$. Le temps d'exécution de l'algorithme NextPriorityConcept a une complexité $O(|\mathcal{B}|\,|G|\,|P|^2\,(c_\sigma + c_\delta))$ (où $\mathcal{B}$ est le nombre de concepts, c_σ est le coût de la stratégie et c_δ est le coût de la description), et une mémoire spatiale en $O(w\,|P|^2)$ (où w est la largeur du treillis de concepts).

4 NextPriorityConcept pour les séquences

Afin d'extraire des séquences temporelles à l'aide de l'algorithme NextPriorityConcept, nous devons définir des descriptions et des stratégies pour un ensemble G de séquences temporelles en entrée dont la taille est plus petite que n (n la taille maximal des séquences de G).

Une description δ est une fonction $\delta : 2^G \to 2^P$ qui définit un ensemble de prédicats $\delta(A)$ décrivant tout les sous-ensembles $A \subseteq G$ de séquences temporelles.

Une stratégie σ est une fonction $\sigma : 2^G \to 2^P$ qui définit un ensemble de sélecteurs $\sigma(A)$ pour sélectionner des sous-ensembles stricts $A' \subset A$ comme prédécesseurs candidats de tout concept $(A, \delta(A))$ dans le treillis de concepts.

Les prédicats et les sélecteurs sont calculés en utilisant la relation de sous-séquence de la forme *«est une sous-séquence distancielle de»*. Pour une meilleure lisibilité, les ensembles $\delta(A)$ et $\sigma(A)$ seront traités soit comme des ensembles de prédicats/sélecteurs, soit comme des ensembles de sous-séquences, ils peuvent réciproquement être déduits les uns des autres.

4.1 Description pour les séquences temporelles

Les sous-séquences distancielles communes maximales $SDCM(A)$ d'un ensemble de séquences temporelles A forment la description $\delta_{\mathrm{D}}(A)$. Cette description peut ensuite être paramétrée par deux contraintes. Une contrainte de *fenêtre* glissante, qui est utilisée pour limiter la recherche de sous-séquences à une partie spécifique des séquences temporelles. Par exemple, si nous voulons analyser les données des touristes et obtenir les activités essentielles par jour, nous pouvons utiliser une fenêtre de $24h$. Et une contrainte d'*écart* maximal qui définit la distance maximale autorisée entre deux éléments de la sous-séquence. Pour chaque concept les contraintes décident si un objet reste dans le concept ou non. Ici, ils sont utilisées au milieu de processus d'analyse, tous comme le calcul de la description. Formellement, une description *Distancielle* est définie pour un ensemble de séquences temporelles $A \subseteq G$ par :

Description *Distancielle***.**

$$\delta_{\mathrm{D}}(A) = \{p = \langle (x_i^p, \lfloor d_i^{p,min}, d_i^{p,max} \rceil), (x_n^p) \rangle_{i<n_p} : p \in SDC(A), p \text{ maximale}\} \tag{2}$$

Contrainte de *fenêtre***.** Pour $A \subseteq G$ et une fenêtre w :

$$\delta_{\mathrm{Dw}}(A, w) = \{p : p \in \delta_{\mathrm{D}}(A) \text{ et } \sum_{i=0}^{n-1} d_i^{p,max} \leq w\} \tag{3}$$

Contrainte d'*écart***.** Pour $A \subseteq G$ et un écart g :

$$\delta_{\mathrm{Dg}}(A, g) = \{p : p \in \delta_{\mathrm{D}}(A) \text{ et } d_i^{p,max} \leq g, \text{ pour } 0 \leq i < n\} \tag{4}$$

Pour calculer la description d'un ensemble A de séquences temporelles, nous itérons sur A en mettant à jour les sous-séquences résultantes de $\delta_{\mathrm{D}}(A)$ avec les parties communes. Par conséquent, la complexité de la description est $c_{\delta_{\mathrm{D}}} = O(s\,|A|\,log(|A|)) \leq O(s\,|G|\,log(|G|))$ où s est la taille maximale des sous-séquences calculées.

La description doit vérifier $\delta(A) \sqsubseteq \delta(A')$ pour $A' \subseteq A$:

Proposition 1 *Pour $A' \subseteq A \subseteq G$, nous avons $\delta_D(A) \sqsubseteq \delta_D(A')$*

Preuve: Soit A et A' sont deux sous-ensembles de séquences tels que $A' \subseteq A$. Soit $c \in \delta_{\mathrm{D}}(A)$, c'est-à-dire, c est une sous-séquence distancielle commune maximale de A. De $A' \subseteq A$ nous pouvons déduire que c est aussi une sous-séquence des séquences de A', mais c n'est pas forcément une sous-séquence distancielle maximale de A'. Si c est une sous-séquence maximale de A' alors $c \in \delta_{\mathrm{D}}(A')$. Sinon, il existe $c' \in \delta_{\mathrm{D}}(A')$ tel que c est sous-séquence de c'. Dans ces deux cas, on peut déduire que, $\delta_{\mathrm{D}}(A) \sqsubseteq \delta_{\mathrm{D}}(A')$. □

4.2 Stratégies et sélecteurs pour les séquences temporelles

Les stratégies sont utilisées par l'algorithme NextPriorityConcept pour affiner chaque concept $(A, \delta(A))$ en concepts comportant moins d'objets et des descriptions plus spécifiques. Ils peuvent générer des sous-séquences avec plus d'éléments comme pour les séquences classiques ou avec des distances plus courtes entre les éléments en exploitant l'information de distance. Nous définissons deux stratégies : La stratégie *Naïve* génère toutes les sous-séquences distancielles possibles. La stratégie *Milieu* génère des sous-séquences en divisant les distances par deux.

Stratégie *Naïve.* Pour $A \subseteq G$ et un pas pas :

$$\sigma_{\text{SDN}}(A, pas) = X \cup Y \cup Z \text{ où} \tag{5}$$

$$X = \{s : \langle x_i^s \rangle = \langle x_i^p \rangle, \forall p \in \delta_{\text{D}}(A) \text{ et } d_i^{s,min} = d_i^{p,min} + pas\} \tag{6}$$

$$Y = \{s : \langle x_i^s \rangle = \langle x_i^p \rangle, \forall p \in \delta_{\text{D}}(A) \text{ et } d_i^{s,max} = d_i^{p,max} - pas\} \tag{7}$$

$$Z = \{s + a : \langle x_i^s \rangle = \langle x_i^p \rangle, \forall p \in \delta_{\text{D}}(A) \text{ et } a \in \Sigma\} \tag{8}$$

La stratégie *Naïve* est définie pour un pas $pas = 1$.

Stratégie *Milieu.* est définie pour un pas égale au milieu des distances :

$$\sigma_{\text{SDM}}(A) = \{s : s \in \sigma_{\text{SDN}}(A, M), M = (d_i^{s,max} - d_i^{s,min})/2\} \tag{9}$$

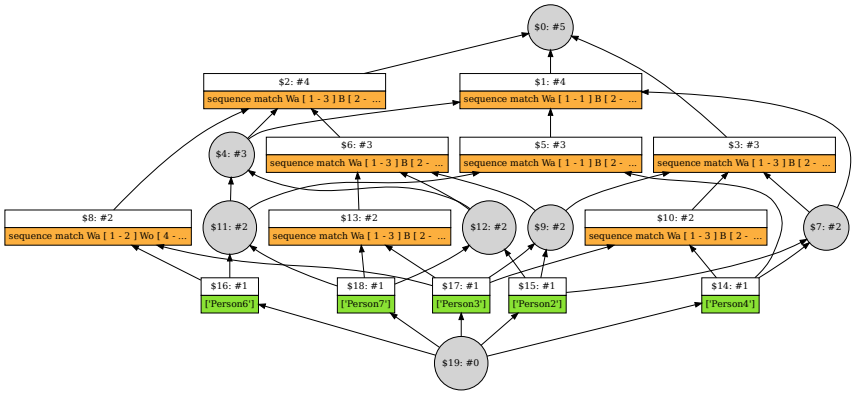


FIG. 1 – *Treillis de concepts réduit obtenu avec la stratégie Naïve et la description Distancielle pour l'exemple de la Table 1.*

La Figure 1 montre le treillis de concepts réduit généré avec la stratégie *Naïve* et la description *Distancielle* pour l'exemple de la Table 1. Chaque concept $(A, \delta_{\text{D}}(A))$ est composé d'un ensemble de séquences A, ainsi que de leurs sous-séquences distancielles communes maximales $\delta_{\text{D}}(A)$. Le treillis contient 20 concepts. Le concept \$1 a les concepts \$4, \$5 et \$7 comme prédécesseurs, qui partagent la même sous-séquence $\langle Wa, B, L, D \rangle$ avec des distances différentes entre les éléments (Table 2). La Figure 2 montre le treillis obtenu avec la stratégie *Milieu* et la description *Distancielle* pour l'exemple de la Table 1. Le treillis est composé de 10 concepts. Le concept \$1 contient la description : *sequence match* $Wa[1-3]B[2-5]L[1-6]S[2-7]D$, alors que la même description a été générée dans le concept \$6 avec la stratégie *Naïve*.

5 Expérimentations

Dans cette section, nous évaluons nos descriptions et nos stratégies. Nous utilisons `GALACTIC`[2] (**GA**lois **LA**ttices, **C**oncept **T**heory, **I**mplicational systems and **C**losures),

2. `https://galactic.univ-lr.fr`

Id concept	Description
\$1	$\langle (Wa, \lfloor 1-1 \rceil), (B, \lfloor 2-5 \rceil), (L, \lfloor 6-8 \rceil), (D) \rangle$
\$4	$\langle (Wa, \lfloor 1-1 \rceil), (B, \lfloor 3-5 \rceil), (L, \lfloor 6-8 \rceil), (D) \rangle$
\$5	$\langle (Wa, \lfloor 1-1 \rceil), (B, \lfloor 2-5 \rceil), (L, \lfloor 6-7 \rceil), (D) \rangle$
\$7	$\langle (Wa, \lfloor 1-1 \rceil), (B, \lfloor 2-3 \rceil), (L, \lfloor 6-8 \rceil), (D) \rangle$

TAB. 2 – *Quelques concepts du treillis généré à l'aide de la stratégie Naïve et la description Distancielle pour l'exemple de la Table 1.*

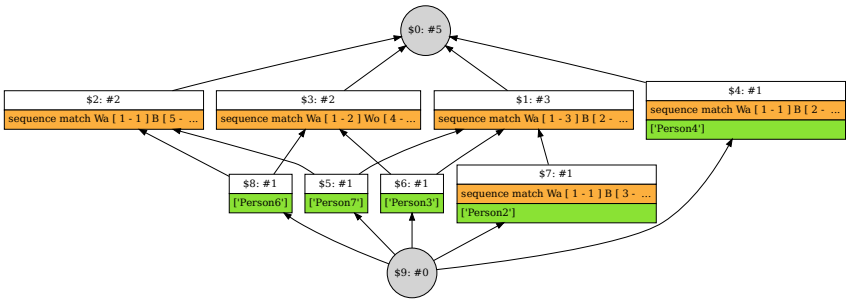


FIG. 2 – *Treillis de concepts réduit obtenu avec la stratégie Milieu et la description Distancielle pour l'exemple de la Table 1.*

une plateforme de développement de l'algorithme NEXTPRIORITYCONCEPT qui, associée à un système d'extensions, permet d'intégrer de nouveaux types de données avec des descriptions et des stratégies spécifiques. Nous avons implémenté nos extensions de description et de stratégie pour les séquences temporelles. Les expériences ont été réalisées sur une machine Intel Core i7 2.20GHz avec 32GB de mémoire principale. Nous avons mené nos expériences sur deux ensembles de données de séquences :

`Cité du Vin` est issu du musée «La Cité du Vin» à Bordeaux, France [3], et a été collecté à partir des visites effectuées sur une période d'un an. Le musée est un grand «open-space», où les visiteurs sont libres d'explorer le musée comme ils le souhaitent. Le jeu de données contient 10000 séquences, avec une taille moyenne de 9.

`Catch-me-if-you-can` (`Catch-me` en abrégé) composé de séquences de navigation d'utilisateurs sur des pages web, disponible sur le site de KAGGLE [4] (Kahn et al., 2016). Le jeu de données contient 82797 séquences, avec une taille moyenne de 6,4.

Dans nos expérimentations, nous faisons varier la taille de l'ensemble de données (nombre de séquences) à l'aide d'une fonction aléatoire, puis chaque expérience est réalisée dix fois, et nous calculons la moyenne de chaque valeur de mesure. Nous utilisons la stabilité logarithmique globale, la représentabilité et la distinctivité (Boukhetta et al., 2021) car nous nous concentrons sur la réduction de la taille du treillis, toute en préservant la stabilité globale, permettant de générer plus de prédicats (qui correspondent aux sup-irréductibles du treillis) tout en distinguant entre les objets (qui correspondent aux inf-irréductibles du treillis). Deux expérimentations sont réalisées, la comparaison

3. `https://www.lacitepduvin.com/en`
4. `https://www.kaggle.com/danielkurniadi/catch-me-if-you-can`

entre les stratégies, et l'impact de l'utilisation de contraintes. Notre description et nos stratégies sont très spécifiques et nous n'avons pas trouvé d'études utilisant l'AFC pour analyser des séquences temporelles et explorer les ensembles de données comme nous le faisons. Nous avons publié une comparaison entre les approches classiques d'exploration de séquences fermées et nos méthodes (Boukhetta et al., 2020a).

