

INFORSID 2020 Forum Jeunes Chercheuses Jeunes Chercheurs

INFORSID 2020 Forum of junior researchers

Hiba Abou Jamra¹, Landy Andriamampianina^{2,3}, Cécile Cayère⁴, Yohann Chasseray⁵, and Pierre-Emmanuel Arduin⁶

¹Laboratoire d'Informatique de Bourgogne (EA 7534), Université Bourgogne Franche-Comté, 9, Avenue Alain Savary, 21078 Dijon, France, Hiba_Abou-Jamra@etu.u-bourgogne.fr

²Institut de Recherche en Informatique de Toulouse (UMR 5505), Université Toulouse 1 Capitole (UT1), 2 Rue du Doyen-Gabriel-Marty, 31042 Toulouse, France,

³Activus Group, 1 Chemin du Pigeonnier de la Cépière, 31100 Toulouse, France, landy.andriamampianina@activus-group.fr

⁴La Rochelle Université, 23 Avenue Albert Einstein, 17000 La Rochelle, France, cecile.cayere1@univ-lr.fr

⁵Laboratoire de Génie Chimique, Université de Toulouse, CNRS, INPT, UPS, 4 allée émile Monso, 31030 Toulouse, France, yohann.chasseray@ensiacet.fr

⁶Université Paris-Dauphine – PSL, DRM (UMR 7088), Place du Maréchal de Lattre de Tassigny, 75775 Paris Cedex 16, France, pierre-emmanuel.arduin@dauphine.psl.eu

RÉSUMÉ. Cet article reprend une sélection de quatre des sept articles courts du Forum Jeunes Chercheuses Jeunes Chercheurs (JCJC) d'INFORSID 2020. Instantané du paysage de la recherche française par des doctorants en systèmes d'information, cet article considère les thèmes de la détection de signaux faibles dans les réseaux sociaux, des graphes temporels, des traces spatio-temporelles numériques dans le domaine du tourisme et du peuplement de bases de connaissances.

ABSTRACT. This article presents a selection of four of the seven short articles from the Forum of junior researchers (JCJC) of INFORSID 2020. Snapshot of the French research landscape by doctoral students in information systems, this article considers the topics of the detection of weak signals in social networks, temporal graphs, digital traces in the field of tourism and populating knowledge bases.

MOTS-CLÉS. Signaux faibles, Graphes temporels, Données spatio-temporelles, Bases de connaissances.

KEYWORDS. Weak signals, Temporal graphs, Spatio-temporal data, Knowledge bases.

1. Introduction ¹

Tous les deux ans, le Forum jeunes chercheuses jeunes chercheurs (JCJC) d'INFORSID permet à des doctorants en première ou deuxième année de thèse de confronter leurs travaux à la communauté INFORSID. Il prodigue aussi – et surtout – un instantané du paysage de la recherche française en systèmes d'information. Quelles préoccupations portent ces collègues en devenir ? Quels sont leurs objets d'étude ? Quelles motivations scientifiques, industrielles ou sociétales ont les doctorants français d'aujourd'hui en systèmes d'information ? Le Forum JCJC d'INFORSID permet d'en apprendre davantage sur ces questions, tout en transmettant à des doctorants les conseils, la bienveillance et la rigueur qui font la force de la communauté scientifique française en systèmes d'information.

Tous les deux ans, donc, le Forum JCJC se tient pendant le congrès INFORSID et les doctorants ayant soumis un article établissent un premier contact avec la communauté. Tous les deux ans sauf... en 2020 ! La crise épidémique mondiale a entraîné l'annulation du congrès INFORSID 2020 à Dijon, mais ce n'a été que partie remise puisque les auteurs d'un article du Forum JCJC 2020 ont pu le présenter pendant

1. Par Pierre-Emmanuel Arduin, organisateur du Forum JCJC 2020

le congrès INFORSID 2021 en distanciel. Cet article reprend une sélection de quatre des sept articles courts du Forum JCJC 2020. Il conviendra de remercier les auteurs – ainsi que leurs directeurs de thèse – pour leur contribution et leur capacité à intégrer les remarques des relecteurs dans un temps record et un contexte de travail à distance tendu que nous avons tous traversé.

Dans la deuxième section, Hiba Abou Jamra présente une approche pour détecter des signaux faibles dans les réseaux sociaux. L'approche est basée sur l'analyse de la structure latente du réseau et se fait par le biais de graphlets. Une série temporelle est construite et les intervalles précédents et pendant un événement significatif sont analysés en terme de vitesse de croissance/décroissance du nombre de graphlets. Dans la troisième section, Landy Andriamampianina s'intéresse à la dimension temporelle des informations au travers de graphes temporels. Un état de l'art des études existantes ainsi que de leurs limites y est présenté. Notamment, la conception des concepts et des formalismes pour le développement d'applications basées sur des graphes temporels est envisagée. Dans la quatrième section, Cécile Cayère propose une approche de traitement et d'exploitation de traces spatio-temporelles numériques dans le domaine du tourisme. L'objectif de cette recherche est de concevoir une plateforme modulaire permettant de créer des chaînes de traitement personnalisées afin de faciliter l'analyse de ces traces. Dans la cinquième section, Yohann Chasseray tente de réduire au maximum le nombre de modules d'importation de données brutes dans un processus de peuplement d'une base de connaissances. Une telle base est souhaitée stable même lors de l'intégration d'une nouvelle source de données brutes, ce que l'auteur propose de discuter.

2. Analyse de la structure latente des réseaux sociaux par graphlets²

2.1. Introduction

Les réseaux sociaux sont utilisés par des millions de personnes liées entre elles par différents types de relations avec pour objectif de produire et de partager des informations sur différents sujets. Ils constituent un reflet de notre société et sont considérés comme une chambre d'écho. Nous proposons une approche basée sur la topologie des graphes pour détecter les signaux faibles qui sont des signes d'alerte précoce d'une menace ou d'une opportunité pour une entreprise. Les signaux faibles sont utilisés en économie, en marketing, en sciences de la terre, en épidémiologie, etc.

2.2. État de l'art

La première théorisation des signaux faibles a été proposée par Ansoff (1975) qui place son étude dans le contexte de la planification et de la gestion des enjeux stratégiques des entreprises. Nous définissons *un signal faible comme une information qui fournit une indication sur des événements à venir ou émergents pouvant avoir des répercussions significatives sur le système.*

La majorité des méthodes analysent des mots-clés afin d'en identifier certains comme signaux faibles. Elles sont basées :

- sur les changements dans la communication avec l'utilisation conjointe de séries temporelles pour prendre en compte l'évolution de la force des signaux et l'analyse de sentiments pour tenir compte des changements de vocabulaire dans les discours ;

2. Par Hiba Abou Jamra

- sur la cartographie des mots-clés qui, en mettant en avant des changements dans leur taux d'apparition, peut révéler une évolution de signaux faibles en signaux forts ;
- sur le regroupement de mots-clés où un mot-clé signal faible se trouve dans un seul groupe fortement cohérent, disjoint des autres et avec un nombre d'occurrences généralement faible ;
- sur des méthodes de probabilité qui déterminent si un signal passe de faible à fort.

Rousseau et al. (2021) ont proposé un état de l'art sur les signaux faibles et les différentes méthodes de leur détection.

2.3. Problématique

La quantité des données produites par les réseaux sociaux est si importante que les méthodes classiques s'appuyant sur des statistiques simples ne permettent pas d'extraire les signaux faibles. La construction d'outils algorithmiques travaillant plus localement est nécessaire. Nous avons alors envisagé trois approches possibles parmi les outils du traitement des signaux comme le débruitage et la décomposition en ondelettes (Ranta et al., 2003), les graphons (Glasscock, 2016) et les graphlets. Cependant, une décomposition en ondelettes sur une matrice d'adjacence ou de distance entre les nœuds est difficilement interprétable. Les graphons s'appliquent plutôt sur des graphes denses, ce qui n'est pas le cas des graphes générés par les données des réseaux sociaux. Nous avons donc choisi de privilégier la piste des graphlets. En effet, les graphlets présentent des caractéristiques généralement associées aux signaux faibles (Lesca, 2002) :

- ce sont de **petits motifs** (fragments d'un graphe) composés de peu de liens entre les nœuds ;
- certains d'entre eux sont **rares** dans un grand volume d'informations ;
- ils sont cependant **interprétables** par les experts métier grâce à leurs formes et orbites prédéfinies.

