Une approche pour évaluer la complétude de données RDF

Fayçal Hamdi, Samira Si-said Cherfi

Laboratoire Cédric, Conservatoire national des arts et métiers Paris, France
{faycal.hamdi,samira.cherfi}@cnam.fr

RÉSUMÉ. *Nous assistons depuis quelques années à une prolifération de données du web. Ceci a accéléré le développement d'application et de logiciels fondés sur l'exploitation et l'analyse des données. Il devient donc urgent de proposer des techniques et des méthodes pour l'évaluation et l'assurance de la qualité de ces données. La qualité est un concept multidimensionnel qui repose sur une variété de dimensions. Une des dimensions critiques pour la prise de décision est la complétude. Cette dimension est reconnue comme difficile à évaluer puisqu'elle requiert souvent l'existence d'une norme d'excellence ou un schéma de référence validé et agréé qui servira de référence universelle à cette complétude. Cependant un tel référentiel est rarement disponible voire inexistant dans la pratique. Dans le présent article, nous proposons une approche pour l'évaluation de la complétude de sources de données RDF (Resource Description Framework). L'approche est un processus en deux étapes. La première étape, que nous pouvons qualifier de fouille de schéma, consiste à extraire un schéma probable pour la description d'une source de données. Ce schéma est ensuite exploité lors de la deuxième étape du processus pour l'évaluation de la complétude. L'article présente, à la fois des concepts théoriques sur lesquels repose l'approche, mais aussi des expérimentations basées sur des données RDF réelles.*

ABSTRACT. *With the development of data based applications, data quality becomes a burning issue in the context of the Web of Data. Organizations as well as researchers need suitable methods and techniques to help ensuring web data quality along the whole process, from data transformation and publication to data querying and exploitation. Among quality dimensions, completeness is recognized as difficult to evaluate, as it often relies on gold standards and/or a reference schema that are neither always available nor realistic from a practical point of view. In this paper, we propose an approach for the assessment of RDF data completeness. The proposed solution consists, first, on inferring a schema using a frequent itemset mining approach, and second, on measuring the completeness regarding the inferred schema. The paper presents both theoretical background and experimental results performed on real-world RDF datasets.*

MOTS-CLÉS : *web de données, qualité des données RDF, complétude, évaluation de la qualité.*

KEYWORDS: *linked Data, RDF data quality, completeness, quality evaluation.*

DOI:10.3166/ISI.21.3.31-52

1. Introduction

Nous assistons depuis quelques années au développement rapide et croissant des technologies du web. Ce phénomène auquel s'ajoute la disponibilité en ligne de gros volumes de données au format numérique a fait de la donnée un atout stratégique pour de nombreuses organisations et une source de revenu non négligeable pour les entreprises. Ce phénomène est amplifié par les efforts collaboratifs, fournis par des communautés diverses et variées couvrant des sociétés civiles, des usagers ou des experts et qui ont contribué à la disponibilité de volumes de données de plus en plus importants. Ces données sont publiées de manière continue et dans un format qui se prête au traitement automatique. Ceci a encouragé l'émergence de nouvelles techniques, technologies, pratiques et méthodes permettant d'exploiter ces données pour des usages précis, commerciaux, gouvernementaux ou scientifiques. Par conséquent, ces données, qui sont accessibles par le plus grand nombre d'usagers de divers horizons, ont un impact certain sur les processus de prise de décision.

Selon un rapport du McKinsey Global Institute (Institute *et al.*, 2012), en utilisant les technologies des réseaux sociaux, les entreprises pourraient augmenter leurs profits de soixante pour cent. Dans le secteur de la recherche, l'exploration des liens sémantiques entre les sources de données a rendu possible des analyses qui ne l'étaient pas par le passé, et a permis dans bien des cas l'émergence de résultats intéressants. Des travaux, tels que ceux présentés dans (Samwald *et al.*, 2011), exploitent des données sur des études cliniques, des études sur l'expression des gènes ou des études sur les médicaments issues de sources comme ClinicalTrials.gov ou LODD [1]. Cependant, comme la publication des données ne nécessite pas de compétence ou d'expertise particulière, la disponibilité des données ne garantit pas toujours leur utilité. Ceci signifie que les données auxquelles on peut avoir accès ne sont pas toujours d'aussi bonne qualité que l'on pourrait supposer. Ceci peut mener soit à une faible valeur ajoutée de ces données, soit à une faible fiabilité des conclusions dérivées.