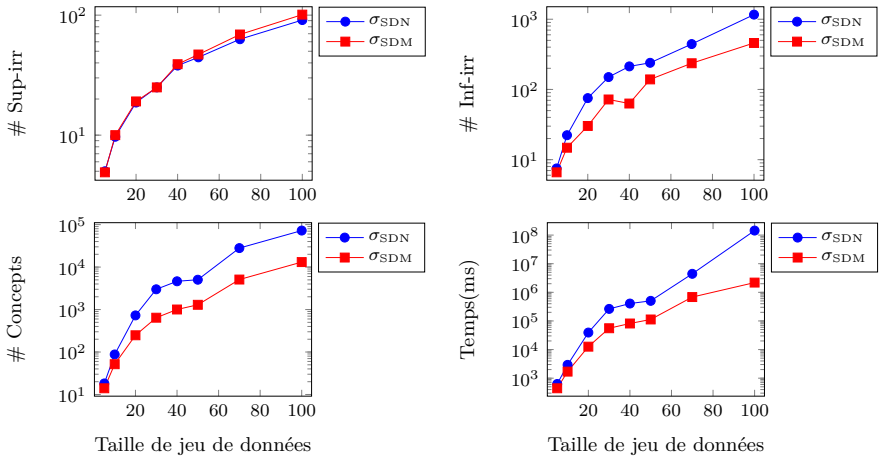


FIG. 3 – *Temps moyen d'exécution et nombre moyen de concepts et d'irréductibles générés par les deux stratégies Naïve et Milieu et la description Distancielle avec le jeu de données* `Cité-du-Vin`.

5.1 Comparaison des stratégies

En faisant varier le nombre de séquences, la Figure 3 montre le temps d'exécution, le nombre de concepts et le nombre d'irréductibles générés par les deux stratégies σ_{SDN} et σ_{SDM} et la description δ_{D} avec le jeu de données `Cité du Vin`. Dans la Figure 3 et le Tableau 3, on peut observer que la stratégie *Milieu* réduit la taille du treillis avec moins de sup-irréductibles. La stratégie Naïve génère plus d'irréductibles car elle calcule tous les concepts possibles. Le nombre de inf-irréductibles est le même avec les deux stratégies, ce qui signifie qu'elles ont la même capacité à distinguer les séquences initiales. Par conséquent, la stratégie *Milieu* est plus rapide que la stratégie *Naïve*, car elle génère moins de concepts. De plus, on peut observer que la stabilité logarithmique dans la Table 3 reste stable pour les deux stratégies, même si la stratégie du *Milieu* génère moins de concepts.

Dans la Figure 4, nous pouvons observer que les mesures de représentabilité et de distinctivité diminuent lorsque la taille des données augmente. La distinctivité est plus élevée avec la stratégie *Milieu* qu'avec la stratégie *Naïve*. Cela signifie que la première maintient un nombre plus élevé de sup irréductibles tout en réduisant la taille du treillis. La représentabilité est légèrement meilleure aussi pour le jeu de données Catch-me. Pour le jeu de données Wine-city, les deux mesures sont plus élevées avec la stratégie *Milieu* par rapport à la stratégie *Naïve*. Ceci est dû au fait que la stratégie *Naïve* génère beaucoup plus de prédicats que la stratégie *Milieu*.

Taille	stratégie	# concepts	# sup	# inf	temps(ms)	stabilité logarithmique globale
100	σ_{SDN}	172.2	93.8	101.1	5293133	0.8167676
	σ_{SDM}	171	93.9	100.4	5169953	0.8167675
300	σ_{SDN}	565.3	261.6	273.6	68418978	0.817374000
	σ_{SDM}	553	263.8	269.9	66631920	0.817374014
500	σ_{SDN}	984	416.4	420.8	240743394	0.81
	σ_{SDM}	964	421.6	416.5	234808014	0.81

TAB. 3 – *Nombre de moyen de concepts, temps moyen d'exécution, sup et inf irréductibles générés par les stratégies Naïve et Milieu en utilisant le jeu de données* `Catch-me`.

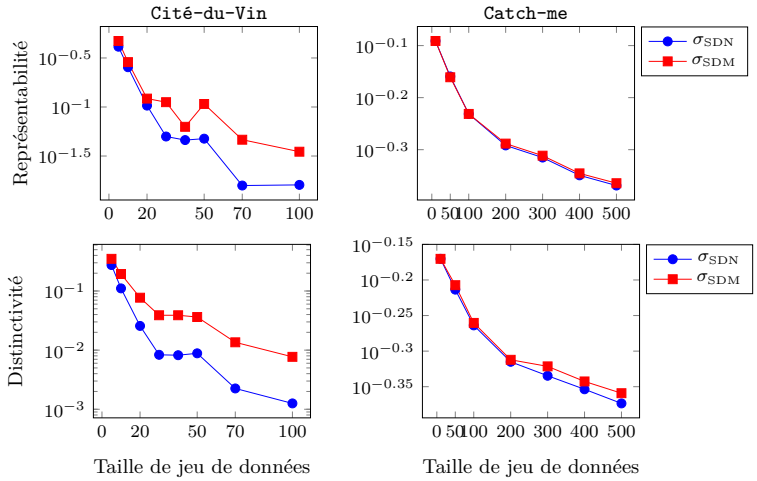


FIG. 4 – *Représentabilité et Distinctivité avec la description Distancielle et des deux stratégies Naïve et Milieu pour les jeux de données* `Catch-me` *et* `Cité-du-Vin`.

5.2 Contraintes

Une autre façon de réduire les motifs est de limiter la description par une fenêtre ou un écart. La Figure 5 montre la taille du treillis généré en utilisant la stratégie *Naïve* et la description *Distancielle* où la fenêtre et l'écart varient. Nous utilisons 3 valeurs de fenêtre : $w = \{3600s, 2700s, 1800s\}$ et 3 valeurs d'écarts : $g = \{120s, 600s, 1800s\}$. Nous pouvons voir que le nombre de concepts est considérablement réduit. Pour un jeu de données de taille 100, le nombre de concepts est réduit de 72427 (sans fenêtre) à 495 (avec fenêtre de $1800s$). De manière similaire, l'utilisation d'un écart permet de réduire la taille des treillis en moyenne à 295.75 avec un écart de $120s$.

6 Conclusion

Dans cet article, nous avons présenté une nouvelle approche d'exploration de séquences temporelles utilisant l'algorithme NEXTPRIORITYCONCEPT. Cet algorithme permet un calcul de motifs génériques par le biais de descriptions et de stratégies

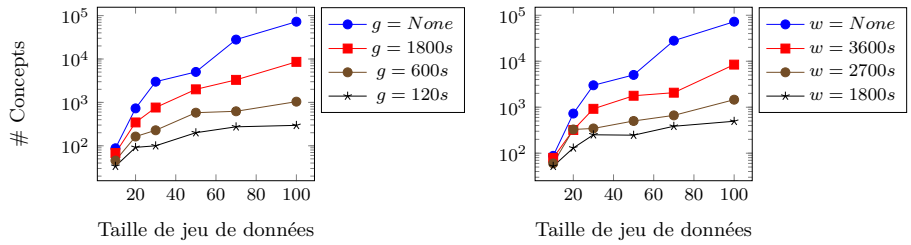


FIG. 5 – *Nombre moyen de concepts utilisant la description Distancielle avec les contraintes fenêtre et écart et la stratégie Naïve pour le jeu de données* `Cité-du-Vin`.

spécifiques. Nous avons défini une description distancielle qui décrit un ensemble de séquences temporelles par leurs sous-séquences distancielles communes maximales, avec des distances entre les éléments. Cette description est étendue avec des contraintes de fenêtre et d'écart. Nous avons proposé également deux stratégies d'exploration. La stratégie *Naïve* génère tous les motifs possibles et la stratégie *Middle* réduit la distance au milieu. Nous pouvons observer que la stratégies *Middle* et la description avec des contraintes permettent de réduire le nombre de concepts tout en maintenant la qualité des concepts. Ces résultats ont été rendus possibles par l'utilisation de l'algorithme NEXTPRIORITYCONCEPT qui permet de se concentrer sur les données (descriptions et stratégies) pour l'exploration et la découverte de motifs. Dans des travaux futurs, nous souhaitons proposer une approche orientée utilisateur qui permettrait à l'analyste de choisir et de tester plusieurs stratégies et descriptions en variant les contraintes d'écart et de fenêtre pour trouver les motifs les plus pertinents.

Remerciements

Nous tenons à remercier le musée de la Cité du Vin de nous avoir permis d'utiliser leurs données dans le cadre de notre expérimentation.

Références

Barbut, M. et B. Monjardet (1970). *Ordres et classifications : Algèbre et combinatoire*. Hachette, Paris. 2 tomes.

Bertet, K., C. Demko, J. Viaud, et C. Guérin (2018). Lattices, closures systems and implication bases : A survey of structural aspects and algorithms. *Theor. Comput. Sci. 743*, 93–109.

Bonchi, F., F. Giannotti, C. Lucchese, S. Orlando, R. Perego, et R. Trasarti (2006). Conquest : a constraint-based querying system for exploratory pattern discovery. In *22nd International Conference on Data Engineering (ICDE'06)*, pp. 159–159.
fr

Bordat, J.-P. (1986). Calcul pratique du treillis de Galois d'une correspondance. *Mathématiques et Sciences humaines 96*, 31–47.

Boukhetta, S. E., C. Demko, K. Bertet, J. Richard, et C. Cayèré (2021). Temporal sequence mining using fca and galactic. In *International Conference on Conceptual Structures*, pp. 185–199. Springer.

Exploration de séquences temporelles à l'aide de FCA et de GALACTIC

Boukhetta, S. E., Ch. Demko, J. Richard, et K. Bertet (2020a). Sequence mining using fca and the NextPriorityConcept algorithm. In *Concept Lattices and Their Applications 2020*, Volume 2668, pp. 209–222.

Boukhetta, S. E., J. Richard, Ch. Demko, et K. Bertet (2020b). Interval-based sequence mining using fca and the NextPriorityConcept algorithm. In *FCA4AI : What can FCA do for AI ?*, Volume 2729, pp. 91–102.

Buzmakov, A., E. Egho, N. Jay, S. O. Kuznetsov, A. Napoli, et C. Raïssi (2013). On projections of sequential pattern structures (with an application on care trajectories).

Casas-Garriga, G. (2005). Summarizing sequential data with closed partial orders. In *Proceedings of the SIAM International Conference on Data Mining*, pp. 380–391. SIAM.

Codocedo, V., G. Bosc, M. Kaytoue, J.-F. Boulicaut, et A. Napoli (2017). A proposition for sequence mining using pattern structures. In *International Conference on Formal Concept Analysis*, pp. 106–121. Springer.

Cram, D., B. Mathern, et A. Mille (2012). A complete chronicle discovery approach : application to activity analysis. *Expert Systems 29*(4), 321–346.

Demko, Ch., K. Bertet, C. Faucher, J.-F. Viaud, et S. O. Kuznetsov (2020). NextPriority-Concept : A new and generic algorithm computing concepts from complex and heterogeneous data. *Theoretical Computer Science 845*, 1 – 20.

Dousson, C. et T. V. Duong (1999). Discovering chronicles with numerical time constraints from alarm logs for monitoring dynamic systems. In *IJCAI*, Volume 99, pp. 620–626. Citeseer.

Dousson, C., P. Gaborit, et M. Ghallab (1993). Situation recognition : Representation and algorithms. In *IJCAI : International Joint Conference on Artificial Intelligence*, Volume 93, pp. 166–172.

Ferré, S. (2002). *Systèmes d'information logiques : un paradigme logico-contextuel pour interroger, naviguer et apprendre*. Doctorat, Univ. of Rennes 1, France.

Ganter, B. et S. O. Kuznetsov (2001). Pattern structures and their projections. In *LNCS of International Conference on Conceptual Structures (ICCS'01)*, pp. 129–142.

Ganter, B. et R. Wille (1999). *Formal Concept Analysis, Mathematical foundations*. Springer Verlag, Berlin.

Giannotti, F., M. Nanni, D. Pedreschi, et F. Pinelli (2006). Mining sequences with temporal annotations. In *Proceedings of the ACM symposium on Applied computing*, pp. 593–597.

Gizdatullin, D., D. Ignatov, E. Mitrofanova, et A. Muratova (2017). Classification of demographic sequences based on pattern structures and emerging patterns. In *Supplementary Proceedings of 14th ICFCA*, pp. 49–66.

Guyet, T. (2020). Enhance sequential pattern mining with time and reasoning.

Guyet, T. et R. Quiniou (2008). Mining temporal patterns with quantitative intervals. In *IEEE International Conference on Data Mining Workshops*, pp. 218–227.

Guyet, T. et R. Quiniou (2020). Negpspan : efficient extraction of negative sequential patterns with embedding constraints. *Data Mining and Knowledge Discovery 34*(2), 563–609.

Hirate, Y. et H. Yamana (2006). Generalized sequential pattern mining with item intervals. *JCP : Journal of Computers 1*(3), 51–60.

Kahn, G., Y. Loiseau, et O. Raynaud (2016). A tool for classification of sequential data. In *European Conference on Artificial Intelligence (FCA4AI)*.

Kaytoue, M. (2020). *Contributions to Pattern Discovery*. Habilitation, Univ. of Lyon, France.

Kaytoue, M., V. Codocedo, A. Buzmakov, J. Baixeries, S. O. Kuznetsov, et A. Napoli (2015). Pattern structures and concept lattices for data mining and knowledge processing. In *ECML-PKDD : European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*.