2.4. Approche et expérimentation

Dans la suite, nous introduisons notre méthode dont l'objectif est d'aider les experts métier à détecter et à interpréter les signaux faibles identifiés afin de leur permettre de prendre des décisions et de planifier des stratégies.

Les graphlets ont été introduits la première fois par Pržulj et al. (2004). Un graphlet est un sous-graphe non isomorphe induit connecté (2 à 5 nœuds) choisi parmi les nœuds d'un grand graphe. Il existe 30 différents types de graphlets, notés de G_0 à G_{29} . La figure 1 montre les 9 premiers graphlets de 2 à 4 nœuds.

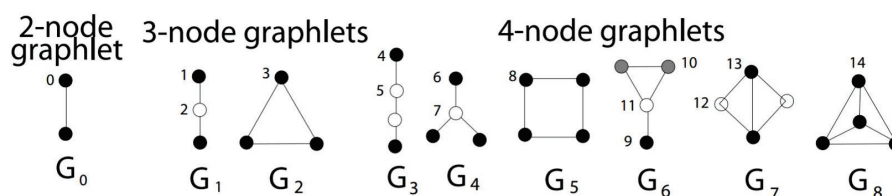


FIGURE 1. Représentation des 9 premiers graphlets de 2 jusqu'à 4 nœuds

Plusieurs méthodes d'énumération de graphlets ont été proposées comme Orca (Hočevar and Demšar, 2014) que nous utilisons dans notre approche³. Les tailles des graphlets et leurs formes, ainsi que les orbites (c'est-à-dire les positions ou les rôles des nœuds dans les graphlets) (Charbey, 2018) permettent de contextualiser les résultats obtenus. Il existe 73 orbites (de O_0 à O_{72}) pour les 30 graphlets. Dans la figure 1, pour un même graphlet, les orbites ayant une même couleur sont échangeables.

Nous avons réalisé des expérimentations sur plusieurs jeux de données, dont celles issues de la plateforme du projet Cocktail⁴. Cocktail a pour but de créer un observatoire en temps réel des tendances, des innovations et des signaux faibles circulant dans les discours des contextes métiers alimentaire et santé sur Twitter.

Un cas réel de situation de crise avec des données issues de Twitter est étudié, les objectifs et les premiers résultats de deux autres expériences sont montrés. Pour terminer, nous présentons nos perspectives.

Cas d'étude : visite du Président Macron à Rouen le 30 octobre 2019 dans le cadre de l'incendie à l'usine Lubrizol

Le 26 septembre 2019, une partie de l'usine Lubrizol à Rouen a été ravagée par un incendie, l'intérêt de l'évènement nous a amené à lancer une collecte pour laquelle 47 hashtags et 111 comptes utilisateurs ont été fournis par les chercheurs en sciences humaines et sociales du projet Cocktail comme critères de collecte. Cette collecte a ramené environ 2 millions de tweets entre le 26 septembre et le 26 novembre 2019. Nous avons ensuite nettoyé ces données en filtrant les tweets contenant `lubrizol` et `rouen` entre le 11 octobre et le 24 novembre. À l'issue de cette opération, le corpus est réduit à 137 561 tweets dont 16 100 sont des tweets originaux et 57 649 comportent des mentions. En construisant la série temporelle des tweets, nous avons vu que la venue du Président Macron à Rouen le 30 octobre au soir a provoqué une forte émission de tweets, indiquant que cette venue était un évènement.

Nous avons modélisé les tweets sous la forme d'un graphe temporel dans lequel les nœuds représentent les comptes utilisateurs et les arêtes la relation Mention. Ce graphe est ensuite découpé en une séquence de s snapshots S^t de même durée égale à une heure, pour la période choisie (avant et durant la visite du président). Pour chaque snapshot, nous avons réalisé les analyses suivantes :

1. À l'aide de l'algorithme Orca, nous avons compté le nombre de graphlets de 2 à 5 nœuds : $(G_0^t, G_1^t, \dots, G_{29}^t)$, G_x^t est le nombre de graphlets de type x dans le snapshot S^t ;
2. Après avoir normalisé le nombre des graphlets \overline{G}_x^t (la normalisation est inspirée de la procédure proposée par Goldin and Kanellakis (1995)), nous calculons leur vitesse afin de mettre en évidence les graphlets qui « émergent rapidement avant les autres » : $\overline{V}_x^t = \overline{G}_x^{t+1} - \overline{G}_x^t, \forall x \in \{0, \dots, 29\}$.

Nous choisissons les k premiers graphlets qui présentent une croissance importante de leur vitesse, par rapport aux autres. Le tableau 1 est un extrait des vitesses pour les trois snapshots avant la visite du Président qui a eu lieu vers 18 heures le 30 Octobre.

Le calcul des vitesses met en évidence certains graphlets dont la vitesse est plus grande que les autres, en particulier dans le snapshot S^{t+1} , c'est-à-dire deux heures avant la visite du Président. Dans le tableau

3. Le code source de l'algorithme Orca est disponible sur <https://github.com/thochevar/orca>

4. Ce travail est soutenu par le programme « Investissements d'Avenir », projet ISITE-BFC (*Initiatives Science Innovation Territoires Économie en Bourgogne-Franche-Comté*) contrat ANR-15-IDEX-0003, <https://projet-cocktail.fr/>











Snapshots du 30 Octobre	G_2 	G_5 	G_8 	G_{15} 	G_{27} 	G_{17} 	G_{22} 
S^t : 15h-16h	0.1869	0.1417	0	0	0	0	0
S^{t+1} : 16h-17h	3.2248	3.1387	3.7340	5.2034	5.5233	0.2030	0.1979
S^{t+2} : 17h-18h	0.4206	0.7610	1.5558	-2.8019	-0.9152	0.5379	0.0727

TABLEAU 1. Vitesses des graphlets pour les trois snapshots précédant l'évènement.

1, les graphlets G_2 , G_5 , G_8 , G_{15} et G_{27} correspondent à ce critère (valeurs surlignées en bleu), ils sont alors considérés comme des précurseurs de cet évènement. Alors que d'autres graphlets, comme G_{17} et G_{22} , n'ont pas présenté un changement important dans leur vitesse pour le même snapshot (valeurs surlignées en rouge).

Nous calculons ensuite la contribution de chacun des graphlets précurseurs par rapport à l'évolution globale de tous les graphlets à l'aide d'un calcul de ratio dans lequel le nombre total d'un graphlet de type x est divisé par le nombre total de graphlets, pour tous les snapshots : $R(G_x) = \frac{\sum_{t=1}^s (G_x^t)}{T(G)}$ avec $T(G) = \sum_{t=1}^s (\sum_{x=0}^{29} (G_x^t))$, s est le nombre de snapshots. Nous remarquons que les graphlets G_2 , G_5 , G_8 et G_{27} ont des valeurs de ratio faibles et sont donc rares, ce qui permet de les qualifier de signaux faibles, alors que le graphlet précurseur G_{15} est éliminé car son ratio est fort.

Nous avons aussi évalué les résultats obtenus à l'aide d'une mesure statistique « cross-correlation », proposée par Ripley and Venables (2002), et qui calcule la corrélation entre les séries temporelles des graphlets et la série temporelle initiale de l'étude. Les graphlets G_2 , G_5 , G_8 et G_{27} présentent des corrélations positives (valeurs entre 0.7 et 0.8) avec un décalage d'une et deux heures avant la série temporelle initiale, ce qui confirme que ces graphlets sont des signaux faibles.

Après avoir identifié les graphlets signaux faibles, nous nous intéressons à la sémantique qu'ils portent en étudiant leurs instances. Nous remarquons que des utilisateurs influents (détectés par application de l'algorithme PageRank) comme les journalistes `manon_leterq` et `massinfabien`, et le site d'information local `76actu`, apparaissent surtout dans l'orbite O_3  de G_2 , et les orbites O_{68}  et O_{69}  de G_{27} . Ces utilisateurs semblent avoir un rôle informatif et d'alerte et doivent être surveillés par les experts métier.