La qualité de l'information et des données a été longtemps un sujet d'intérêt surtout dans le contexte des bases de données relationnelles (Wang, Strong, 1996 ; Lee *et al.*, 2002). Elle constitue, cependant, une recherche émergente et un réel défi dans le contexte du web de données [2] (LOD). En effet, le web de données utilise des URI qui permettent l'identification uniforme de ressources sur le web. Ceci facilite la publication et la navigation à travers ces données sans en garantir ni la qualité ni l'utilité (Bechhofer *et al.*, 2013). Il est par conséquent nécessaire de développer des approches et des techniques adaptées pour assurer la qualité du web de données.

La qualité du web de données fait l'objet d'un intérêt récent et les contributions vont dans deux directions principales. La première s'appuie essentiellement sur la contribution des usagers qui se trouvent chargés de qualifier la qualité des données qu'ils publient. Cette approche est subjective car elle ne s'appuie que sur le jugement

1. http://esw.w3.org/topic/HCLSIG/LODD
2. http://lod-cloud.net

des acteurs qui publient ces données. La seconde, plus objective, est essentiellement restreinte à l'évaluation de la provenance car elle s'appuie sur des mesures et des outils.

Dans cet article nous adressons le problème de la complétude des données comme dimension de la qualité des données et proposons un moyen de l'évaluer. La complétude des données est en effet une caractéristique essentielle exigée lorsqu'il s'agit de processus de prise de décision. Ainsi, une information précise de la complétude d'une source permet de mesurer ou au moins de prendre en considération le degré de fiabilité que l'on souhaite lui accorder de manière objective.

La complétude des données renferme deux facettes. La première permet de savoir si toutes les données qui auraient pu être enregistrées sont disponibles. Dans ce cas on parle de complétude structurelle qui est opposée à la complétude de contenu (Ballou, Pazer, 2003). La complétude structurelle fait référence à un schéma de référence auquel les donner doivent se conformer. La complétude de contenu est plus difficile car, en plus du schéma, cette dernière exige les valeurs des données sous la forme d'une source de données de référence supposée être consensuelle et complète. La manière traditionnelle, issue des bases de données relationnelles, de mesurer la complétude structurelle s'appuie sur le taux de valeurs manquantes. Dans ces approches, on ne fait d'ailleurs pas de distinction entre l'absence de valeurs temporaires et l'absence pour inapplicabilité, et encore moins entre le fait que certaines valeurs sont essentielles et que d'autres le sont moins. Ceci implique que toutes les propriétés du schéma de référence sont supposées être d'égale importance. Dans le contexte du web de données, la disponibilité d'un tel schéma ainsi que les différentes hypothèses sur ses propriétés ne sont pas réalistes.

Prenons par exemple une source de données issue d'un effort collaboratif de personnes ou de communautés, ayant des centres d'intérêts et des points de vue divers, comme c'est le cas de DBpedia ou Freebase. Pour de telles sources, nous sommes loin de la démarche traditionnelle de population d'une base de données, où le schéma est prédéfini. En effet, dans ce cas aussi bien les données que le schéma qui les décrit évoluent constamment. Le défi est alors de proposer une méthode pour l'évaluation de la complétude qui tienne compte de l'absence d'un schéma de référence pour les données. Dans ces conditions, nous adoptons dans la suite de cet article la définition de la complétude proche de la notion de complétude de schéma de (Pipino *et al.*, 2002) ou de celle de densité des données décrites dans (Naumann *et al.*, 2004). Par conséquent, l'incomplétude est liée à l'absence de données ; le moins une source a de valeurs absentes le plus elle sera dite complète. Il est donc nécessaire de disposer d'un schéma par rapport auquel une donnée sera dite manquante. Un tel schéma sera calculé par notre approche puisqu'il est absent de la source de données.

L'objectif de l'article est de proposer une approche pour l'évaluation de la complétude d'une source de données RDF. Notre approche s'appuie sur un schéma qui sera inféré depuis les données. Ce schéma est supposé représenter un état ayant un sens pour la source de données considérée. Pour résumer, le papier présente les contributions suivantes:

1. Nous utilisons une approche de fouille de données pour inférer un schéma.

2. Nous introduisons une nouvelle approche pour l'évaluation de la complétude s'appuyant à la fois sur le schéma inféré et sur les données.