Linding, C. (2002). Fast concept analysis. In *Working with Conceptual Structures-Contributions to ICC*, pp. 235–248.

Nica, C., A. Braud, et F. Le Ber (2020). Rca-seq : An original approach for enhancing the analysis of sequential data based on hierarchies of multilevel closed partially-ordered patterns. *Discrete Applied Mathematics 273*, 232–251.

Pei, J., J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, et M.-C. Hsu (2001). Prefixspan : Mining sequential patterns efficiently by prefix-projected pattern growth. In *icccn*, pp. 0215. IEEE.

Sahuguède, A., E. Le Corronc, et M.-V. Le Lann (2018). An ordered chronicle discovery algorithm. In *3nd ECML/PKDD Workshop on Advanced Analytics and Learning on Temporal Data, AALTD'18*.

Srikant, R. et R. Agrawal (1996). Mining sequential patterns : Generalizations and performance improvements. In *International conference on extending database technology*, pp. 1–17. Springer.

Ugarte, W., P. Boizumault, B. Crémilleux, A. Lepailleur, S. Loudni, M. Plantevit, C. Raïssi, et A. Soulet (2017). Skypattern mining : From pattern condensed representations to dynamic constraint satisfaction problems. *Artificial Intelligence*.

Wang, J. et J. Han (2004). Bide : Efficient mining of frequent closed sequences. In *Proceedings. 20th international conference on data engineering*, pp. 79–90. IEEE.

Wille, R. (1982). Restructuring lattice theory : an approach based on hierarchies of concepts. *Ordered sets*, 445–470. I. Rival (ed.), Dordrecht-Boston, Reidel.

Yan, X., J. Han, et R. Afshar (2003). Clospan : Mining : Closed sequential patterns in large datasets. In *Proceedings of the SIAM international conference on data mining*, pp. 166–177. SIAM.

Yen, S.-J. et Y.-S. Lee (2013). Mining non-redundant time-gap sequential patterns. *Applied Intelligence 39*(4), 727–738.

Zaki, M. J. (2001). Spade : An efficient algorithm for mining frequent sequences. *Machine learning 42*(1-2), 31–60.

Représentation et combinaison de l'information géographique pour l'apprentissage profond

Azelle Courtial*, Guillaume Touya*

* LASTIG Univ. Gustav Eiffel, ENSG, IGN, Champs-sur-Marne
prenom.nom@ign.fr,

Résumé. L'apprentissage profond permet maintenant de générer des cartes transformées à partir d'images d'autres cartes. Mais contrairement aux méthodes traditionnelles de prédiction de carte qui reposent sur des couches de données vectorielles stockées dans des bases de données géographiques, l'image ne transmet qu'une vue limitée des informations contenues dans la version vectorielle des données. Dans cet article, nous nous intéressons à la représentation de l'information géographique sous forme de tenseurs pour améliorer la génération de cartes par apprentissage profond. Nous proposons d'abord une stratégie alternative pour la création des données d'apprentissage : un ensemble de masques où chacun décrit les formes et positions d'un type d'objet géographique sur une même portion de carte (bâtiments, routes, ...). Nous étudions ensuite comment combiner de l'information géographique additionnelle dans les mécanismes d'apprentissage pour améliorer l'abstraction des cartes générées.

1 Introduction

L'apprentissage profond est de plus en plus utilisé pour générer de l'information géographique et notamment des cartes, afin d'accélérer les processus de production. Le problème de génération de cartes est souvent traité comme un problème de traduction d'image à image comme par exemple passer d'une image aérienne vers une carte (Isola et al., 2017). Néanmoins, de telles approches ne permettent pas à proprement parler de générer des cartes, mais plutôt des images dans le style d'une carte ou ressemblant à une carte. En effet, la création d'une carte inclut la symbolisation de l'information géographique, mais il s'agit surtout de l'abstraction d'une portion de terrain à une certaine échelle pour répondre à un certain objectif. Les méthodes de génération de cartes doivent apprendre à la fois la combinaison, l'adaptation à l'échelle cible (généralisation) et la représentation de l'information géographique dans l'image cible. Pour cela, l'image initiale doit permettre de discerner l'ensemble des informations nécessaires à ces processus.

Ces informations incluent non seulement la forme et la localisation des objets, mais aussi des caractéristiques implicites telles que leurs importances, leurs rôles dans le paysage, leurs relations spatiales, ... L'acquisition de ces connaissances et leur utilisation dans un processus automatique, sont des défis majeurs de la généralisation automatique (Sester, 2005; Weibel et al., 1995). Lorsque nous nous intéressons à la génération de cartes à partir d'images de

cartes plutôt que de données vectorielles, l'accès à ces informations est compliqué. Si certaines de ces informations sont implicitement visibles dans les exemples d'apprentissage (sinuosité d'une route, régularité d'un groupe de bâtiments, ...), il reste à s'assurer qu'elles sont comprises et utilisées correctement par le réseau d'apprentissage. Et surtout une part importante des informations est masquée soit par la représentation initiale des données (fonction des bâtiments, sens de circulation des tronçons routiers, ...), soit par le découpage de l'espace en tuiles (densité d'un îlot urbain, place d'un arc routier dans le réseau routier, ...), soit par le format image (limite des objets adjacents non-visibles, unicité de valeur de pixel en cas d'intersection d'objets géographiques, ...). Cette inadéquation entre la forme des données d'apprentissage et l'objectif d'apprentissage est la principale limite des premières expérimentations pour générer des nouvelles cartes par apprentissage (Courtial et al., 2021b).

Nous proposons donc dans cet article de tester comment une bonne combinaison de l'information géographique peut améliorer la génération de cartes en permettant d'intégrer des informations supplémentaires non contenues dans l'image, et/ou de mettre en valeur les relations spatiales entre objets géographiques. Cet article est organisé comme suit : d'abord, la partie 2 est un l'état de l'art qui décrit les limites des différentes approches de cartographie usant d'apprentissage profond, et comment la combinaison d'informations peut être utilisée pour améliorer la génération de cartes. Puis nous présentons nos expérimentations dans la section 3. Ensuite, les sections 4 et 5 présentent nos propositions pour (a) la structuration et la représentation des données initiales, (b) la combinaison d'informations additionnelles. Enfin, nous concluons sur l'importance de la représentation et de la combinaison des données initiales en apprentissage profond dans la section 6.

2 État de l'art

2.1 Apprentissage profond et cartographie

La traduction d'image aérienne vers une tuile dans le style de Google Maps figure parmi les nombreux exemples d'usage des premiers réseaux antagonistes de génération d'images (GAN) (Isola et al., 2017; Zhu et al., 2017). Toutes ces expérimentations, menées par des non-cartographes, démontrent la capacité de ces réseaux à reproduire les principales caractéristiques d'une carte, préserver l'information géographique saillante dans l'image initiale et reproduire une symbolisation. Plus tard, des variations de ces réseaux, spécialisées pour la cartographie tentent d'améliorer les résultats dans ce cas et s'intéressent particulièrement à la préservation de l'information géographique initiale. Par exemple, Chen et al. (2020) proposent une fonction d'apprentissage qui utilise le gradient des images pour décrire leurs structures et s'assurer de leurs préservations lors de la génération. Cette fonction peut être apparentée à une contrainte de préservation des relations topologiques de la carte. De façon similaire, Fu et al. (2019) proposent une fonction d'apprentissage qui s'assure de la consistance géométrique de la prédiction, c'est-à-dire qui applique une transformation géométrique de l'image en entrée et compare la prédiction pour cette image avec la transformation de l'image produite depuis l'image initiale. Cette caractéristique est une contrainte spatiale importante sur les données géographiques à une échelle fixe. Ganguli et al. (2019) proposent quant à eux un réseau plus proche du transfert de style, évaluant à la fois la ressemblance de l'image produite avec le style désiré et avec l'image initiale. Li et al. (2020) complètent un GAN avec un classifieur qui

doit apprendre ce qu'est une carte pour améliorer le rendu des prédictions. L'objectif de tous ces travaux reste la représentation des données et l'abstraction est laissée de côté : la fonction et l'information mise en valeur par la carte finale est exactement similaire à celle de la carte initiale. Il ne s'agit pas d'une nouvelle carte, mais de la même carte avec un style différent.

Au contraire, d'autres études travaillent à montrer que ces réseaux sont capables d'abstraction via la génération d'informations géographiques généralisées (à plus petite échelle que les données initiales). Elles ne traitent pas du tout de la représentation et se concentrent sur un type d'objet particulier, par exemple : les bâtiments (Feng et al., 2019), les traits de côte (Du et al., 2021), et les routes de montagne (Courtial et al., 2020). Ces approches n'abordent pas la représentation de l'information et mènent à la prédiction d'images assez éloignées d'une image de carte. Mais elles démontrent la capacité des réseaux d'apprentissage à sélectionner, simplifier, déplacer, déformer, abstraire de l'information géographique pour répondre à un objectif de lisibilité à une échelle cible, et par conséquent la capacité des réseaux d'apprentissage profond à construire une carte (et non lui appliquer un style). Pour étayer ces conclusions Jenny et al. (2020) apprend le positionnement d'ombrage sur une carte à partir d'un Modèle Numérique de Terrain.

Quelques expérimentations visent à la fois l'abstraction et la représentation de l'information initiale en une carte généralisée pour une échelle cible. Courtial et al. (2021b) tentent d'apprendre à généraliser l'information d'une carte topographique pour une représentation qui est également une carte topographique, mais à une échelle intermédiaire. Tandis que Kang et al. (2019) tente à partir de données OSM (donc avec un niveau de détail hétérogène) de produire une carte homogène de type Google Map. Ces premières expérimentations ont montré que la représentation des données d'apprentissage sous forme d'une rasterisation des données vecteurs symbolisées est source de limitations importantes :

— La symbolisation doit être adaptée pour éviter la superposition des symboles et permettre l'apprentissage. L'ajout de contour permet de définir les limites des objets (Touya et al., 2019) mais cela élargit les formes et ajoute de la complexité à la symbolisation.
— La symbolisation choisie doit permettre de différencier sans ambiguïté les pixels représentant des objets différents. Les couleurs trop proches sont donc à éviter au sens de distance entre les valeurs de pixels comme au sens visuel. Les attributs des objets ne sont accessibles que s'ils sont représentés dans l'image via des variables visuelles, telles que la couleur ou la taille de l'objet, cela permet de représenter par exemple la nature des bâtiments ou l'importance des routes. Mais le nombre d'attributs représentés par objet est très rapidement limité.
— Enfin le découpage de l'espace d'étude en tuile peut amener à créer des effets de bord qui perturberont l'apprentissage. Par exemple, des tuiles presque vides en bordure de zone perturbent l'apprentissage, la généralisation des objets étant influencée par leur contexte géographique.

En effet, les productions de ces deux expérimentations sont prometteuses, mais ne tirent pas entièrement partis des information initiales. Finalement, Chen et al. (2021) tente d'apprendre la traduction d'image aérienne en une carte à plusieurs échelles en tirant parti de la similarité entre les échelles pour apprendre ce qu'est une carte. Mais là encore les résultats sont limités par les données initiales qui ne permettent pas une lisibilité suffisante de l'information géographique du terrain.

2.2 Apprentissage profond et fusion d'information.

La fusion de données est une utilisation possible de l'apprentissage profond, il s'agit d'utiliser plusieurs images ou tenseurs en entrée du réseau de neurones pour une tâche unique de classification ou de segmentation d'image. En cartographie ces méthodes pourront servir pour intégrer de l'information géographique additionnelle dans l'image de carte initiale. Mais les usages actuels de ces réseaux ne sont pas orientés pour la cartographie. Par exemple, Hazırbas et al. (2016) proposent la fusion d'information de profondeur pour améliorer la segmentation de photographies de scènes en intérieur. L'architecture proposée repose sur l'encodage et la fusion progressive des informations des deux images initiales puis la création d'une segmentation. Ainsi, différents niveaux de relations entre l'image en couleurs et celle de profondeur peuvent être encodés. Guo et al. (2019) propose d'utiliser un réseau de fusion d'information comme générateur d'un réseau antagoniste génératif (GAN). Les GAN sont des réseaux d'apprentissage constitué de deux réseaux de convolutions : un générateur qui réalise à proprement parler la tâche cible et un discriminateur qui classifie les prédictions du générateur. Ces deux réseaux apprennent de façon antagoniste : le générateur à tromper le discriminateur et le discriminateur à améliorer sa classification, pour produire des images plus réalistes (qui ressemblent à une image de cible). Zhang (2021) réalise un comparatif des stratégies de fusion d'image pour apprendre à combiner des images avec différentes parties floues en une image nette. Finalement, Zhang et al. (2020) propose d'améliorer l'apprentissage de la génération de cartes à partir d'image aérienne avec de l'information supplémentaire telle que des traces GPS.

3 Expérimentations

Dans cette section, nous présentons nos expérimentations : dans un premier temps notre cas d'étude, puis les modèles d'apprentissage profond employés.