Nous avons réalisé des expériences supplémentaires sur d'autres jeux de données pour valider notre méthode⁵. Elles ont pour objectifs :

La reproductibilité : Nous avons choisi les retweets publiés sur les 6 épisodes de la dernière saison de la série télévisée Game of Thrones pour tester la reproductibilité de notre méthode. Les résultats obtenus ont montré que les graphlets signaux faibles sont détectés quelques heures avant l'heure de diffusion des 6 épisodes. De plus, nous avons remarqué que les mêmes types de graphlets se retrouvent dans plusieurs épisodes comme signaux faibles ;

L'absence de faux positifs : Nous avons analysé les interactions entre les élèves d'une école primaire sur deux jours consécutifs⁶, avec comme évènements, les pauses du midi et les récréations prévus à

5. Les données et les programmes sont disponibles sur : <https://github.com/hibaaboujamra/Weak-Signals-Detection-and-Interpretation-BEAM>

6. Le jeu de données est disponible sur <http://www.sociopatterns.org/datasets/primary-school-temporal-network-data/>

des moments connus. Nous avons montré que la méthode ne détecte aucun signal faible de ces événements connus. Nous avons ainsi montré l'importance du calcul de la contribution des graphlets précurseurs, qui permet de qualifier certains de signaux faibles.

Nous avons également réalisé des expériences sur les réseaux Facebook et MathOverflow, avec des événements inconnus ou aléatoires. Ces expériences ont montré que les jeux de données ne doivent pas être trop généralistes.

2.5. Conclusion et perspectives

Les expériences réalisées ont montré la robustesse de notre méthode, et ont conforté notre hypothèse qui considère que les graphlets peuvent être vus comme des signaux faibles.

Nous travaillons actuellement sur l'impact du changement des paramètres de la méthode, comme la période de l'étude, la taille des snapshots et le nombre des graphlets choisis comme précurseurs, sur la détection des signaux faibles. Nous étudions aussi notre méthode sur des graphes de co-occurrences de hashtags, ainsi que l'évolution de la structure communautaire une fois les signaux faibles identifiés. De surcroît, de nouvelles expériences vont être réalisées sur d'autres types et sur de plus grands réseaux.

3. Vers une approche générique pour capturer l'évolution temporelle dans les graphes⁷

3.1. Introduction

Un graphe modélise naturellement les relations complexes entre des entités telles que celles qu'on trouve dans les réseaux de communication, informatiques, sociaux, biologiques, etc. Un graphe se compose d'un ensemble de nœuds représentant des entités, et d'un ensemble de liens représentant des relations entre les entités. Il est généralement statique, c'est-à-dire qu'il reflète un snapshot d'un graphe⁸. Cependant, les applications du monde réel reposant sur une structure en graphe évoluent constamment dans le temps. Les graphes statiques ne parviennent pas à représenter, gérer et suivre leurs changements dans le temps. Certaines études ont introduit le concept de graphe temporel pour modéliser l'évolution temporelle dans un graphe. Les graphes temporels existants permettent de modéliser certains types d'évolution et sont spécifiques à certaines applications. L'objectif de notre travail est de proposer une solution générique pour gérer tout type d'évolution temporelle pour toute application basée sur les graphes.

3.2. État de l'art

Un graphe temporel est un graphe dont les composants (nœuds, liens, attributs) changent au cours du temps. En d'autres termes, l'évolution temporelle d'un graphe montre les changements qui se produisent sur ces composants. Elle peut être classée en deux types, à savoir (i) *l'évolution de topologie*⁹ et (ii) *l'évolution des données*¹⁰ (Zaki et al., 2016).

7. Par Landy Andriamampianina

8. une image du graphe entier à un temps donné

9. la manière dont les nœuds et les liens sont disposés dans un graphe.

10. la valeur des attributs des nœuds et des liens.

L'évolution temporelle de la topologie d'un graphe est discutée dans la littérature à travers l'ajout et la suppression de nœuds et de liens au cours du temps (Kostakos, 2009). Quant à l'évolution temporelle des données d'un graphe, elle est discutée dans la littérature à travers les changements de la valeur des attributs des nœuds (Desmier et al., 2012) ou des liens (Zhao et al., 2020) au cours du temps. Néanmoins, les évolutions ne se limitent pas à ces deux types. L'ajout et la suppression de nouveaux attributs sont ignorés dans les travaux existants et méritent de faire de l'objet d'une étude plus aboutie en vue d'une gestion complète de l'évolution.

Pour gérer ces évolutions, les approches classiques dans la littérature se basent sur la séquence de snapshots d'un graphe¹¹. Cette dernière consiste à découper le temps en plusieurs fenêtres et créer un snapshot du graphe pour chaque fenêtre de temps (Fard et al., 2012). D'autres études gèrent l'évolution temporelle du graphe en attachant à chaque nœud et lien du graphe son temps de présence au lieu d'attacher le temps au graphe entier (Latapy et al., 2017). Il est nécessaire de comparer ces approches afin de proposer une gestion efficace et efficiente de ces évolutions.

3.3. *Problématique*

Le premier problème que nous voulons aborder dans nos travaux de recherche est la représentation de plusieurs types d'évolution temporelle d'une application basée sur un graphe. L'état de l'art a montré que les modèles de graphes temporels existants ne capturent que partiellement l'évolution temporelle d'un graphe. A notre connaissance, ils ne capturent pas l'évolution temporelle de la structure de données d'un graphe. Pourtant, ce type d'évolution est une nouvelle source d'information qui peut enrichir l'analyse d'une application. A titre d'exemple, considérons un réseau social où les utilisateurs et leurs relations sont respectivement représentés par des nœuds et des liens dans un graphe temporel. Les nœuds et les liens ont des attributs pour décrire les caractéristiques des utilisateurs et de leurs relations. Lorsqu'un utilisateur ajoute un nouvel ami, il crée un nouveau lien dans le graphe modifiant la topologie du graphe et son attribut « nombre d'amis » est recalculé et mis à jour. Lorsqu'un utilisateur obtient un emploi pour la première fois, un nouvel attribut appelé « statut professionnel » doit être ajouté à l'ensemble de ses attributs. Cette information peut constituer une nouvelle connaissance pour comprendre les tendances d'évolution dans le réseau social. Par exemple, une augmentation importante du nombre d'amis d'un utilisateur peut être corrélée avec l'ajout de son statut professionnel. Cela signifie qu'il a peut-être noué de nouvelles relations professionnelles. Par conséquent, certaines applications nécessitent l'aspect complet de l'évolution temporelle du graphe : la topologie, les données et la structure des données. Dans ce contexte, la première question de recherche à laquelle nous voulons répondre est de savoir comment étendre et généraliser les concepts de graphes temporels existants afin qu'ils soient suffisamment génériques pour capturer tout type d'évolution et réutilisables dans toute application incluant de l'évolution et suivant une structure de graphe.

Le deuxième problème que nous voulons aborder dans nos travaux de recherche est le mécanisme d'historisation des modèles de graphes temporels. La séquence de snapshots est l'approche de modélisation la plus utilisée. Cependant, elle ne fournit pas une solution directe pour suivre et conserver les changements dans un graphe. D'une part, il faut choisir des fenêtres de temps de capture des snapshots pertinentes pour s'assurer d'extraire des informations intéressantes. Ce qui présente un sujet de recherche

11. désignée par G_1, G_2, \dots, G_T où G_i est une image du graphe entier à l'instant i et $[1; T]$ est la durée de vie du graphe.

en soi (Ribeiro et al., 2013). D'autre part, pour observer des changements dans le graphe, il faut identifier les différences entre deux snapshots. Il est alors difficile de suivre l'évolution individuelle des nœuds et des liens. Cela est dû au fait que l'évolution temporelle est gérée au niveau du graphe entier. Dans ce contexte, la deuxième question de recherche à laquelle nous voulons répondre est de savoir comment conserver et suivre les traces d'évolution à différents niveaux du graphe (nœud, liens et graphe) contrairement aux approches existantes basées sur les snapshots.

3.4. Proposition

L'objectif de nos travaux est de proposer des solutions pour le développement d'applications basées sur les graphes temporels. Pour ce faire, nous étudions deux axes de recherche : la modélisation et la manipulation des graphes temporels. Dans le premier axe de recherche, notre but est de mettre en place une solution de modélisation complète qui couvre trois niveaux à savoir, conceptuel, logique et physique. Cette solution de modélisation doit capturer tout type de changement et son implémentation doit permettre une interrogation efficace (identifier les évolutions de manière directe via le modèle) et efficiente (produire des temps d'exécution raisonnables). Dans le second axe de recherche, notre but est de mettre en place une solution de manipulation des graphes temporels qui se fait de manière déclarative via des opérateurs algébriques pour rendre la complexité technique transparente aux utilisateurs non-experts.