3. Nous présentons les résultats d'expérimentations que nous avons menées sur des données réelles. Ces expérimentations nous ont permis d'analyser aussi bien la relation entre les mesures de complétude et le nombre de propriétés composant le schéma inféré, mais aussi la robustesse de notre mesure de complétude lorsque la taille de la source de données varie.

L'article suit la structure suivante : la section 2 présente un exemple qui permet d'illustrer le problème que nous adressons ; la section 3 fournit une définition formelle du problème ; la section 4 décrit en détail l'approche d'inférence du schéma ainsi que la méthode d'évaluation de la complétude; la section 5 détaille et analyse les expérimentations menées; la section 6 résume la littérature existant sur le sujet et enfin la section 7 présente un ensemble de conclusions sur le travail et des directions futures.

2. Illustration par l'exemple

Nous présentons dans cette section l'idée principale à travers un exemple dont l'objectif est de montrer les difficultés et les problèmes à résoudre lorsqu'il s'agit d'évaluer la complétude d'une source de données. Prenons l'exemple de l'ensemble des scientifiques décrits dans DBpedia. Nous souhaitons savoir si, en formulant une requête sur un scientifique particulier, l'information qui est retournée par DBpedia est complète. Nous souhaitons également savoir si les scientifiques dans cette source sont bien décrits et si leurs informations sont complètement renseignées.

Une façon de répondre à cette question est de considérer les propriétés utilisées pour la description d'un scientifique et de les comparer aux propriétés du schéma que prévoit la source pour décrire un scientifique (appelé ontologie). Ceci revient à la vision de la complétude basée sur l'absence de valeurs. Dans DBpedia, la classe *Scientist*[3] possède des propriétés parmi lesquelles on trouve le directeur de recherche (*doctoralAdvisor*). Ces propriétés ne sont pas les seules utilisées pour la description d'un scientifique. Néamoins, nous constatons que certaines propriétés telles que la date de naissance (*birthdate*) n'apparaissent pas dans cette liste. En effet, la classe *Scientist* hérite de sa super classe *Person*. Sa description peut théoriquement utiliser des propriétés de *Person* ainsi que celles de toutes ces super-classes. Par conséquent, si l'on veut obtenir la liste exhaustive de toutes les propriétés pouvant servir à la description de *Scientist* nous devrions calculer l'union des propriétés de la classe *Scientist*

3. http://mappings.dbpedia.org/server/ontology/classes/

et de toutes les classes dont elle hérite. Pour notre exemple, le schéma de référence que nous appellerons $Scientist\_Schema$ pourrait être calculé comme suit :

$$Scientist\_Schema = \{\textit{Propriétés de Scientist}\} \cup$$
$$\{\textit{Propriétés de Person}\} \cup \{\textit{Propriétés de Agent}\} \cup$$
$$\{\textit{Propriétés de Thing}\}$$
$$\text{Tel que: } Scientist \sqsubseteq Person \sqsubseteq Agent \sqsubseteq Thing$$

Par conséquent, la complétude d'un scientifique (ex. *Albert_Einstein*) pourrait être calculée comme le ratio entre le nombre de propriétés réellement utilisées dans sa description sur le nombre total de propriétés dans $Scientist\_Schema$. Pour la source de données DBpedia, en appliquant une simple requête SPARQL [4], nous pouvons obtenir le nombre de propriétés de $Scientist\_Schema$ qui est de 664 (propriétés). Ceci nous conduit à une valeur de la complétude pour la description de *Albert_Einstein* résultant du calcul suivant :

$$Comp(Albert\_Einstein) = \frac{|\textit{Propriétés de Albert\_Einstein}|}{|Scientist\_Schema|}$$
$$= \frac{21}{664} = 4,21\%$$

Cette valeur paraît faible surtout si l'on considère la description de *Albert_Einstein* qui, du fait de sa renommée, est malgré tout bien décrit dans DBpedia. Il est donc légitime de s'interroger sur la pertinence de la mesure de complétude précédemment décrite. En réalité, bien que la classe *Scientist* hérite de plusieurs autres classes, toutes les propriétés héritées ne sont pas réellement utilisées et ne sont pas toujours pertinentes.