3.1 Cas d'étude

Nous nous intéressons au problème de génération de cartes topographiques d'espace urbain à une échelle intermédiaire autour du 1 :50k, à partir de donnée détaillées produite pour des cartes à 1 :25k. Notre cas d'étude inclut la combinaison et la symbolisation des objets géographiques suivants : routes, bâtiment, et rivières. La densité importante de ces objets en milieu urbain, requiert une abstraction importante de l'information initiale malgré le saut d'échelle faible : outre la simplification des formes individuelles (rectangularisation et grossissement de bâtiment, lissage des lignes), les bâtiments et le réseau routier doivent être sélectionnés et déplacés voire déformés, de sorte à conserver la structure globale et les agencements remarquables (i.e. les alignements de bâtiments, réseau en étoile, ...) , mais réduire l'encombrement et produire une carte lisible. Cette tâche a été choisie pour nos premières expérimentations, car elle demande une abstraction importante, et la carte topographique répond à une symbolisation, un objectif et une fonctionnalité clairement définis quel que soit le niveau de familiarité de l'utilisateur avec le lieu représenté, ce qui facilitera l'évaluation de la carte en sortie. Notre jeu d'exemple est produit par tuilage régulier des données dans l'aire urbaine de Saint-Jean-de-Luz. On obtient alors 3 700 paires d'images initiales et cibles, on en garde 91 pour l'évaluation et le reste sert à l'entraînement du modèle.

3.2 Modèle de génération

Les précédentes expérimentations visant à la symbolisation d'information géographique, ou à la génération de cartes ont montré le potentiel des réseaux antagonistes, c'est pourquoi nous avons concentré nos expérimentations sur ce type de réseaux. Les GANs sont des réseaux qui apprennent grâce à l'opposition de deux réseaux d'apprentissage : un générateur qui produit une image la plus proche possible du domaine cible à partir de l'image initiale et un discriminateur qui prédit la crédibilité de la prédiction. Ces deux réseaux fonctionnent en opposition : le générateur apprend à tromper le discriminateur en générant des images toujours plus réalistes tandis que le discriminateur s'entraîne à distinguer les images de la base d'exemple de celles générées. Ces réseaux sont donc particulièrement adaptés pour générer des images ressemblant à des cartes (Courtial et al., 2021b).

Comme notre but n'est pas d'étudier l'intérêt de ces réseaux pour la génération de cartes, mais plutôt de démontrer qu'une meilleure organisation de l'information initiale permet un meilleur apprentissage des tâches relatives à la cartographie nous avons uniquement testé l'entraînement du réseau GAN pix2pix (Isola et al., 2017) dans son implémentation en Pytorch avec leur paramètre et construction par défaut.

3.3 Générateur et combinaison

Le générateur du GAN a pour but d'encoder l'information initiale puis de la décoder en une prédiction, nous nous intéressons donc à la façon dont il est capable de combiner l'information géographique. Le générateur de pix2pix comme pour de nombreux GAN le générateur est en forme de U-Net Ronneberger et al. (2015). Il s'agit d'un réseau d'apprentissage qui encode et décode l'information dans une image via une série de convolutions ascendantes et descendantes, chaque convolution permet de synthétiser la valeur d'un pixel en fonction de sa valeur et celle de ses voisins. Durant la phase descendante, le nombre d'informations observées diminue, mais la quantité de valeurs pour chaque nœud augmente, durant la phase ascendante, c'est le contraire, on reconstruit une image de la taille initiale grâce à une pile d'information toujours plus petite. En connectant chaque étape de décodage au niveau d'encodage de même taille d'image, il permet de saisir différents niveaux d'information dans l'image, notamment la structure générale et le détail. Mais cette méthode d'encodage ne permet pas réellement la combinaison de différents niveaux d'information.

Au contraire le FuseNet (Hazırbas et al., 2016) est un réseau d'apprentissage conçu pour apprendre à combiner une image d'information principale avec une information additionnelle qui sert d'indication. Le cas d'étude pour lequel ce réseau a été conçu est la segmentation des images d'intérieurs à partir de photographies et d'images de profondeurs en niveau de gris qui servent d'informations additionnelles. Ce réseau permet de mieux distinguer les objets sur différents plans lors de la segmentation. L'architecture de ce réseau est faite de deux chemins d'encodage parallèles, un pour les images en niveau de gris et l'autre pour les images couleurs, à chaque étape des informations de profondeurs sont réintégrées et fusionnées à l'image en couleurs encodée, ensuite l'image en couleurs est décodée en une segmentation. Il peut être adapté à la combinaison d'information géographique où la forme et la localisation des entités géographiques sont l'information principale, mais non suffisante à la génération de cartes.

On propose donc de comparer une architecture de GAN dont le générateur est un U-Net qui encode directement une pile d'images comme pour Pix2pix (Isola et al., 2017) et une architec-

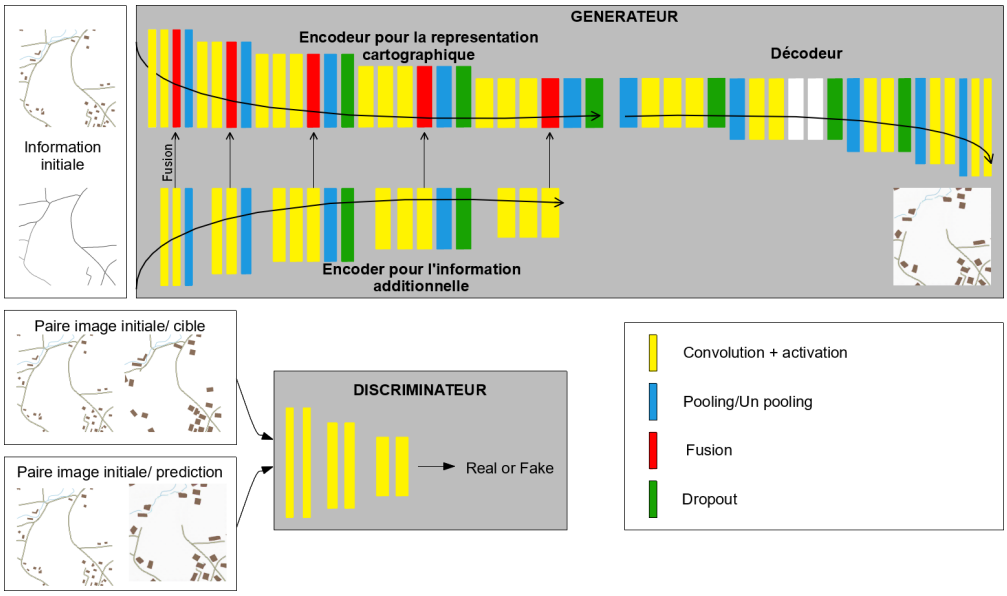


FIG. 1 – *Architecture du GAN proposé opposant un Fusenet à un discriminateur.*

ture de GAN dont le générateur est un FuseNet qui fusionne progressivement les informations additionnelles avec l'image de carte initiale puis reconstruit une carte généralisée. Cette architecture est décrite dans la Figure 1.

4 Représentation de données vectorielles avec un tenseur

Nous proposons de comparer la représentation des données initiales sous forme d'extrait de cartes avec une autre représentation initiale de l'information qui limiterait les problèmes de coalescence entre objets de types différents. (Dans tous les cas, l'image cible est un extrait de la carte cible.) Cette représentation consiste à utiliser une pile de masques pour représenter la base de données initiale, chaque masque de la pile représentant un type d'objet. Dans sa version simple où on représente uniquement l'emprise des objets sans informations supplémentaires la pile d'images binaire route, hydrographie, bâtiments, serait alors suffisante pour représenter une ville comme dans l'exemple Figure 2. L'apprentissage des relations entre objets de types différents (Par exemple : la rivière longe une route.) serait facilité, car les deux éléments sont toujours représentés même en cas de colocalisation, et les relations entre objets ou groupes d'objets du même type seraient également plus lisibles, car non parasités par d'autres symboles en superposition. Pour aller plus loin, on peut sur le même principe proposer des images en niveau de gris pour représenter des attributs ou informations additionnelles (gradation ou catégorie) dans chaque image de la pile (et donc thème de la carte).

La Figure 3 présente un extrait de notre jeu test et les prédictions associées à chacune des approches.

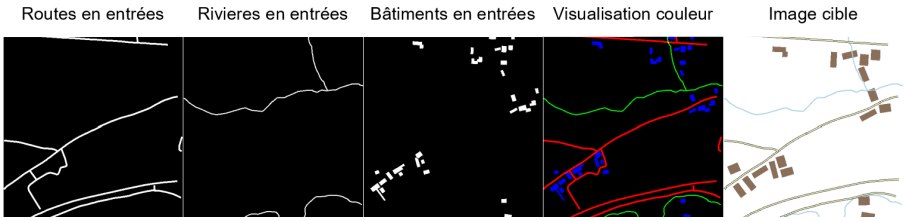


FIG. 2 – *Extrait des images par thème de notre jeu de donnée.*

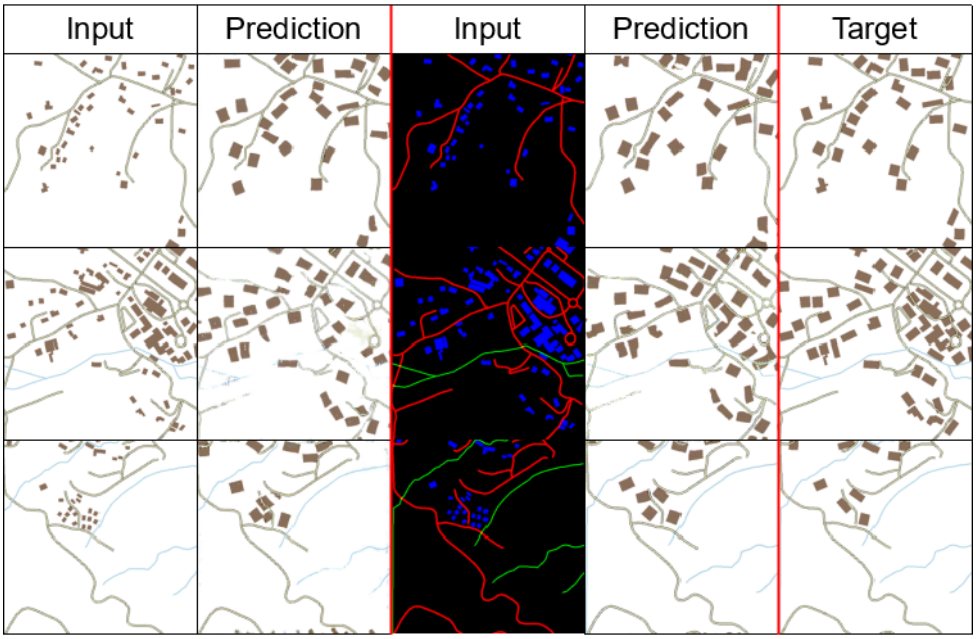


FIG. 3 – *Comparaison des images de cartes générées par le GAN en fonction de la représentation initiale de l'information.*

On observe que pour la plupart des situations, comme la première ligne de la Figure 3, le mode de représentation n'impacte pas la qualité de la prédiction. Mais pour les cas où l'information est dense et la superposition entre symboles est importante dans l'image initiale, comme dans le cas décrit par la deuxième ligne de la Figure 3 la représentation par thème limite les effacements d'informations liés à ces superpositions. La superposition des symboles dans la prédiction n'est cependant pas totalement évitée. Enfin, dans le cas où l'information serait dense, mais où il n'y a pas de superposition dans l'image initiale, la prédiction basée sur la représentation par thème comporte moins de superpositions liées au grossissement des bâtiments, mais elle n'est pas totalement évitée.

5 Ajouter le contexte autour de l'imagette

On se concentre ici sur les informations additionnelles pouvant être représentées sous forme d'un tenseur de même dimension que la représentation symbolisée de la donnée géographique, ainsi le lien avec les données initiales est aisé : le pixel de coordonnée (i,j) de l'image symbolisée représente et décrit la même portion de terrain que le pixel de même coordonnée (i,j) du tenseur additionnel. Ces informations additionnelles peuvent être de 3 types principaux décrits et illustrés dans la table 1.

Type	Description	Exemple
Attributaire	Renseigne sur les valeurs d'un attribut pour un type d'objet	La fonction d'un bâtiment.
Contextuel	Renseigne sur le contexte d'un objet géographique, permet d'avoir une information sur ce qu'il se passe en dehors de la tuile	Densité dans un îlot urbain.
Relationnel	Renseigne sur les relations spatiales implicites entre les objets dans l'espace représenté	Alignement de bâtiments.

TAB. 1 – *Type et exemple d'information additionnelle.*

Chacune des informations dans la Table 1 est représentée par un tenseur couvrant la même emprise que la représentation initiale en Figure 4. Ainsi, les pixels des images d'informations additionnelles peuvent prendre la forme d'une image en niveau de gris, une segmentation en n classes, ou d'une image binaire, mais on pourrait imaginer des cas plus complexes où l'information serait sous forme d'une image en couleur.

Pour nos expérimentations nous nous sommes concentrés sur le cas d'étude d'une information sur les routes en encodant la probabilité de chaque portion de route d'être gardée après généralisation. L'information est issue d'un premier modèle de prédiction sur les arcs du réseau routier avec un réseau d'apprentissage par convolution sur les graphes (Courtial et al., 2021a). Cette information est représentée sous forme d'une image en niveau de gris, les pixels de route ont pour valeur la probabilité de ne pas être conservé de la route à laquelle ils appartiennent, les autre pixels valent 1. Nous espérons que la combinaison de l'information additionnelle choisie améliore l'apprentissage de l'abstraction de ce thème. C'est-à-dire que les arcs peu importants

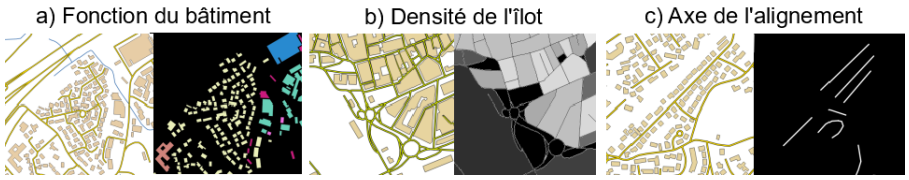


FIG. 4 – *Représentation des exemples d'information additionnelle. a) Fonction du bâtiment, segmentation en n classes. b) Densité des îlots, niveau de gris. c) Axe d'alignement, segmentation binaire.*

FIG. 5 – *Extrait des images symbolisées et additionnelles de notre jeu de données.*

et/ou causant des problèmes de lisibilité ne soient pas représentés, sans que cela altère la forme et la structure générale du réseau routier. La Figure 5 est un exemple de cette information pour quelques tuiles de notre zone d'étude.