3.5. Travaux effectués

Nous avons proposé une solution complète pour la gestion des données de graphes temporels, allant d'un modèle de données conceptuel orienté métier à une implémentation (Andriamampianina et al., 2021).

Nous avons défini notre graphe temporel conceptuel comme représentant des entités et des relations évoluant dans le temps appelées respectivement *entités temporelles* (distinguées de manière unique via leur identifiant) et *relations temporelles*. Une entité (ou relation) temporelle est composée d'un ensemble d'états. Chaque état contient une liste d'attributs ainsi que leur valeur durant une période de validité. A travers ces concepts, on gère, d'une part, l'évolution de la topologie en comparant deux ensembles d'états (d'entités et/ou de relations) correspondant à des périodes de validité différentes, et d'autre part, l'évolution des données et de leur structure en comparant deux états (d'une même entité ou relation).

Par rapport aux modèles conceptuels existants, notre modèle présente plusieurs avantages. Tout d'abord, il gère l'évolution temporelle à un niveau de granularité plus fin en affectant des périodes de validité aux entités et aux relations plutôt qu'au graphe entier. Ensuite, il permet de capturer tout type de changement qui peut exister dans un graphe. Cela se fait par la création d'un nouvel état lors (i) de l'ajout et la suppression d'une entité ou d'une relation, (ii) de l'ajout et la suppression d'un ou plusieurs attributs d'une entité ou d'une relation, et (iii) du changement de la valeur des attributs d'une entité ou d'une relation.

Nous avons implémenté notre modèle à travers une série d'expérimentations pour évaluer sa faisabilité, son utilisabilité, son efficacité et son passage à l'échelle. Pour ce faire, nous avons utilisé le système de gestion de bases de données orientées graphes Neo4j. Pour éviter tout biais dans nos expérimentations,

nous avons utilisé à la fois un jeu de données d'un benchmark et trois jeux de données réelles. Ces jeux de données fournissent différents facteurs d'échelle. De plus, nous avons créé des requêtes de référence avec une couverture complète des différents types d'analyse possibles (classique, temporelle ou focalisée sur des composants du graphe).

Nous avons tout d'abord mené une étude de faisabilité pour prouver que notre modèle est implémentable et interrogeable. Pour cela, nous avons implanté notre modèle et lancé les requêtes dans trois bases de données graphes différentes dans Neo4j. Puis, nous avons mené une étude d'utilisabilité pour évaluer la facilité d'exploration des données graphes temporelles avec notre modèle. Nous avons proposé différentes analyses via un ensemble de requêtes exécutées sur un jeu de données réelles. Ensuite, nous avons fait une étude comparative de l'efficacité de notre modèle avec deux modèles basés sur les snapshots pour évaluer ses avantages. Nous avons implanté les trois modèles et lancé les requêtes dans Neo4j pour comparer le temps d'exécution des requêtes. Enfin, nous avons mené une étude des temps d'exécution des requêtes quand les jeux de données passent à l'échelle. Pour cela, nous avons implanté notre modèle dans Neo4j en nous basant sur les trois jeux de données réelles ayant différentes échelles pour comparer le temps d'exécution des requêtes.

3.6. Travaux en cours et futurs

En nous basant sur notre solution proposée de modélisation de graphes temporels, nous avons entamé la proposition d'une solution de manipulation de graphes temporels. Dans un premier temps, nous proposons un ensemble de requêtes pour expliciter la pertinence et la puissance d'analyse descriptive de notre solution. Dans un second temps, nous souhaitons compléter les possibilités d'analyse de graphes temporels en proposant des opérateurs d'analyse prédictive. Pour répondre à ce besoin, nous souhaitons orienter nos travaux vers l'utilisation et l'extension voire la transformation d'algorithmes d'apprentissage afin de supporter des analyses prédictives sur des graphes temporels tels que nous les avons définis précédemment. Dans un troisième temps, ces différentes propositions seront formalisées via des opérateurs algébriques exploitant les graphes temporels.

Pour faciliter l'exploration des données graphes temporelles, nous travaillons également sur un outil d'exploration reposant sur une interface graphique dédiée à des non-informaticiens. Un premier prototype a été développé. Il permet de restituer de manière ergonomique des données et leurs changements éventuels; ce qui n'était pas possible avec les outils actuels du marché dédiés à la gestion et à l'interrogation de graphes. Cela est dû au fait qu'ils sont initialement conçus pour des graphes statiques. La prochaine étape de nos travaux doit s'orienter vers la proposition complète d'analyse de graphes temporels telles que l'analyse descriptive, l'analyse exploratoire et l'analyse prédictive.

4. Plateforme ETL dédiée à l'analyse de la mobilité touristique dans une ville¹²

4.1. Contexte

La traçabilité de la mobilité humaine est un phénomène qui prend beaucoup d'ampleur de par l'évolution des technologies GPS et l'augmentation des déplacements humains. Dans le projet régional

12. Par Cécile Cayère

Nouvelle-Aquitaine DA3T (Dispositif d'Analyse des Traces numériques pour la valorisation des Territoires Touristiques), nous faisons l'hypothèse que les traces numériques de mouvement laissées par les touristes peuvent aider les décideurs locaux dans la gestion et l'aménagement des territoires touristiques. Notre contribution vise des verrous de recherche *Mobilités et Trajectoires*¹³ à des fins : "(i) d'extraction de sémantique à partir de données issues de capteurs, (ii) de conception de méthodes de confrontation et de traitement conjoint de ces données avec des données issues d'enquêtes et d'analyse de l'environnement et (iii) de meilleure compréhension des comportements de mobilité". Nous utilisons la fouille de données pour (i) calculer des indicateurs à partir de trajectoires brutes (p. ex. objectif du déplacement, arrêts et déplacements, points chauds, etc.), (ii) créer différentes interprétations des trajectoires sur la base d'annotations et (iii) catégoriser les trajectoires de touristes.

Nous disposons actuellement de traces numériques de touristes volontaires issues de l'application mobile GeoLuciole. Elle a été développée dans le cadre du projet et capture à intervalles de temps réguliers la position spatiale horodatée d'un téléphone mobile. Dans ce contexte, l'objectif de ce travail est de concevoir une plateforme modulaire permettant de créer des chaînes de traitement personnalisées afin de faciliter l'analyse de ces traces. Elle doit être simple d'utilisation et ergonomique car elle est destinée à des utilisateurs non-informaticiens. Elle doit proposer un ensemble d'outils modulaires permettant, par exemple, l'automatisation de la détection de zones d'intérêt (p. ex. points chauds et froids etc.) ou la corrélation d'un certain comportement de mobilité avec le contexte courant (p. ex. météo, période de l'année, etc.).

4.2. État de l'art

De la même manière que nous laissons des empreintes de notre passage dans un environnement physique, nous laissons également des empreintes numériques dans un environnement informatique. La trace est l'observation et l'interprétation d'un ensemble d'empreintes par un observateur (Mille, 2013). Lorsque l'empreinte possède une dimension spatiale et une dimension temporelle, on parle de **données spatio-temporelles**. (Flouvat, 2019) classe ces données dans quatre catégories : mobilité, événement, région et réseau. Les données qui nous intéressent sont de type mobilité et décrivent les positions spatiales d'objets mobiles dans le temps, nous parlons de **traces de mouvement**. (Parent et al., 2013) décrivent une **trajectoire brute** comme une sous-partie d'intérêt de la trace de mouvement exhaustive. Pour transformer une trace de mouvement en une trajectoire brute, elle doit passer par un processus de reconstruction qui consiste à (i) nettoyer les données imprécises en les corrigeant ou les supprimant, (ii) faire de la cartospondance lorsque la trace de mouvement suit un réseau routier ou piétonnier (Newson and Krumm, 2009) et (iii) compresser les données afin de pouvoir les stocker efficacement. Afin de réaliser une analyse plus poussée, une trajectoire brute peut être enrichie avec des données provenant (i) de bases de données externes (p. ex. bases de données ouvertes du tourisme, des collectivités, etc.), (ii) de capteurs externes, (iii) de calculs utilisant les données brutes (p. ex. vitesse, orientation, etc), (iv) de l'avis d'experts ou (v) d'enquêtes ou d'interviews. (Spaccapietra and Parent, 2011) proposent la notion d'**annotation** pour lier ces données additionnelles à la trajectoire ou à une partie de la trajectoire. Il introduit également le concept d'**épisode**. Il s'agit de la partie de trajectoire la plus longue répondant à un prédicat défini, basé sur les positions spatio-temporelles ou sur les annotations. Une **interprétation** de la trajectoire est une

13. Website : <http://gdr-magis.imag.fr/actions-prospectives/mobilites-et-trajectoires/>

séquence d'épisode de cette trajectoire appartenant à une thématique donnée (p. ex. modes de transport, activités, etc.), c'est-à-dire que tous les épisodes de cette interprétation appartiennent à cette thématique. Ainsi enrichie, la trajectoire brute devient une **trajectoire sémantique**. Les trajectoires peuvent être analysées de manière individuelle ou agrégée. Ces deux types d'analyse permettent de traiter la problématique de la mobilité touristique sous différents angles.