A titre d'exemple, *weapon* (arme) fait partie de $Scientist\_Schema$ mais est loin d'avoir du sens pour la description d'*Albert_Einstein* (dans DBpedia, cette propriété n'est d'ailleurs utilisée dans la description d'aucun scientifique bien que faisant partie du schéma de $Scientist$). Cette propriété, dans la description de *Scientist*, ne peut par conséquent pas être considérée comme étant aussi importante que des propriétés telles que le nom ou l'université d'appartenance (*name* et *university* resp.). C'est également le cas de nombreuses autres propriétés appartenant à $Scientist\_Schema$. D'ailleurs, un scientifique dans DBpedia est décrit avec 21 propriétés de type A-Box [5] auxquelles s'ajoutent 66 T-Box en plus d'autres propriétés externes, ce qu'on pourrait finalement qualifier de bonne description fournissant une information assez complète et utile. Donc déclarer que la description de *Albert_Einstein* n'est complète qu'à $4{,}21$ % contredit cet état de fait. La valeur de complétude, calculée par la formule précédente, donne d'ailleurs pour les 1000 premiers scientifiques une complétude de $1{,}37$ %, bien

4. Exécutée sur : http://dbpedia.org/sparql
5. http://dbpedia.org/resource/Albert_Einstein

inférieure à celle obtenue pour le « célèbre » *Albert_Einstein*. Ceci est expliqué par le fait que, comme la description dans DBpedia est le résultat d'un effort volontaire et non celui d'une initiative institutionnelle et rigoureuse, les sujets moins connus du grand public font l'objet de moins d'efforts de description, impactant ainsi le nombre de propriétés. La conséquence directe est que, sur l'ensemble de la source DBpedia comprenant 18 233 scientifiques [6], la valeur de complétude baisse considérablement du fait du nombre de scientifiques peu connus du grand public.

Pour résumer, on peut dire que la complétude ainsi calculée n'est pas pertinente, au regard de la description réelle des données dans la source, puisqu'elle ne reflète pas la réalité.

Afin de résoudre ce problème, il paraît nécessaire de s'appuyer sur la réalité des données. Ceci nécessite l'exploration des instances et des propriétés qui les décrivent en analysant le degré d'utilisation de ces propriétés ainsi que leurs importances. Nous proposons dans cet article une solution qui exploite les techniques de fouille de données. Elle permettra d'extraire à partir d'instances, issues de la même classe, un schéma correspondant au pattern constitué des propriétés les plus utilisées dans la description de ces instances. Une mesure de complétude, s'appuyant sur ce schéma, est ensuite proposée.

3. Problématique

L'évaluation de la complétude ne vise pas à fournir une valeur absolue de la complétude, mais de calculer la valeur qui tienne à la fois compte du contexte d'usage et des données dont on dispose. Dans notre cas, le contexte est celui d'un ensemble de données décrivant une catégorie ou un ensemble de catégories telles que *Actor* ou *Organization*. Le tableau 1 illustre un extrait des instances de la catégorie *Actor* sous la forme de triplets RDF issus de la source de données DBpedia.

Tableau 1. Extrait de la catégorie Actor de DBpedia

Subject	Predicate	Object
Ben_Affleck	birthDate	1972-01-01
Ben_Affleck	residence	Los Angeles
Angelina_Jolie	birthDate	1975-06-04
Angelina_Jolie	citizenship	United_States
Adam_West	birthDate	1928-09-19
Adam_West	citizenship	American
Adam_West	residence	Ketchum,_Idaho

Chaque catégorie est décrite par un ensemble de propriétés (prédicats) et une instance de cette catégorie pourrait avoir des valeurs pour un ensemble constitué de toutes

6. http://wiki.dbpedia.org/Datasets/DatasetStatistics (statistics of the DBpedia 2014 version for the english language)

ou d'une partie de ces propriétés. Cet ensemble est appelé transaction. Le tableau 2 illustre l'ensemble des transactions composées à partir des triplets présentés dans le tableau 1.

Tableau 2. Les transactions correspondant aux triplets

Instance	Transaction
Ben_Affleck	birthDate, residence
Angelina_Jolie	birthDate, citizenship
Adam_West	birthDate, citizenship, residence

De manière plus formelle, soit l'ensemble de données $\mathcal{D}$ défini comme un triplet (C, I_C, P), où C représente l'ensemble des catégories (ex. *Actor*, *City*), I_C est l'ensemble des instances des catégories de C (ex. *Ben_Affleck* est instance de la catégorie *Actor*), et $P = \{p_1, p_2, ..., p_n\}$ est l'ensemble des propriétés (ex. *residence(Person,Place)*).