Nous proposons de comparer l'empilement dans un unique tenseur de cette information avec l'image symbolisée de l'information initiale, avec l'utilisation de mécanisme de fusion de cette information additionnelle. Ces deux stratégies se traduisent par l'utilisation de deux générateurs différents dans une architecture GAN 3.3. D'une part, un générateur en forme de U-Net qui prend en entrée le tenseur d'empilement, et d'autre part, un générateur de type FuseNet qui fusionne deux tenseurs. Dans les deux cas, le générateur est opposé à un réseau de convolutions (CNN) qui sert de discriminateur. Cela permet de générer des images qui ressemblent à des cartes généralisées. La Figure 6 présente les images générées par chacun de ces GAN pour un extrait de notre jeu test.

On observe dans un premier temps que les images sont bruitées et les formes de bâtiments manquent de régularité. On peut expliquer cette diminution de qualité par l'augmentation du niveau d'abstraction entre image initiale et finale qui complexifie l'apprentissage par rapport à celle testée dans (Courtial et al., 2021b). On note également que les deux réseaux apprennent à effacer des routes pour éviter la superposition entre symboles de routes et de bâtiments. Mais le réseau procédant par empilement ne semble pas tirer profit de la carte de probabilités, en effet, il efface des portions d'arcs et non des arcs complets créant des déconnexions et les arcs sélectionnés ne permettent pas de préserver la structure générale du réseau routier. De plus, cette méthode semble aussi dégrader la représentation des rivières. Au contraire, le réseau

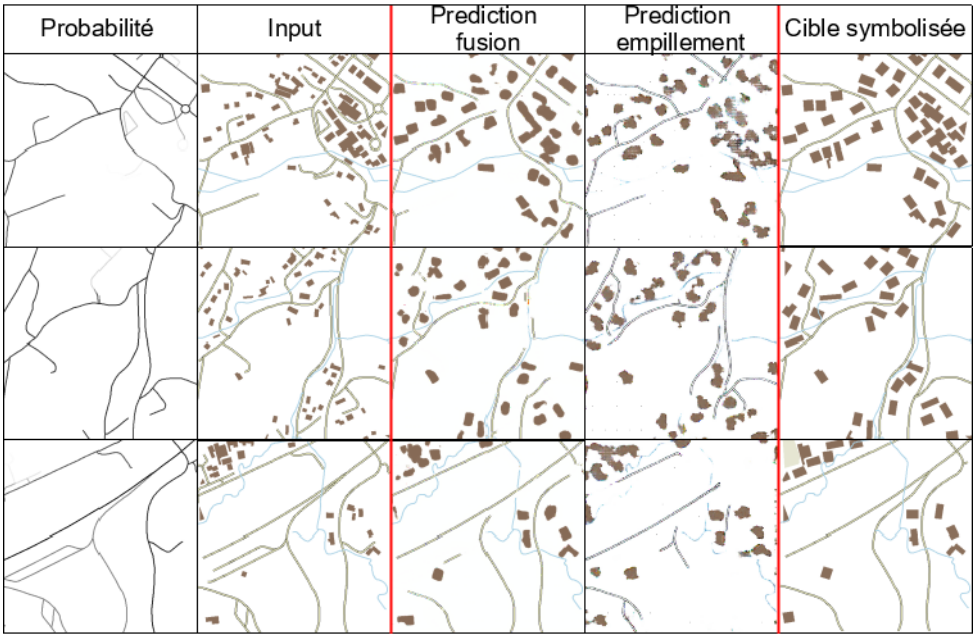


FIG. 6 – *Comparaison des images de cartes générées par un GAN à partir de l'empilement ou de la fusion d'une image de carte et de l'information additionnelle de probabilité de sélection.*

ayant un générateur en forme de FuseNet ne sur-sélectionne pas le réseau routier et préserve davantage la structure générale. Par exemple, pour la première situation générée la structure principale en Y du réseau routier est préservée par fusion (colonne 3) mais pas par empilement (colonne 4)

6 Conclusion

En conclusion, la représentation et la combinaison de ces informations figurent parmi les défis majeurs de la génération de cartes. Nos expérimentations permettent de décrire et de comparer des méthodes pour ce problème. Nous avons notamment montré qu'une représentation initiale par thème plutôt que symbolisée permet une meilleure description des relations spatiales de l'espace à cartographier. De plus, l'utilisation d'informations additionnelles est essentielle à l'amélioration des réseaux de générations de cartes, et cela, d'autant plus qu'ils demandent un niveau d'abstraction important, et que la fusion d'information est dans notre cas d'étude plus efficace que l'empilement pour intégrer de l'information géographique additionnelle. À l'issue de ce travail, nous notons qu'en plus de la recherche en cours dans la littérature sur des architectures mieux adaptées à l'information géographique, il est utile d'étudier l'enrichissement de l'information initiale fournit aux réseaux de neurones et d'élaborer des architectures de fusion d'informations plus performantes (GAN de fusion avec données non appariées, ou semi-appariées, permettant la fusion de plus d'informations additionnelles,

...). Enfin, nous pensons que la possibilité d'utiliser une information additionnelle sous une autre forme (image de taille différente, graphe, table de valeur, ...) figure parmi les perspectives importantes de ce travail, puisque la représentation sous forme d'image peut limiter la transmission de l'information géographique nécessaire à la généralisation cartographique.

Références

Chen, X., S. Chen, T. Xu, B. Yin, J. Peng, X. Mei, et H. Li (2020). SMAPGAN : Generative Adversarial Network-Based Semisupervised Styled Map Tile Generation Method. *IEEE Transactions on Geoscience and Remote Sensing*, 1–19. Conference Name : IEEE Transactions on Geoscience and Remote Sensing.

Chen, X., B. Yin, S. Chen, H. Li, et T. Xu (2021). Generating Multi-scale Maps from Remote Sensing Images via Series Generative Adversarial Networks. *arXiv :2103.16909 [cs, eess]*.

Courtial, A., A. El Ayedi, G. Touya, et X. Zhang (2020). Exploring the Potential of Deep Learning Segmentation for Mountain Roads Generalisation. *ISPRS International Journal of Geo-Information 9*(5), 338. Number : 5 Publisher : Multidisciplinary Digital Publishing Institute.

Courtial, A., G. Touya, et X. Zhang (2021a). Can Graph Convolution Networks Learn Spatial Relations ? *Abstracts of the ICA 3*, 1–2. DOI : 10.5194/ica-abs-3-60-2021.

Courtial, A., G. Touya, et X. Zhang (2021b). Generative adversarial networks to generalise urban areas in topographic maps. In *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Volume XLIII-B4-2021, pp. 15–22. Copernicus GmbH. ISSN : 1682-1750.

Du, J., F. Wu, R. Xing, X. Gong, et L. Yu (2021). Segmentation and sampling method for complex polyline generalization based on a generative adversarial network. *Geocarto International 0*(0), 1–23. Publisher : Taylor & Francis _eprint : https ://doi.org/10.1080/10106049.2021.1878288.

Feng, Y., F. Thiemann, et M. Sester (2019). Learning cartographic building generalization with deep convolutional neural networks. *International Journal of Geo-Information*.

Fu, H., M. Gong, C. Wang, K. Batmanghelich, K. Zhang, et D. Tao (2019). Geometry-Consistent Generative Adversarial Networks for One-Sided Unsupervised Domain Mapping. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, pp. 2422–2431. IEEE.

Ganguli, S., P. Garzon, et N. Glaser (2019). GeoGAN : A Conditional GAN with Reconstruction and Style Loss to Generate Standard Layer of Maps from Satellite Images. *arXiv :1902.05611 [cs]*.

Guo, X., R. Nie, J. Cao, D. Zhou, L. Mei, et K. He (2019). FuseGAN : Learning to Fuse Multi-Focus Image via Conditional Generative Adversarial Network. *IEEE Transactions on Multimedia 21*(8), 1982–1996. Conference Name : IEEE Transactions on Multimedia.

Hazırbas, C., L. Ma, C. Domokos, et D. Cremers (2016). FuseNet : Incorporating Depth into Semantic Segmentation via Fusion-Based CNN Architecture.

Isola, P., J.-Y. Zhu, T. Zhou, et A. A. Efros (2017). Image-to-image Translation with Conditional Adversarial Networks.

Jenny, B., M. Heitzler, D. Singh, M. Farmakis-Serebryakova, J. C. Liu, et L. Hurni (2020). Cartographic Relief Shading with Neural Networks. *arXiv :2010.01256 [cs]*.

Kang, Y., S. Gao, et R. Roth (2019). Transferring Multiscale Map Style Using Generative Adversarial Network.

Li, J., Z. Chen, X. Zhao, et L. Shao (2020). MapGAN : An Intelligent Generation Model for Network Tile Maps. *Sensors 20*(11), 3119. Number : 11 Publisher : Multidisciplinary Digital Publishing Institute.

Ronneberger, O., P. Fischer, et T. Brox (2015). U-Net : Convolutional Networks for Biomedical Image Segmentation. Volume abs/1505.04597, pp. 234–241.

Sester, M. (2005). Optimization approaches for generalization and data abstraction. *International Journal of Geographical Information Science 19*(8), 871–897.

Touya, G., X. Zhang, et I. Lokhat (2019). Is deep learning the new agent for map generalization ? *International Journal of Cartography 5*(2-3), 142–157. Publisher : Taylor & Francis _eprint : https ://doi.org/10.1080/23729333.2019.1613071.

Weibel, R., S. Keller, et T. Reichenbacher (1995). Overcoming the knowledge acquisition bottleneck in map generalization : The role of interactive systems and computational intelligence. In A. U. Frank et W. Kuhn (Eds.), *Spatial Information Theory A Theoretical Basis for GIS*, Volume 988 of *Lecture Notes in Computer Science*, pp. 139–156. Berlin Heidelberg : Springer.

Zhang, X. (2021). Deep Learning-based Multi-focus Image Fusion : A Survey and A Comparative Study. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–1. Conference Name : IEEE Transactions on Pattern Analysis and Machine Intelligence.

Zhang, Y., Y. Yin, R. Zimmermann, G. Wang, J. Varadarajan, et S.-K. Ng (2020). An Enhanced GAN Model for Automatic Satellite-to-Map Image Conversion. *IEEE Access 8*, 176704–176716. Conference Name : IEEE Access.

Zhu, J.-Y., T. Park, P. Isola, et A. A. Efros (2017). Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, Venice, pp. 2242–2251. IEEE.

Summary

Deep learning allows the generation of maps from images. However, images only convey a limited view of the vectorgeographic information used to creat map. In this article, we explore the methods to combine and represent cartographic information as tensors to improve map generation, and especially generalised map generation using deep learning. First, we propose a new information representation that separate geographic themes (roads, rivers or buildings,...) to avoid some limitations of usual image representation. Then, we study how to combine additional information with our input image using a FuseNet architecture. Our first results permit us to show the interest of information combination compared to simple map images.

Recherche de motifs fréquents dans un multi-graphe étiqueté et orienté. Application à des graphes spatio-temporels synthétiques et environnementaux.

Aurélie Leborgne*, Ezriel Steinberg*, Laurine Lafontaine*,**,
Florence Le Ber*, Antoine Vacavant***

*Université de Strasbourg, CNRS, ENGEES, ICube UMR 7357, F-67000 Strasbourg
{aurelie.leborgne,florence.le-ber}@unistra.fr
** Université Gustave Eiffel, ESIPE, F-77400 Champs-sur-Marne
*** Université Clermont Auvergne, CNRS, SIGMA Clermont, Institut Pascal,
F-63000, Clermont-Ferrand, France
antoine.vacavant@uca.fr

Résumé. Cet article présente un algorithme, LD-MuGraM, inspiré de l'algorithme MuGraM et conçu pour extraire des sous-graphes fréquents dans un multi-graphe étiqueté orienté. Nous nous intéressons en particulier aux graphes spatio-temporels, qui permettent de représenter des objets spatiaux évoluant dans le temps. L'algorithme a été testé à la fois sur des données synthétiques, issues d'un générateur de graphes spatio-temporels incluant des motifs, et sur des données temporelles d'occupation des sols issues de bases de données nationales. Les résultats de ces expérimentations sont présentés et discutés.