4.3. Conception d'une plateforme modulaire

Notre objectif est de proposer une plateforme modulaire permettant à un utilisateur de construire sa propre chaîne de traitement à partir des modules disponibles pour faciliter l'analyse des données de mobilité dont il dispose. Un module prend un ensemble de trajectoires (dans un format défini selon notre modèle) en entrée, effectue un traitement sur ces données en tenant compte du paramétrage et renvoie un ensemble de trajectoires résultant de ces traitements (dans le format défini selon notre modèle). Les modules s'enchaînent dans la chaîne de traitement afin de réaliser un traitement plus élaboré des données. Les modules peuvent être classés dans trois étapes inspirées des procédures ETL (i.e. EXTRACT, TRANSFORM, LOAD). La partie supérieure de la 2 montre l'enchaînement des différentes étapes et la partie inférieure illustre un exemple de chaîne de traitement. EXTRACT (cf. figure 2, bloc 1) regroupe les modules conçus pour extraire les données provenant de sources hétérogènes mais aussi les modules permettant de pré-traiter les données (p. ex. nettoyage, cartospondance, compression, etc.). TRANSFORM (cf. figure 2, bloc 2) peut être séparé en deux sous-étapes : (i) la reconstruction qui rassemble les modules permettant de transformer les traces de mouvement en trajectoires brutes et (ii) l'enrichissement qui rassemble les modules permettant de transformer les trajectoires brutes en trajectoires sémantiques. Enfin, LOAD (cf. figure 2, bloc 3) réunit les modules de visualisation et de stockage des résultats des étapes précédentes. Cette plateforme est destinée à des utilisateurs non-informaticiens, son utilisation doit donc être intuitive afin qu'elle soit facile à prendre en main. Ainsi, les types d'entrées et de sorties ainsi que les descriptions de chaque module doivent être documentés. L'ergonomie de la plateforme doit permettre à l'utilisateur de concevoir rapidement une chaîne de traitement idoine sur les données dont il dispose.

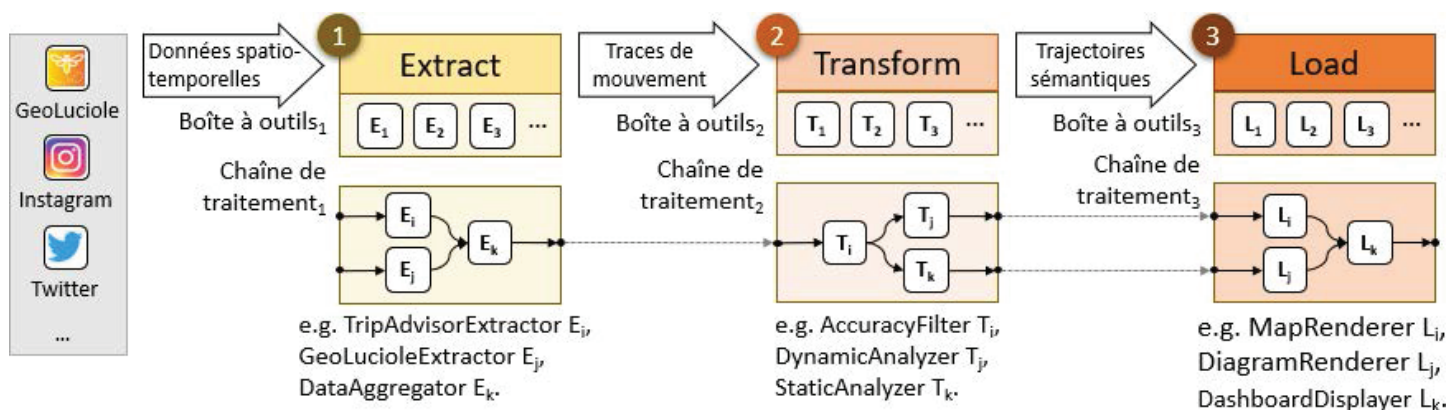


FIGURE 2. Chaîne de traitement des données inspirée des procédures ETL

Nous nous appuyons sur l'article de (Parent et al., 2013) pour modéliser les données qui circuleront à travers la chaîne de traitement. Une trajectoire brute Tb d'identifiant k d'un objet mobile O est définie telle que :

$$Tb = (k, O, \langle (t_i, p_i, v_i) \rangle_{i \leq n})$$

Chaque tuple de la séquence contient la position p_i de l'objet mobile O capturée au temps t_i , où v_i représente l'ensemble possiblement vide des données brutes additionnelles (i.e. vitesse, orientation, précision, etc.).

La trajectoire sémantique Te d'identifiant k d'un objet mobile O enrichie par l'ensemble des interprétations I_{Σ_k} est définie telle que :

$$Te = (k, O, \langle (t_i, p_i, v_i) \rangle_{i \leq n}, \{I_{\Sigma_k}\})$$

Une interprétation est une séquence $\langle (t_{a_j}, t_{b_j}, a_j) \rangle_{j \leq m}$ où chaque tuple est un épisode, avec a_j l'annotation de l'épisode appartenant à l'alphabet Σ_k compris entre les temps t_{a_j} et t_{b_j} .

4.4. Actions réalisées

Cette section illustre l'état actuel de notre travail à travers un exemple de chaîne de traitement (cf. figure 3) créée à l'aide de notre plateforme ETL.