1 Introduction

Les progrès technologiques récents entraînent une production massive de données complexes dans de nombreux domaines. Cette complexité est liée en particulier à l'existence de relations multiples entre les objets observés. Par exemple, dans le domaine social, deux personnes peuvent être en lien *via* plusieurs réseaux sociaux (Facebook, LinkedIn, Twitter, *etc*). Une manière intuitive de modéliser ces données est d'utiliser un multi-graphe dans lesquel les éléments (dans l'exemple, les personnes) sont représentés par des nœuds, et chaque relation/interaction entre deux entités est représentée par une arête. La figure 1 (temps 1) illustre cette représentation : Tom, Noé, Lise et Elsa sont représentés par des nœuds reliés par trois types d'arêtes différents représentant leurs relations dans les trois réseaux LinkedIn, Twitter et Facebook. Si on s'intéresse maintenant à l'évolution de ces relations, on voit sur la figure 1 (temps 2) qu'Elsa a quitté Twitter et n'entretient donc plus de relation avec Tom, et n'est plus reliée que par deux réseaux avec Lise.

Les multi-graphes peuvent donc représenter des données incluant une dimension temporelles, ici des réseaux sociaux, ou dans d'autres domaines des objets spatiaux : par exemple Del Mondo et al. (2010) utilisent un graphe spatio-temporel pour modéliser la propagation de ronces ; Leborgne et al. (2019) utilisent ce même modèle pour représenter l'évolution des

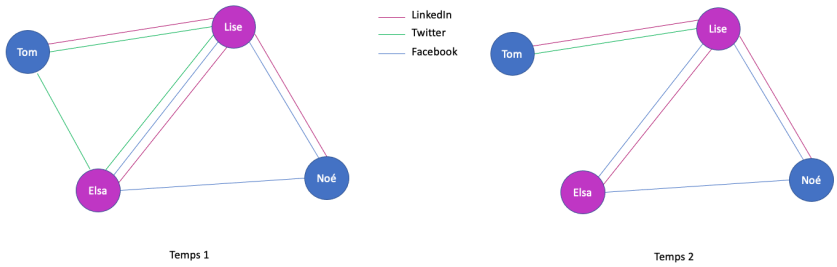


FIG. 1: Exemple de multi-graphe représentant des relations dans des réseaux sociaux

cultures de parcelles agricoles. Ce modèle de graphe spatio-temporel, qui est un multi-graphe étiqueté orienté, permet de représenter à la fois des informations spatiales entre les éléments à un instant donné mais aussi des informations entre temporalités successives comme l'évolution spatiale des éléments ou la transmission de l'identité. Il peut donc y avoir plusieurs arcs entre deux éléments.

Dans des applications concrètes, les graphes spatio-temporels peuvent atteindre plusieurs dizaine de milliers de nœuds, et sont donc difficilement interprétables. Il est donc crucial de disposer de méthodes pour en extraire des informations pertinentes. L'approche choisie dans ce travail est de les analyser de manière non supervisée, sans *a priori*. Il n'existe pas à notre connaissance d'outils pour fouiller ce type de graphe, c'est pourquoi nous avons développé une méthode de recherche de motifs fréquents dans un multi-graphe étiqueté orienté ; notre approche s'appuie sur la méthode, nommée MuGraM, proposée par Ingalalli et al. (2018). Nous l'avons adaptée et avons étudié son efficacité. Nous utilisons pour cela des graphes spatio-temporels synthétiques générés par un outil *ad hoc* (Leborgne et al., 2020) ainsi que des données temporelles sur l'occupation du sol issues de bases nationales.

L'article est organisé comme suit. Dans la partie 2, nous présentons les définitions nécessaires à la compréhension de la méthode et nous discutons des travaux menés sur des problématiques connexes. Dans la partie 3, nous rappelons les prinicpes de MuGrAM et détaillons les adaptations faites pour extraire les motifs fréquents dans un multi-graphe étiqueté orienté. Puis, la partie 4 présente quelques expérimentations menées pour évaluer la complexité et la fiabilité de l'algorithme développé. La partie 5 dresse quelques conclusions et perspectives.

2 Définitions et état de l'art

2.1 Définitions

Un graphe orienté est constitué d'un ensemble de sommets et d'un ensemble d'arcs. Chaque arc relie un couple de sommets. Dans la suite on considère que les arcs sont typés et les sommets étiquetés.

Définition 1 *Un **multi-arc** est un ensemble d'arcs joignant le même couple de sommets où chaque arc est d'un type différent.*

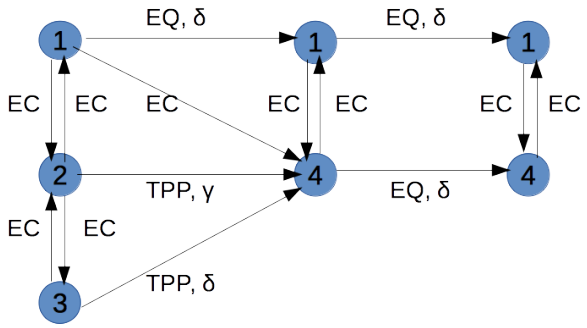


FIG. 2: Un exemple de graphe spatio-temporel : au temps 1, l'objet 2 est voisin (EC, relation symétrique) des objets 1 et 3 : au temps 2, les objets 2 et 3 fusionnent dans un objet 4 : ils sont parties propres tangeantielles (TPP) de cet objet, et ont une filiation γ (dérivation) ou δ (continuation) ; l'objet 1 reste identique à lui-même (relation EQ, continuation δ) ; au temps 3 les objets ne changent pas

Définition 2 *Un **multi-graphe orienté aux sommets étiquetés** est par abus de langage appelé multi-graphe étiqueté orienté dans ce papier. Il est défini comme un tuple* (V, E, f, g, L, T) *où :*

- V *est l'ensemble des sommets*
- $E \subseteq V \times V$ *est l'ensemble des multi-arcs*
- L *est l'ensemble des étiquettes des sommets*
- T *est l'ensemble des types d'arcs*
- f *est une application de l'ensemble des sommets* V *dans* L*. Pour tout sommet* v*,* $f(v)$ *est appelée étiquette de* v*.*
- $g : E \rightarrow 2^T$ *est une fonction d'étiquetage, qui assigne un sous-ensemble de* T *à chaque multi-arc de* E*.*

Nous nous intéressons à des multi-graphes orientés particuliers, les graphes spatio-temporels, tels qu'introduits par Del Mondo et al. (2010). Ces graphes permettent de représenter l'évolution d'objets spatiaux. L'ensemble T des types d'arcs peut être décrit ainsi : $T = \mathcal{B} \cup \mathcal{F}$ où $\mathcal{B}$ est l'ensemble de base des relations RCC8 (relations spatiales qualitatives entre deux régions) (Randell et al., 1992) et $\mathcal{F}$ est un ensemble de relations de filiation (continuation, dérivation, *etc.*) (Del Mondo et al., 2010). Un exemple illustratif est présenté en figure 2.

Définition 3 $G_1 = (V_1, E_1, f, g_1, L_1, T_1)$ *est un **sous-graphe** de* $G = (V, E, f, g, L, T)$ *si* $V_1 \subseteq V$ *et* $E_1 \subseteq E \cap (V_1 \times V_1)$ *de sorte que si deux sommets* u*,* v *sont reliés par au moins un multi-arc dans* G*, alors* E_1 *contient au moins l'un des couples* (u, v)*,* (v, u)*. De plus* $L_1 \subseteq L$*,* $T_1 \subseteq T$ *et* $\forall (u, v) \in E_1, g_1(u, v) \subseteq g(u, v)$*.*

Définition 4 *Deux sous-graphes* G_1 *et* G_2 *sont dits **isomorphes** s'il existe une bijection* $h : V_1 \rightarrow V_2$ *telle que* $(u, v) \in E_1$ *si et seulement si* $(h(u), h(v)) \in E_2$ *et les étiquettes des sommets et des multi-arcs sont préservées.*

Définition 5 *Un **motif** de G est un sous-graphe de G. Le **support** d'un motif est généralement défini comme le nombre de sous-graphes de G isomorphes à ce motif. Un motif est dit **fréquent** si son support est supérieur à un seuil donné $\sigma > 0$.*

2.2 Recherche de motifs fréquents dans un graphe

Il existe de nombreuses approches pour l'extraction de motifs dans les graphes. La recherche de structures dans des graphes est une question ancienne, avec de nombreuses applications, par exemple l'étude de structures chimiques (Raymond et al., 2002), mais aussi l'étude des réseaux sociaux, où les graphes sont de tailles importantes et évoluent dans le temps (Falkowski et al., 2008). Différentes approches ont été proposées pour extraire des sous-graphes fréquents, soit à partir d'un seul grand graphe, soit à partir d'un ensemble de graphes. Un des premiers algorithmes généraux, AGM (Inokuchi et al., 2000) est fondé sur une extension de l'algorithme Apriori de recherche de motifs fréquents (Agrawal et Srikant, 1994). Il permet de trouver les sous-graphes fréquents dans un ensemble de graphes. Différents algorithmes inspirés de cette approche ont été proposés, par exemple (Elseidy et al., 2014) qui permet la recherche de sous-graphes fréquents dans un seul grand graphe. Cette approche a été étendue aux graphes pondérés (Le et al., 2020). Plus proches de notre problématique, Flouvat et al. (2020) ont développé une méthode de recherche de motifs fréquents sur des phénomènes spatiaux-temporels représentés par des graphes orientés acycliques. Dans ces graphes, seuls les sommets sont étiquetés, et les arcs ne représentent que des voisinages temporels.

Des approches neuronales spécifiques aux graphes ont été développées, initialement pour réaliser la projection de graphes dans un espace euclidien, où peuvent s'appliquer les méthodes existantes (Scarselli et al., 2009). Toutefois, à notre connaissance, une seule méthode de recherche de motifs fréquents basée sur un apprentissage profond, SPMiner (Ying et al., 2020), a été publiée. Cette méthode comprend deux étapes : *(i)* plongement des sous-graphes candidats (voisinage de chaque nœud) dans un espace ordonné en 2 dimensions, appelé E ; *(ii)* génération des motifs fréquents par l'ajout itératif de nœuds grâce à une marche monotone dans E. Il est à noter que cette méthode n'est applicable que sur un graphe simple, sans étiquette et ne permet d'extraire que la structure de motifs.

3 Méthode

3.1 Principes de l'algorithme MuGraM (Ingalalli et al., 2018)

L'algorithme que nous avons développé s'appuie sur MuGraM, décrit par Ingalalli et al. (2018). MuGraM effectue la recherche de sous-graphes dans un multi-graphe non orienté, en exploitant le caractère monotone décroissant (dit antimonotonie) des supports vis-à-vis de l'inclusion des motifs (les graphes contenant un sous-graphe non fréquent sont non-fréquents).

La définition générale de support de permet pas de vérifier ce caractère, du fait des recouvrements possibles entre sous-graphes isomorphes à un motif. C'est pourquoi MuGraM utilise le support obtenu en calculant le *Minimum Node Image* (MNI), défini ainsi (Bringmann et Nijssen, 2008) : soit un motif $M = (V_M, E_M)$ et $\{g_1, g_2, ..., g_n\}$ l'ensemble de ses projections (graphes isomorphes) dans le graphe général ; on note h_i la fonction as-

sociant aux sommets de V_M les sommets correspondants de g_i. Alors le support MNI est $\Delta(M) = min_{v \in V_M} |\{h_i(v), i = 1...n\}|$.

MuGraM effectue un parcours en profondeur, à partir de « graines », *i.e.* des sous-graphes de deux nœuds dont le support MNI est supérieur à un seuil donné. Chaque graine est décrite comme une liste des types d'arêtes reliant les deux nœuds. Chaque graine est ensuite successivement augmentée des autres graines, jusqu'à obtenir un sous-graphe non fréquent. Différentes optimisations permettent de limiter l'exploration de l'espace et la vérification du support des sous-graphes construits.

La recherche de motifs dans MuGraM a une complexité temporelle maximale de l'ordre de $O(V^u.2^{V^2}.V!)$, où u est la taille du motif, V la taille du graphe. De nombreuses optimisations permettent de réduire cette complexité (Ingalalli et al., 2018).

3.2 LD-MuGraM pour les multi-graphes étiquetés orientés

L'algorithme LD-MuGraM s'appuie sur les principes généraux de MuGraM, mais des modifications ont été apportées pour tenir compte des spécificités des graphes spatio-temporels :

— ajout de l'étiquette dans l'indexation des nœuds : pour rendre la recherche de sous-graphes isomorphes plus rapide, MuGraM utilise une méthode d'indexation (Ingalalli et al., 2016). LD-MuGraM prend en compte l'étiquette des nœuds.
— modification des structures de données : dans LD-MuGraM, une graine est définie comme un tuple comportant : *(i)* l'étiquette du premier nœud, *(ii)* l'étiquette du second nœud, *(iii)* l'énumération des types des arcs allant du premier nœud au second nœud puis *(iv)* l'énumération des types des arcs allant du second nœud au premier nœud.

De plus, quelques optimisations ont été mises en œuvre. Tout d'abord, notons que l'attribution d'une étiquette à chaque nœud rend les motifs fréquents à rechercher plus spécifiques. Par conséquent, ils sont moins nombreux et la recherche est donc plus rapide.

Cependant, manipuler des multi-arcs à la place des multi-arêtes augmente exponentiellement l'espace de recherche. Par exemple, lorsqu'il existe des relations symétriques, comme c'est le cas de certaines relations spatiales dans les graphes spatio-temporels (voir les arcs EC sur la figure 2), l'information est dédoublée : une relation symétrique est représentée par deux nœuds u et v reliés par deux multi-arcs (u, v) et (v, u) de même type. Trois graines peuvent alors être générées contre une seule pour un multi-graphe non orienté, comme le montre la figure 3. Pour limiter l'explosion du nombre de graines, on ne garde que la graine maximale où les deux multi-arcs sont conservés.

Par ailleurs, pour réduire le temps de recherche des motifs, on peut utiliser les spécificités des données et les objectifs de la recherche. La procédure d'indexation permet de préciser les caractéristiques attendues des motifs, par exemple le degré des nœuds, ou la valeur des étiquettes. Lors de la recherche des motifs, les nœuds n'ayant pas ces caractéristiques ne seront pas visités. Il faut néanmoins limiter le nombre de caractéristiques ainsi indexées, car si cela diminue le nombre de tentatives d'appariemment, cela augmente aussi le temps de recherche de ces appariemments.

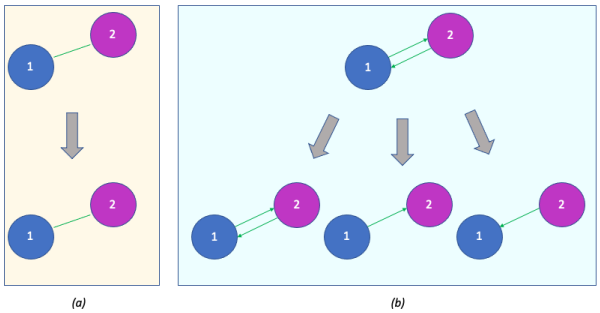


FIG. 3: Graines pouvant être générées (a) à partir d'une multi-arête entre deux nœuds, (b) à partir de deux multi-arcs entre deux nœuds

4 Expérimentations

Nous n'avons pas pu comparer LD-MuGraM à d'autres algorithmes s'appliquant à des muti-graphes orientés étiquettés, car il n'existe pas de tels algorithmes à notre connaissance.

4.1 Données synthétiques

Pour tester LD-MuGraM, nous avons utilisé un générateur de graphes spatio-temporels (Leborgne et al., 2020, 2021), qui permet de comparer les motifs trouvés à une « vérité terrain ». Ce générateur a été conçu pour construire des graphes spatio-temporels (multi-graphes étiquetés orientés particuliers) incluant des motifs dont on peut contrôler la taille (en nombre de sommets et d'arcs), l'étiquetage et le nombre de répétitions. Différents paramètres (décrits au tableau 1) permettent aussi de contrôler les caractéristiques moyennes des graphes générés.

Paramètre	**Caractéristique contrôlée**
λ_n	nombre moyen de nœuds dans le graphe
λ_r	nombre moyen de nœuds par temporalité
Λ_e	nombre moyen d'arcs spatiaux / spatio-temporels / de filiation par nœud
$labels_n$	Etiquettes pour les nœuds
$labels_e$	Etiquettes pour les types de relation
p	Pourcentage de nœuds appartenant à des motifs

TAB. 1: Paramètres pour la génération de graphes spatio-temporels ; ces paramètres s'appliquent aussi à la génération des motifs (sauf le dernier)

Les expérimentations ont été menées avec les valeurs de paramètres suivantes :
— Taille de graphe : $\lambda_n = 1000$
— Proportion de nœuds des motifs : $p \in [20; 75]$
— Taille des motifs : 5 nœuds
— Nombre d'arcs par nœud dans le graphe [5,5,2], dans le motif par défaut [5,5,2] ou variable ;

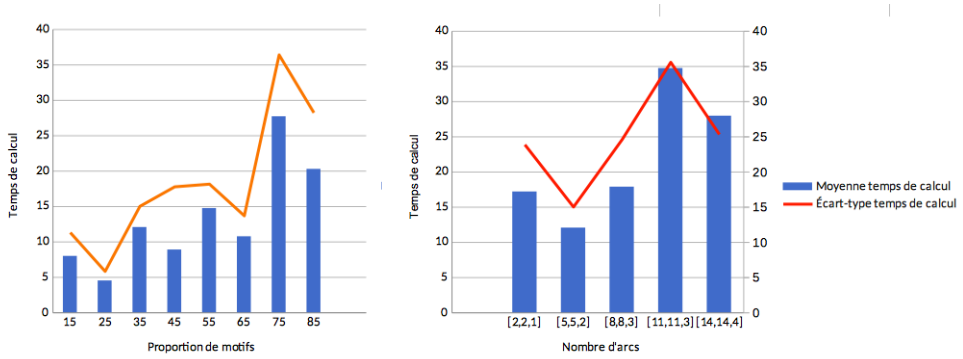


FIG. 4: Temps de recherche (mn) des motifs en fonction de la proportion de motifs (motif de taille 5), à gauche, et du nombre d'arcs par nœud dans les motifs, à droite

— Nombre de répétitions du motif (fixé pour contrôler le nombre de motifs distincts) : de 30 à 170 ;
— Etiquettes : 8 types pour les arcs spatiaux, 8 pour les arcs spatio-temporels, 2 pour les arcs de filiation ; 8 valeurs pour les nœuds.

Pour chaque combinaison de valeurs, 10 graphes ont été générés. Sont alors mesurés, le temps de calcul pour extraire les motifs ainsi que le nombre et l'exactitude des motifs trouvés (taux de motifs exacts trouvés, et motifs approximatifs à 20% de nœuds près). Les expérimentations ont été faites sur ordinateur Ubuntu 20.04.2 version LTS avec le CPU suivant : Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz.

Les temps de calcul (moyenne et écart-type) sont présentés en figure 4. On observe une augmentation des temps avec la densité de motifs, jusqu'à une certaine saturation (75%), où le temps décroît. Concernant le nombre d'arcs, on observe le même phénomène, avec un temps de calcul réduit quand le nombre d'arcs par nœud dans les motifs coïncide avec ce nombre dans le graphe général (valeur par défaut, [5,5,2]). Remarquons que les écarts-types sont élevés, il y a une variabilité importante des temps de recherche, liée au caractère aléatoire des graphes générés et de l'insertion des motifs.

Concernant l'exactitude des motifs retrouvés, les résultats sont différents dans les deux expérimentations : dans la première, tous les motifs insérés ont été retrouvés à l'identique. Pour la seconde, l'algorithme trouve le plus souvent des motifs plus grands que ceux recherchés (avec une distance d'édition moyenne supérieure à 4). Ceci est à relier au fait que le nombre d'arcs dans les motifs est différent du nombre attendu dans le graphe ([5,5,2] par nœud).

Notons que certains graphes générés n'ont pas permis à l'algorithme d'aboutir en temps raisonnable et que, de manière générale, certaines configurations (motifs de taille supérieure à à 10, graphes de taille supérieure à 10 000) conduisent à un temps de calcul très grand qui ne permet d'utiliser l'algorithme LD-MuGraM sur ces graphes aléatoires.

4.2 Données réelles

Nous utilisons des données des années 2015 à 2019 issues du registre parcellaire graphique [1], registre dans lequel les agriculteurs sont tenus de déclarer annuellement les cultures mises en place sur leurs parcelles. A partir de ces données, nous construisons des graphes spatio-temporels où une parcelle agricole, une année donnée, est modélisée par un nœud ayant pour étiquette la culture principale cette année là. Les arcs spatiaux ont comme type la relation voisin (symétrique). Les arcs spatio-temporels ont comme type la relation EQ (égalité). Nous n'avons pas utilisé les relations de filiation, qui sont redondantes avec les relations spatio-temporelles dans ces données.

Dans un premier temps, nous avons construit un graphe à partir d'une zone d'environ 2 000 parcelles située en Eure et Loir. Les caractéristiques du graphe sont les suivantes : nombre de nœuds : 10.866 (20 étiquettes : 1 étiquette couvre 21% des nœuds, 6 en couvrent 46%, les 7 suivantes en couvrent 24%, les autres en représentent chacune moins de 1%), nombre d'arcs spatiaux : 89.246 (une seule étiquette), nombre d'arcs temporels : 3.601 (une seule étiquette). On voit que la particularité de ces données est le grand nombre d'étiquettes des nœuds et leur variabilité.

Les expérimentations ont été réalisées sur un ordinateur Ubuntu 20.04.3 LTS, Intel(R) Xeon(R) W-2235 CPU @ 3.80GHz, 32 Go de RAM. La figure 5 (gauche) présente les temps de calcul pour des supports variant de 800 à 2000. Sur cette figure, on observe un coude pour une valeur de support 900. Sous cette valeur, avec nos moyens de calcul, nous n'avons pas réussi à obtenir de résultats. En revanche, lorsque le support dépasse 900, le temps de calcul est inférieur à dix minutes. Sur la même figure à droite est représenté le nombre de motifs extraits en fonction du support. Cette expérience met en évidence que le nombre de motifs trouvés, très faible pour les grandes valeurs du support, augmente quand le support descend en dessous de 1200, puis très rapidement quand le support est inférieur à 900. Ce résultat est cohérent avec la forme de la courbe de gauche.

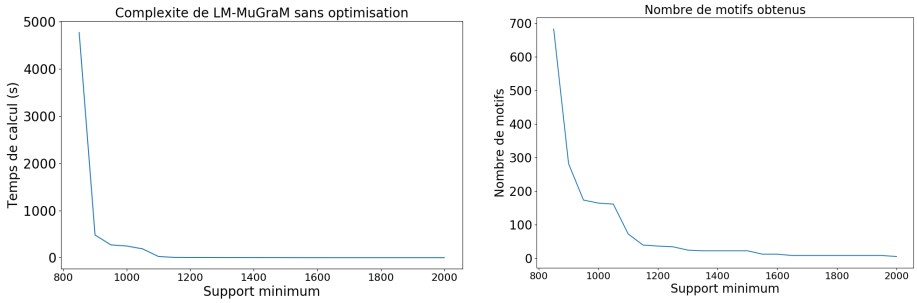


FIG. 5: Temps de recherche (s) des motifs dans le graphe Eure-et-Loir en fonction du support seuil, à gauche, et nombre de motifs trouvés, à droite

On s'intéresse maintenant à la richesse des motifs extraits. La figure 6 présente la distribution des motifs en nombre d'arcs (à gauche) et de nœuds (à droite). On observe logiquement

1. https://www.data.gouv.fr/fr/datasets/registre-parcellaire-graphique-rpg-contours-des-parcelles-et-ilots-culturaux-et-leur-groupe-de-cultures-majoritaire/

que la taille (en nœuds et arcs) des motifs trouvés augmente quand le support diminue. La diversité (en taille) des motifs augmente également. Rappelons que seuls les motifs maximaux sont extraits pour un seuil de support donné. Aux seuils les plus bas, de cent à plusieurs centaines de motifs de 4 à 7 nœuds peuvent être extraits, mais il s'agit de motifs purement temporels ou purement spatiaux. Les motifs spatio-temporels, plus rares, n'apparaissent pas.

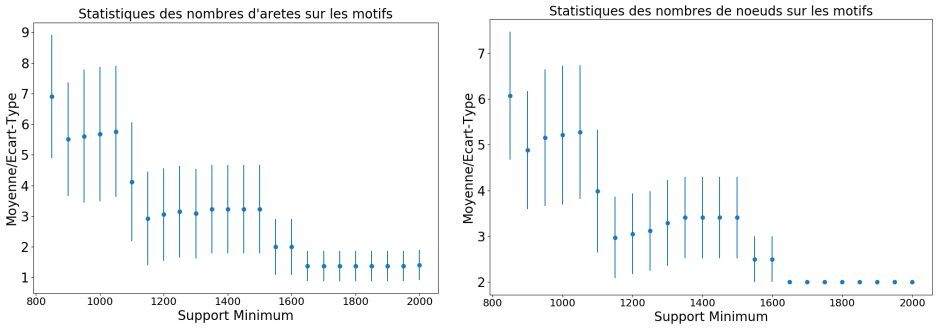


FIG. 6: Ecart-type et nombre moyen d'arcs (à gauche) et de nœuds des motifs trouvés dans le graphe Eure-et-Loir en fonction du support seuil

Dans un deuxième temps, nous avons restreint l'espace de recherche en ajoutant des contraintes sur les motifs à extraire : ils doivent contenir au moins un multi-arc spatio-temporel (ces multi-arcs étant moins nombreux que les multi-arcs spatiaux). Pour cela on ne garde que les graines contenant au moins un multi-arc spatio-temporel, ce qui a pour effet de restreindre drastiquement le nombre de graines. Nous obtenons alors des résultats avec des seuils beaucoup plus bas et en un temps beaucoup plus court. Pour les données d'Eure et Loir, l'extraction a été réalisée en 15,6 secondes pour un support de 100 : on trouve 76 motifs, de taille moyenne 3 nœuds avec un écart type de 1. Ces motifs sont de fait moins nombreux et plus petits que ceux obtenus sur le graphe complet pour des supports plus élevés (figures 5 à droite et 6 à droite). La figure 7 montre deux exemples des motifs spatio-temporels ainsi obtenus, qui représentent les occupations du sol parmi les plus fréquentes (blé tendre d'hiver 21,4%, colza d'hiver 10,4%, surface agricole temporairement non exploitée 6,7%).

Des motifs plus complexes peuvent être obtenus en un temps raisonnable avec cette approche restreignant l'espace de recherche. Un travail est en cours sur d'autres régions où la distribution des cultures (et donc des étiquettes des nœuds) est différente.

5 Conclusion et perspectives

Dans cet article, nous avons présenté LD-MuGraM, un algorithme de recherche de motifs fréquents dans un multi-graphe étiqueté orienté, fondé sur MuGraM (Ingalalli et al., 2018) et qui se comporte globalement comme lui en termes de complexité.