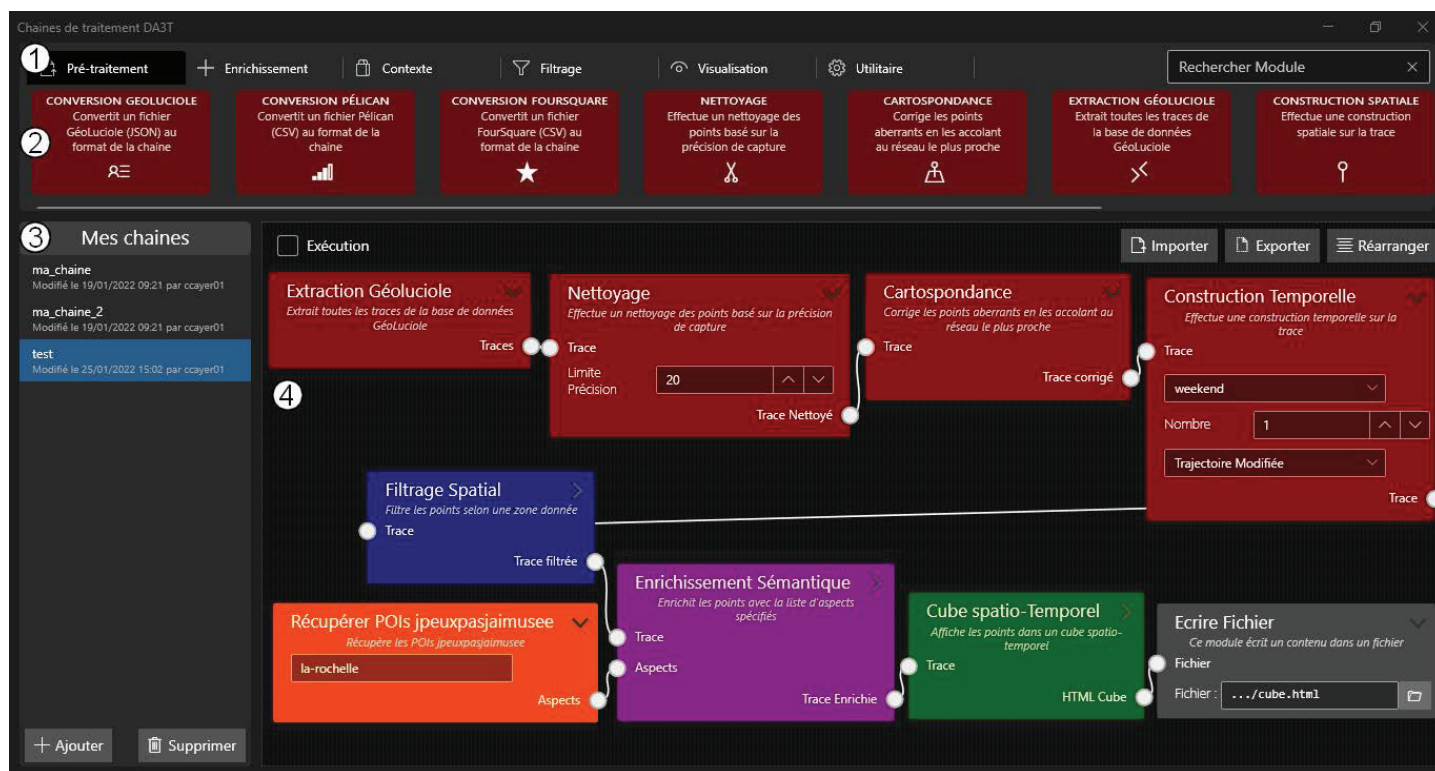


FIGURE 3. Chaîne de traitement de trajectoires de mobilité construite avec la plateforme DA3T

Dans notre plateforme, chaque module est un composant logiciel effectuant un traitement spécifique de bas niveau sur des trajectoires en entrée. Les trajectoires résultantes du traitement sont renvoyées par le premier module et deviennent l'entrée du module suivant de la chaîne de traitement, et ainsi de suite pour aboutir à un résultat final.

Les modules sont classés dans différentes catégories qui sont mises à disposition dans la plateforme (cf. figure 3, 1). Les catégories proposées sont *Pré-traitement* (regroupe les modules d'extraction, de nettoyage, de construction de trajectoires, etc.), *Enrichissement* (regroupe les modules d'enrichissement et de segmentation), *Contexte* (regroupe les modules apportant des données d'enrichissement provenant de différentes sources), *Filtrage* (regroupe les modules de filtrage spatial, temporel, thématique, etc.), *Visualisation* (regroupe les modules permettant de visualiser les données) et *Utilitaire* (regroupe les modules

utilitaires qui permettent d'importer, d'exporter, de prévisualiser le fichier de données). Les modules disponibles sur la plateforme (cf. figure 3, 2) pour créer et exécuter les chaînes de traitement sont stockés sous la forme d'API sur un serveur distant. L'utilisateur peut créer, sauvegarder, importer et exporter des chaînes de traitement personnalisées et paramétrables (cf. figure 3, 3). La chaîne de traitement de notre exemple (cf. figure 3, 4) vise à répondre à la question suivante : Quels points d'intérêt intéressent les touristes le week-end à La Rochelle ?

Pour y répondre, nous utilisons des modules appartenant à six catégories différentes. Les données transitent entre les modules dans un fichier qui instancie notre modèle de trajectoire sémantique (Cayère et al., 2021). Dans un premier temps, les traces brutes sont extraites, nettoyées et les trajectoires sont construites durant la phase de pré-traitement. Un filtrage est réalisé sur ces trajectoires pour ne garder que celles se situant à La Rochelle. Les trajectoires sont ensuite enrichies avec les points d'intérêts de la ville collectés grâce à une API externe. Enfin, la phase de visualisation permet d'afficher les trajectoires sémantiques ainsi créées. Le résultat de la chaîne (c.-à-d. le cube spatio-temporel) est finalement enregistré dans un fichier grâce à un module utilitaire. La plateforme peut être exécutée en flux temps réel, au fur et à mesure qu'une chaîne de traitement est construite, grâce au bouton *Exécution* en haut à gauche de l'espace de travail (cf. figure 3, 4).

4.5. Conclusion et perspectives

Nous avons conçu et développé une plateforme modulaire permettant la création et l'exécution de chaînes de traitement personnalisées et paramétrables. Une chaîne de traitement est construite à partir de modules situés sur un serveur distant qui sont requêtés, un par un, lors de l'exécution de la chaîne. Les données qui transitent entre les modules sont une instanciation de notre modèle de trajectoire sémantique multiple aspects (Cayère et al., 2021).

Dans de futurs travaux, nous proposerons de nouveaux modules concernant, entre autres, le calcul de similarité entre deux trajectoires sémantiques. Nous souhaitons à présent évaluer l'efficacité de la plateforme pour traiter et faciliter l'analyse de données de mobilité. Pour cela, nous menons une expérimentation avec les géographes du projet. Nous mettrons à disposition le code de la plateforme et des modules à la fin du projet DA3T.

5. Un métamodèle pour la population d'ontologie indépendamment du domaine¹⁴

5.1. Contexte

Dans de nombreux domaines scientifiques et techniques, l'utilisation de systèmes experts permet d'offrir à l'humain une assistance pour des tâches allant du diagnostic à la prise de décision. De tels systèmes permettent de résoudre des problèmes complexes en se basant sur l'association d'éléments de connaissance et de règles de déduction simples. L'un des pré-requis pour l'utilisation de tels systèmes est l'existence de bases de connaissances permettant de transcrire l'état de la connaissance d'un domaine dans un format structuré et interprétable par ces systèmes experts. Le développement des ontologies répond en partie à ce besoin de structuration de la connaissance en fournissant à la fois la possibilité de définir les

14. Par Yohann Chasseray

concepts clés d'un domaine et, via des moteurs d'inférence, de raisonner sur ces concepts. Depuis leur définition par Gruber (1993), les ontologies ont été largement développées dans des domaines tels que le domaine médical ou celui de la gestion de crise. Malheureusement ces ontologies ne peuvent pas être utilisées dans des cas d'application concrets sans qu'un expert du domaine ne les complète. Parallèlement, le nombre de sources de données renfermant de la connaissance ne cesse de se multiplier. Malheureusement, la connaissance existant dans ces sources de données (articles et ouvrages scientifiques, bases de données, comptes-rendus d'experts) est souvent non structurée. Il s'agit donc de s'intéresser à l'extraction automatique et générique de la connaissance contenue dans ces sources de données afin de peupler des ontologies, quel que soit le domaine associé.

5.2. *État de l'art*

Aujourd'hui, de nombreuses études s'intéressent à l'automatisation de la population d'ontologies. Les plateformes comme Datalift (Scharffe et al., 2012) ainsi que les modèles comme l'Ontology Definition Metamodel (ODM) développé par l'Object Management Group sont autant d'outils pour traiter différents formats de données structurées et les inter-relier sous un standard commun. Mais majoritairement, les systèmes d'extraction sont dédiés à des formats de données déjà structurés ou à une source unique de données non structurées (Faria et al., 2014). Certains cadres méthodologiques présentent l'avantage de traiter une grande diversité de sources de données (Remolona et al., 2017). Néanmoins, ces approches utilisant de la connaissance du domaine déjà structurée sont limitées pour des domaines dans lesquels ces données n'existent pas encore. Enfin, les systèmes qui ont une approche générique vis-à-vis du domaine altèrent dans le même temps la structure de l'ontologie voire recréent celle-ci à partir des données (Hillairet et al., 2008) et perdent la structure préalablement établie par les experts du domaine. Ainsi, les systèmes de population automatique d'ontologies se classent en deux catégories : soit ils sont spécifiques à un domaine et/ou à une source de données, soit ils altèrent la structure de l'ontologie à peupler.

5.3. *Problématique*

Comme évoqué dans la section précédente, les limites que l'on retrouve dans les systèmes de population d'ontologie sont (i) la dépendance du système vis-à-vis du domaine décrit par l'ontologie, (ii) la limitation des chaînes d'extraction à certaines sources de données et (iii) la modification de la structure de l'ontologie lors du processus de population. L'objectif des travaux présentés ici est de mettre en place un système de population d'ontologie (semi-)automatisé sans altérer cette dernière. Ce système se veut ainsi interopérable et indépendant à la fois des sources de données à traiter et de l'ontologie ciblée.

5.4. *Actions réalisées*

Pour faire le pont de manière générique entre les données hétérogènes et les domaines métiers pour lesquels des ontologies sont définies, un métamodèle générique (Cf. Figure 4) pour l'extraction de données a été construit. D'une part, ce métamodèle est défini dans un souci de généricité de manière à englober un large spectre de sources de données. D'autre part, la structure du métamodèle défini se rapproche de la structure généralement adoptée par les ontologies. De cette manière, il est aisé de dériver des alignements entre ce métamodèle et l'ontologie cible.