Nous avons testé cet algorithme sur des graphes spatio-temporels, une catégorie de multi-graphes étiquetés orientés. Pour ce faire, nous avons utilisé un générateur (Leborgne et al., 2020, 2021) afin d'avoir une vérité terrain. Ces expériences nous ont permis de mettre en évidence que LD-MuGraM retrouvait les motifs insérés. Elles ont montré qu'il était sensible aux

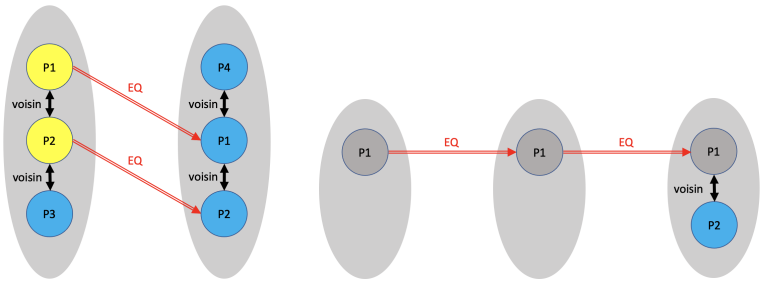


FIG. 7: Deux motifs extraits du graphe représentant les occupations du sol d'une zone d'Eure et Loir dans la période 2015-2019 : en bleu, blé tendre d'hiver, en jaune, colza d'hiver, en gris, surface agricole temporairement non exploitée

caractéristiques des graphes (densité et taille des motifs, nombre d'arcs, variété et distribution des étiquettes). L'étude des effets des différents paramètres est à poursuivre.

Nous avons également testé cet algorithme sur des données réelles, représentant l'occupation agricole des sols d'une zone d'Eure et Loir pendant la période 2015-2019. Le graphe construit, de l'ordre de 10.000 nœuds, a pu être traité par LD-MuGraM en temps raisonnable jusqu'à un support de 900. En contraignant les caractéristiques des motifs recherchés, nous avons pu descendre à un support de 100 et extraire des motifs de taille 4 à 6 permettant d'identifier des ensembles de parcelles voisines évoluant de la même manière.

Une perspective de ce travail est de rechercher des motifs fréquents approximatifs car un phénomène réel ne se répète pas forcément à l'identique. Une autre perspective pourrait être de diminuer le temps de calcul en utilisant des heuristiques ou des architectures parallèles. Les méthodes d'apprentissage profonds peuvent également être envisagées, mais nécessiteront des développements nouveaux.

Références

Agrawal, R. et R. Srikant (1994). Fast algorithms for mining association rules. In *VLDB*, pp. 487–499.

Bringmann, B. et S. Nijssen (2008). What is frequent in a single graph ? In *PKDD*, pp. 858–863.

Del Mondo, G., J. G. Stell, C. Claramunt, et R. Thibaud (2010). A graph model for spatio-temporal evolution. *Journal of Univers Comput Sci 16*, 1452–1477.

Elseidy, M., E. Abdelhamid, S. Skiadopoulos, et P. Kalnis (2014). GRAMI : Frequent Subgraph and Pattern Mining in a Single Large Graph. *VLDB 7*(7), 517–528.

Falkowski, T., A. Barth, et M. Spiliopoulou (2008). Studying Community Dynamics with an Incremental Graph Mining Algorithm. In *AMCIS Proc.*, pp. 29.

Flouvat, F., N. Selmaoui-Folcher, J. Sanhes, C. Mu, C. Pasquier, et J.-F. Boulicaut (2020). Mining evolutions of complex spatial objects using a single-attributed directed acyclic graph. *Knowl Inf Syst 62*, 3931–3971.

Ingalalli, V., D. Ienco, et P. Poncelet (2016). Sumgra : Querying multigraphs via efficient indexing. In *DEXA*, pp. 387–401. Springer.

Ingalalli, V., D. Ienco, et P. Poncelet (2018). Mining Frequent Subgraphs in Multigraphs. *Information Sciences 451-452*, 50–66.

Inokuchi, A., T. Washio, et H. Motoda (2000). An Apriori-Based Algorithm for Mining Frequent Substructures from Graph Data. In *PKDD*, pp. 13–23.

Le, N.-T., B. Vo, L. B. Nguyen, H. Fujita, et B. Le (2020). Mining weighted subgraphs in a single large graph. *Information Sciences 514*, 149–165.

Leborgne, A., M. Kirandjiska, et F. Le Ber (2021). Génération aléatoire d'un graphe spatio-temporel localement cohérent. In *CNIA*, Bordeaux, France, pp. 76–83.

Leborgne, A., A. Meyer, H. Giraud, F. Le Ber, et S. Marc-Zwecker (2019). Un graphe spatio-temporel pour modéliser l'évolution de parcelles agricoles. In *SAGEO*, pp. 1–13.

Leborgne, A., J. Nuss, F. Le Ber, et S. Marc-Zwecker (2020). An approach for generating random temporal semantic graphs with embedded patterns. In *Graph Embedding and Mining, ECML-PKDD*, pp. 1–14.

Randell, D. A., Z. Cui, et A. G. Cohn (1992). A Spatial Logic based on Regions and Connection. In *KR*, pp. 165–176. Morgan Kaufmann Publishers.

Raymond, J. W., E. J. Gardiner, et P. Willett (2002). Heuristics for similarity searching of chemical graphs using a maximum common edge subgraph algorithm. *Journal of Chem Inf and Comp Sci 42*(2), 305–316.

Scarselli, F., M. Gori, A. C. Tsoi, M. Hagenbuchner, et G. Monfardini (2009). The graph neural network model. *IEEE T Neural Networ 20*(1), 61–80.

Ying, R., A. Wang, J. Yu, et J. Leskovec (2020). Frequent subgraph mining by walking in order embedding space. In *ICML*.

Summary

This paper introduces LD-MuGraM, an algorithm based on MuGraM and designed for extracting frequent sub-graphs from directed labeled multigraphs. We focus on a specific category of multigraphs, namely spatio-temporal graphs, that represent spatial objects and their time changes. Our algorithm was tested both on synthetic data, obtained by a generator of spatio-temporal graphs including patterns, and on temporal land-use data obtained from French national databases. Experimental results are presented and discussed.

Apprentissage de comportements à partir de données temporelles hétérogènes

Nida Meddouri*, François Rioult*, Bruno Crémilleux*

*UNICAEN, ENSICAEN, CNRS - UMR GREYC, Normandie Univ – 14000 Caen France
prenom.nom@unicaen.fr

Récemment, l'analyse et l'apprentissage du comportement d'un ou plusieurs agents (humains ou artificiels) a fait l'objet de plusieurs travaux de recherche (Darty et al., 2014) (Sun et al., 2021). La modélisation d'un comportement nécessite de concevoir et mettre en place de nouvelles approches et méthodes de fouille des données. Cette modélisation dépendra des observations d'un comportement, et donc d'une trace de données en tenant compte de l'aspect temporel (Dix et al., 1993).

Dans le cadre d'un projet pour étudier le comportement des apprentis soignants, les observations seront acquises d'un simulateur de réalité virtuelle modélisant la chambre d'un patient à l'hôpital. Des données hétérogènes – numériques et catégorielles – correspondent aux actions d'un apprenant en fonction de l'état des objets de la chambre. L'analyse par des techniques d'apprentissage des traces d'un apprenant et la modélisation de son comportement sont deux défis majeurs dans le cadre de ce projet.

1 Analyse de comportements à partir des données complexes et hétérogènes

La scène mise en oeuvre par le simulateur est une chambre de patient dans un hôpital. Elle contient potentiellement 142 objets. Parmi eux, sept sont mal placés, par exemple une seringue se trouve par terre. L'apprenant devra découvrir, dans un intervalle de temps fixe (une dizaine de minutes), ces objets aux placements erronés. Le simulateur permet à un apprenant de se déplacer librement dans la chambre, sélectionner un objet à la fois, signaler une erreur et justifier sa décision interactivement parmi une liste de choix. En fonction de sa position dans la chambre et des objets dans son champ de vision, un apprenant effectuera une action qui sera annotée par le simulateur : l'apprenant peut ignorer un objet, ou le sélectionner, soit pour le signaler et justifier une erreur, soit le remettre sans le signaler. Les coordonnées et les actions de l'apprenant ainsi que l'état des objets apparaissant dans le viseur du casque seront enregistrées à chaque instant d'une simulation. Ces traces représentent une séquence de données complexes (spatiales et temporelles) et hétérogènes (quantitatives et qualitatives).

Un comportement est un ensemble de réactions d'un sujet objectivement observables (Bloch et al., 2002). Il est considéré comme un choix tactique, évolue selon la dynamique de l'environnement et dépend de l'état mental de l'apprenant. Dans le cadre de ce travail, l'apprenant effectue une action en fonction des objets présents dans la chambre virtuelle et selon ses propres

règles (une stratégie). Par exemple, un apprenant aura tendance à inspecter les objets qui se trouvent à sa droite avec une priorité pour les objets en hauteur, ensuite les objets au sol, au fur est à mesure qu'il avance dans la chambre. Un autre apprenant aura tendance à aller directement vers le patient pour inspecter sa fiche médicale, en jetant un coup d'oeil sur la totalité de objets de la chambre, avant de commencer à inspecter les objets au voisinage du patient. Dans tous les cas, une séquence temporelle est une séquence d'actions, ordonnées chronologiquement, effectuées par un apprenant durant un intervalle de temps fini.

L'objectif de notre travail est d'apprendre un modèle comportemental à partir de traces d'interactions avec les objets. De plus, nous souhaitons faire émerger un modèle explicable et interprétable, pour échanger avec l'expert de la formation des apprentis soignants. Pour cela, nous proposons une formalisation logique du comportement d'un apprenant.

2 Approche proposée

Nous avons décidé d'extraire des règles d'induction (Cohen, 1995), qui ont la particularité d'être expressives, facile à générer, à interpréter. Ces règles sont de la forme Si condition Alors conclusion. Plus précisément, nous souhaitons focaliser notre attention sur des règles dont la condition représente une conjonction d'états de la simulation et la conclusion est une action entreprise par l'apprenant. Donnons deux exemples : (i) Si une seringue usée se trouve à droite et sur la table Alors l'apprenant va la saisir et la signaler ; (ii) Si le temps écoulé est inférieur une minute et l'apprenant s'approche du patient Alors il va ignorer tout les objets de la chambre. Nous considérons que le comportement d'un apprenant est un ensemble de règles d'induction.

L'étape suivante consiste à analyser et comparer les comportements de plusieurs apprenants pour enrichir le retour d'expérience auprès de l'expert. Nous proposons donc de regrouper les comportements similaires. Pour cela, nous allons générer un ensemble de règles associé à chaque comportement. Dans un premier temps, nous allons comparer les modèles générés en utilisant une mesure statistique comme indice de similarité. Cette mesure nous permet de construire des groupes homogènes d'apprenants à partir de comportements similaires. Enfin, il sera utile de caractériser chaque groupe par un comportement spécifique.

Pour pallier le problème de l'hétérogénéité des données, nous prévoyons de discrétiser les attributs quantitatifs, de façon a faire émerger des séquences de item ou d'ensembles d'items. Il nous semble raisonnable d'appliquer ensuite des algorithmes d'extraction de motifs séquentiels (Fournier Viger et al., 2017) disponibles dans la communauté [1].

3 Conclusion

Dans ce travail, chaque comportement d'apprenant est représenté par un ensemble de règles d'induction. L'analyse et la comparaison de ces ensembles permettront de grouper les comportements similaires des apprenants selon une mesure de similarité. Un comportement typique est détectable à partir des règles en commun dans les ensembles similaires.

Remerciements Ce travail est financé par le projet INCA – Interactions Naturelles avec des Compagnons Artificiels (région Normandie).

1. Par exemple `https://www.philippe-fournier-viger.com/spmf/`

Références

Bloch, H., R. Chemama, et E. Dépret (2002). *Grand dictionnaire de la psychologie*. Collectif Larousse.

Cohen, W. W. (1995). Fast effective rule induction. In *In Proceedings of the Twelfth International Conference on Machine Learning*, pp. 115–123. Morgan Kaufmann.

Darty, K., J. Saunier, et N. Sabouret (2014). Analyse des comportements agents par agregation aux comportements humains.

Dix, A., J. Finlay, et R. Beale (1993). Analysis of user behaviour as time series. In *Proceedings of the Conference on People and Computers VII*, HCI'92, USA, pp. 429–444. Cambridge University Press.

Fournier Viger, P., C.-W. Lin, U. Rage, Y. S. Koh, et R. Thomas (2017). A survey of sequential pattern mining. *Data Science and Pattern Recognition 1*, 54–77.

Sun, D., T. Li, F. You, M. Hu, et Z. Li (2021). Prediction of learning behavior characters of MOOC's data based on time series analysis. *Journal of Physics : Conference Series 1994*(1), 012009.

Summary

L'apprentissage d'un comportement à partir des données temporelles a fait l'objet de plusieurs travaux de recherche récemment. Les approches et les techniques de la fouille des données temporelles, sont des outils qui permettent de modéliser un comportement. Dans le cadre d'une chambre de patient dans un hôpital, nous proposons de sauvegarder une trace des actions d'un apprenant soignant et l'état de chaque objets dans cette chambre à chaque instant. Cette trace permettra de modéliser le comportement d'un apprenant sous forme d'un ensemble de règles d'induction. L'analyse et la comparaison des ensembles de règles générées permettra de comparer et regrouper les comportements de plusieurs apprenants et de déduire un comportement typique parmi ceux les plus similaires.