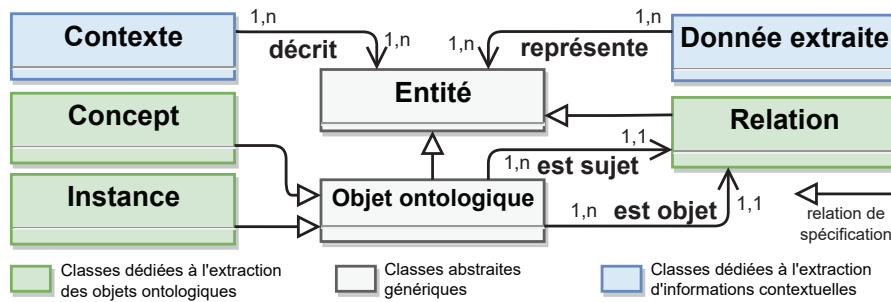


FIGURE 4. Métamodèle générique pour l'extraction de données hétérogènes.

Dans le métamodèle présenté, la classe *Entité* représente les objets extraits des données et pouvant être soit associés à des objets déjà présents dans l'ontologie cible, soit ajoutés à cette ontologie cible. La classe *Objet ontologique* représente les objets constitutifs de la connaissance (*Concept* ou *Instance*) que l'on retrouve habituellement dans une ontologie. La classe *Relation* identifie les relations détectées dans les données entre deux *Objets ontologiques*. Il n'est pas utile de représenter l'ensemble des données brutes dans ce métamodèle. Néanmoins, la classe *Donnée extraite* embarque une partie de cette donnée brute (phrase contenant une *Entité* par exemple) et permet de garder une trace de la donnée dont est issue une *Entité*. Enfin, la classe *Contexte* permet d'embarquer des éléments de contexte pour décrire plus précisément les *Entités* extraites. Ces derniers peuvent être de nature très variée en fonction du type de données exploitées (termes co-occurents, titre de figure, méta-données).

Un cadre méthodologique pour la population générique d'ontologies (Cf. Figure 5) est construit autour du métamodèle présenté précédemment. Dans la lignée de travaux antérieurs sur la représentation des ontologies (Gašević et al., 2009), ce cadre s'appuie sur les principes de l'Ingénierie Dirigée par les Modèles (Kent, 2002). Le métamodèle défini précédemment y est utilisé comme un pivot entre les données et l'ontologie cible que l'on souhaite peupler. Cette ontologie peut varier en fonction du domaine qu'elle décrit. On définit alors la base de connaissances comme la réunion des instances et de l'ontologie cible que viennent alimenter ces instances.

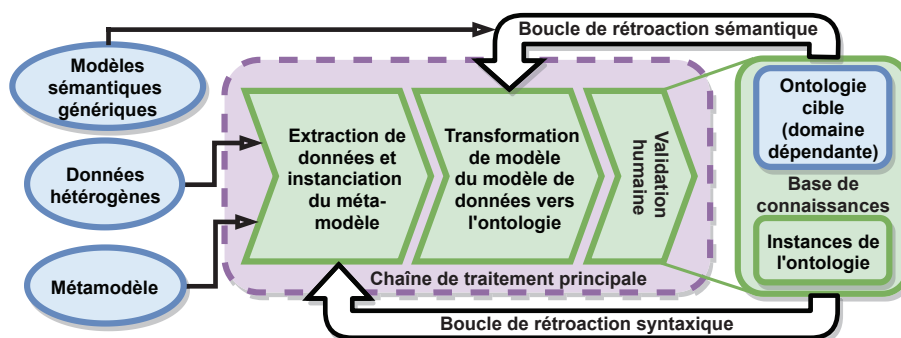


FIGURE 5. Cadre méthodologique pour la population d'ontologies.

Le cadre se structure en trois chaînes de traitement qui sont la *chaîne de traitement principale*, la *boucle de rétroaction sémantique* et la *boucle de rétroaction syntaxique*. La *chaîne de traitement principale* traite la donnée en deux étapes. La première étape extrait l'information de la donnée brute et l'organise en connaissance en s'appuyant sur les concepts du métamodèle. Cela peut être effectué avec des méthodes d'extraction classiques (extraction par règles, traitement d'image, exploration de bases de données par exemple). De cette première étape naît un modèle de données qui contient des candidats potentiels (*Entités*) pour l'instanciation de l'ontologie. La deuxième étape est une transformation de modèle du modèle

de données vers l'ontologie cible. Cette transformation de modèle se base sur des règles d'alignement définies au niveau du métamodèle et exprimées dans une logique d'ordre 1. Elle permet par exemple de transformer des instances issues du métamodèle en instances de l'ontologie. Une étape de validation humaine est incluse à la chaîne de traitement principale afin de valider ou d'invalider les appariements effectués. Cette étape de validation a notamment pour rôle de prévenir l'introduction d'instances incohérentes, qui pourraient mettre en péril la cohérence de la base de connaissances. Les deux boucles de rétroaction permettent de rendre la méthode itérative et de limiter ainsi l'intervention humaine au fil des itérations. De nouvelles règles d'extraction ainsi que de nouveaux alignements peuvent de cette manière être déduits à partir respectivement des boucles de rétroaction syntaxiques et sémantiques.

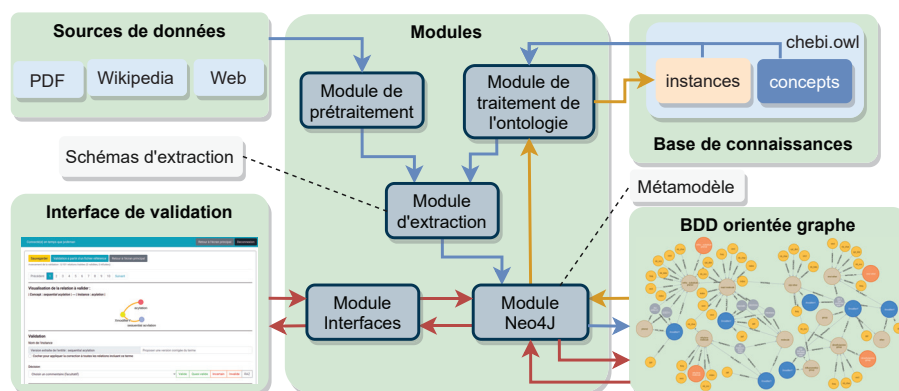


FIGURE 6. Architecture globale du prototype logiciel pour l'extraction de relations à partir de données textuelles.

Le cadre méthodologique explicité ci-dessus a été implémenté au sein d'une preuve de concept logicielle. La Figure 6 illustre la structure globale du prototype logiciel mis en place. Ce dernier met en jeu différents modules interagissant pour réaliser les processus d'extraction (fléchage bleu), de validation (fléchage rouge) puis de transformation (fléchage orange) des informations contenues au sein de données textuelles.

Dans ce prototype, les modèles issus du métamodèle pivot et construits à partir des données textuelles sont représentés et stockés sous la forme de bases de données (BDD) orientées graphes. Une application web permet quand à elle de soumettre les relations extraites au regard d'un expert. En dépit de son caractère facultatif, cette étape fournit la possibilité de valider ou d'invalider certaines des relations extraites et par ce biais, d'estimer la précision des règles d'extraction utilisées.

L'utilisation du prototype associé à trois schémas d'extraction génériques a permis de réaliser la détection de relations à partir (i) d'ontologies touchant à des domaines divers comme la chimie (Hastings et al., 2016) ou la gestion de crise (Benaben et al., 2020) et (ii) de sources de données variées (documents PDF, pages web, articles de l'encyclopédie Wikipedia).

5.5. Actions futures

Si elles ne sont pas explicitement représentées sur la Figure 6, les boucles de rétroaction syntaxique et sémantique sont également intégrées au prototype logiciel. L'efficacité de la boucle de rétroaction sémantique est néanmoins fortement dépendante de la capacité du système à extraire des entités d'intérêt vis-à-vis du domaine des données non structurées. Hors, l'extraction de ces entités à partir d'un document brut se révèle généralement très bruitée. L'investigation de méthodes non supervisées pour l'extraction

de candidats à l’instanciation est donc une piste qui sera explorée afin de rendre le cadre méthodologique complètement autonome.

Une seconde voie à approfondir concerne l’enrichissement des schémas d’extraction afin d’étendre l’extraction à d’autres types de relations, plus horizontales (cause-conséquence, tout-partie, relations décrites dans l’ontologie et qui sont propres au domaine, par exemple).

Enfin, afin d’assurer le maintien de la cohérence de la base de connaissances, une étape de vérification de cette cohérence – s’appuyant sur les règles définies dans l’ontologie – doit également être envisagée comme une priorité des développements futurs de la méthodologie. L’existence de cette étape de vérification permettrait alors d’établir la certitude que chaque ajout d’une nouvelle instance n’entre pas en contradiction avec l’état courant de la base de connaissances.

6. Conclusion et remerciements ¹⁵

Le bureau d’INFORSID a toute ma reconnaissance pour l’honneur qui m’a été fait de me confier ce 10^e Forum JCJC, gérer les contributions des doctorants, la préparation des actes, l’organisation du Forum à distance ainsi que d’une singulière et fertile session « dragons et chevaliers » pendant le congrès INFORSID 2021 et enfin la coordination de cet article. Le lecteur aura pu apprécier le caractère temporel, applicatif, collecte et analyse de traces et peuplement de bases de connaissances des travaux présentés dans cet article. Il y a là un paysage de la recherche française en systèmes d’information clairement symptomatique de l’actualité que nous traversons alors : une recherche dynamique d’interactions traçables, recommandables et explicables.

Bibliographie

Andriamampianina, L., Ravat, F., Song, J., and Vallès-Parlangeau, N. (2021). Towards an Efficient Approach to Manage Graph Data Evolution : Conceptual Modelling and Experimental Assessments. In Cherfi, S., Perini, A., and Nurcan, S., editors, Research Challenges in Information Science, pages 471–488. Springer International Publishing.

Ansoff, H. I. (1975). Managing strategic surprise by response to weak signals. California management review, 18(2) :21–33.

Benaben, F., Fertier, A., Montarnal, A., Mu, W., Jiang, Z., Truptil, S., Barthe-Delanoë, A.-M., Lauras, M., Mace-Ramete, G., Wang, T., et al. (2020). An ai framework and a metamodel for collaborative situations : Application to crisis management contexts. Journal of Contingencies and Crisis Management, 28(3) :291–306.

Cayèré, C., Sallaberry, C., Faucher, C., Bessagnet, M.-N., Roose, P., Masson, M., and Richard, J. (2021). Multi-Level and Multiple Aspect Semantic Trajectory Model : Application to the Tourism Domain. ISPRS International Journal of Geo-Information, 10(9) :592.

Charbey, R. (2018). Sociabilités en ligne, usages et réseaux. PhD thesis, Paris, ENST.

15. Par Pierre-Emmanuel Arduin, organisateur du Forum JCJC 2020

- Desmier, E., Plantevit, M., Robardet, C., and Boulicaut, J.-F. (2012). Cohesive co-evolution patterns in dynamic attributed graphs. In International Conference on Discovery Science, pages 110–124. Springer.
- Fard, A., Abdolrashidi, A., Ramaswamy, L., and Miller, J. (2012). Towards Efficient Query Processing on Massive Time-Evolving Graphs. In Proceedings of the 8th IEEE International Conference on Collaborative Computing : Networking, Applications and Worksharing, pages 567–574. IEEE.
- Faria, C., Serra, I., and Girardi, R. (2014). A domain-independent process for automatic ontology population from text. Science of Computer Programming, 95 :26–43.
- Flouvat, F. (2019). Extraction de motifs spatio-temporels : co-localisations, séquences et graphes dynamiques attribués. PhD thesis, Université de la Nouvelle-Calédonie.
- Gašević, D., Djuric, D., and Devedžic, V. (2009). Model driven engineering and ontology development. Springer Science & Business Media.
- Glasscock, D. (2016). What is a graphon ?
- Goldin, D. Q. and Kanellakis, P. C. (1995). On similarity queries for time-series data : constraint specification and implementation. In International Conference on Principles and Practice of Constraint Programming, pages 137–153. Springer.
- Gruber, T. R. (1993). A translation approach to portable ontology specifications. Knowledge acquisition, 5(2) :199–220.
- Hastings, J., Owen, G., Dekker, A., Ennis, M., Kale, N., Muthukrishnan, V., Turner, S., Swainston, N., Mendes, P., and Steinbeck, C. (2016). Chebi in 2016 : Improved services and an expanding collection of metabolites. Nucleic acids research, 44(D1) :D1214–D1219.
- Hillairet, G., Bertrand, F., and Lafaye, J. Y. (2008). Un processus dirigé par les modèles pour la création de bases de connaissance ontologiques. IDM.
- Hočevár, T. and Demšar, J. (2014). A combinatorial approach to graphlet counting. Bioinformatics, 30(4) :559–565.
- Kent, S. (2002). Model driven engineering. In International Conference on Integrated Formal Methods, pages 286–298. Springer.
- Kostakos, V. (2009). Temporal graphs. Physica A : Statistical Mechanics and its Applications, 388(6) :1007–1023.
- Latapy, M., Viard, T., and Magnien, C. (2017). Stream Graphs and Link Streams for the Modeling of Interactions over Time. arXiv :1710.04073 [physics, stat].
- Lesca, Humbert et Blanco, S. (2002). Contribution à la capacité d’anticipation des entreprises par la sensibilisation aux signaux faibles. In 6è Congrès international francophone sur la PME.
- Mille, A. (2013). De la trace à la connaissance à l’ère du Web. Introduction au dossier. Intellectica, 59(1) :7–28.

- Newson, P. and Krumm, J. (2009). Hidden markov map matching through noise and sparseness. In Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems, pages 336–343.
- Parent, C., Spaccapietra, S., Renso, C., Andrienko, G., Andrienko, N., Bogorny, V., Damiani, M. L., Gkoulalas-Divanis, A., Macedo, J., Pelekis, N., et al. (2013). Semantic trajectories modeling and analysis. ACM Computing Surveys (CSUR), 45(4) :1–32.
- Pržulj, N., Corneil, D. G., and Jurisica, I. (2004). Modeling interactome : scale-free or geometric? Bioinformatics, 20(18) :3508–3515.
- Ranta, R., Louis-Dorr, V., Heinrich, C., Wolf, D., and Guillemin, F. (2003). Débruitage par ondelettes et segmentation de signaux non-stationnaires : réinterprétation d'un algorithme itératif et application à la phonoentérographie. Traitement du signal, 20(2) :119–135.
- Remolona, M. F. M., Conway, M. F., Balasubramanian, S., Fan, L., Feng, Z., Gu, T., Kim, H., Nirantar, P. M., Panda, S., Ranabothu, N. R., et al. (2017). Hybrid ontology-learning materials engineering system for pharmaceutical products : Multi-label entity recognition and concept detection. Computers & Chemical Engineering, 107 :49–60.
- Ribeiro, B., Perra, N., and Baronchelli, A. (2013). Quantifying the effect of temporal resolution on time-varying networks. Scientific Reports, 3(1) :3006.
- Ripley, B. D. and Venables, W. (2002). Modern applied statistics with S. springer.
- Rousseau, P., Camara, D., and Kotzinos, D. (2021). Weak signal detection and identification in large data sets : a review of methods and applications.
- Scharffe, F., Bihanic, L., Képéklian, G., Atemezeng, G., Troncy, R., Cotton, F., Gandon, F., Villata, S., Euzenat, J., Fan, Z., et al. (2012). Enabling linked data publication with the datalift platform. In Workshops at the Twenty-Sixth AAAI Conference on Artificial Intelligence.
- Spaccapietra, S. and Parent, C. (2011). Adding Meaning to Your Steps (Keynote Paper). In Jeusfeld, M., Delcambre, L., and Ling, T.-W., editors, Conceptual Modeling ? ER 2011, Lecture Notes in Computer Science, pages 13–31, Berlin, Heidelberg. Springer.
- Zaki, A., Attia, M., Hegazy, D., and Amin, S. (2016). Comprehensive Survey on Dynamic Graph Models. International Journal of Advanced Computer Science and Applications, 7(2).
- Zhao, A., Liu, G., Zheng, B., Zhao, Y., and Zheng, K. (2020). Temporal paths discovery with multiple constraints in attributed dynamic graphs. World Wide Web, 23 :313–336.