Soit $\mathcal{T} = \{t_1, t_2, ..., t_m\}$ l'ensemble des transactions tel que $\forall k, 1 \leq k \leq m$: $t_k \subseteq P$ est un vecteur de transactions composé à partir de P, et $E(t_k)$ l'ensemble des items d'une transaction t_k. Chaque transaction est l'ensemble des propriétés utilisées pour la description des instances de l'ensemble $\mathcal{I}' = \{i_1, i_2, ..., i_m\}$ avec $\mathcal{I}' \subseteq I_C$ (ex. propriétés utilisées pour décrire l'instance *Ben_Affleck* sont : *birthDate* et *residence*). Nous notons $\mathcal{CP}$ la complétude de $\mathcal{I}'$ au regard des propriétés utilisées dans la description de ses instances.

Définition du problème Étant donné un ensemble $\mathcal{D}$, un ensemble d'instances $\mathcal{I}'$, et un ensemble de transactions $\mathcal{T}$ construit à partir de $\mathcal{I}'$, l'objectif est de calculer la complétude $\mathcal{CP}$ de $\mathcal{I}'$.

4. Extraction du schéma d'une source de données RDF

La mesure de la complétude, au niveau des données, s'appuie sur les valeurs manquantes (Pipino *et al*., 2002). Cette vision requiert, au vu des spécificités déjà citées du web de données, l'extraction du schéma à partir de la source de données elle-même. Nous avons montré dans la section 2 que le schéma qui résulte de l'ontologie sous-jacente à une source de données telle que DBpedia n'est pas pertinent. La raison essentielle vient du fait qu'il existe deux sémantiques distinctes pour l'absence de valeurs qui sont l'absence temporaire et l'absence pour inapplicabilité (Codd, 1986). L'absence pour inapplicabilité survient lorsque la propriété ne s'applique pas ou n'a pas de sens pour l'instance ou l'objet en question. Ceci nous permet de conclure que toutes les propriétés décrivant les instances d'une source ne sont pas d'égales importances.

Pour intégrer cet aspect, nous proposons une approche permettant de calculer la complétude d'une source de données. Nous avons choisi de le formuler le problème comme un problème de fouille du schéma le plus vraisemblable. Par conséquent, le

processus de fouille prendra en entrée les données réelles de la source. Ce processus comprend deux étapes :

1. **Extraction du schéma de la source :** Soit la source de données $\mathcal{D}$, nous représentons tout d'abord l'ensemble des propriétés qui décrivent les instances $\mathcal{D}$, comme un vecteur de transactions. Nous appliquons ensuite l'algorithme FP-growth (Han *et al.*, 2000 ; 2004) afin d'extraire les itemsets les plus fréquents (nous avons choisi l'algorithme FP-growth pour des raisons de performance des calculs. Un autre algorithme aurait tout à fait pu être utilisé). Nous ne retenons au final qu'un sous-ensemble, dit « Maximal » de ces ensembles les plus fréquents (Jr., 1998 ; Gouda, Zaki, 2001 ; Grahne, Zhu, 2003). Ce choix est motivé, d'une part par le fait que nous accordons une importance à *l'expression* du pattern fréquent et, d'autre part parce que le nombre de patterns les plus fréquents peut être exponentiel lorsque la taille du vecteur de transactions est importante (voir section 4.1 pour plus de détails).

2. **Évaluation de la complétude :** Une fois que l'itemset le plus fréquent maximal $\mathcal{MFP}$ aura été généré, nous exploitons la fréquence d'occurrence des items (propriétés) dans $\mathcal{MFP}$, pour assigner à chacun de ses items un poids qui reflète l'importance de la propriété sous-jacente dans la description des instances. Les poids ainsi définis serviront à l'évaluation de la complétude de chaque transaction (en tenant compte de la présence ou de l'absence des propriétés dans la transaction) et par la suite la complétude de l'ensemble de la source de données.

Nous détaillons ci-après, chacune des étapes.

4.1. Extraction du schéma de la source

Soit $\mathcal{D}(C, I_C, P)$ une source de données $\mathcal{I}'$ un sous-ensemble d'instances avec $\mathcal{I}' \subseteq I_C$. Initialement, $\mathcal{T} = \phi$, $\mathcal{MFP} = \phi$. Pour chaque $i \in \mathcal{I}'$ nous générons une transaction t. En effet, chaque instance i est liée à des valeurs (des ressources ou des littéraux) à travers un ensemble de propriétés. Par conséquent, une transaction t_k d'une instance i_k est un ensemble de propriétés tel que $t_k \subseteq P$. Les transactions générées ainsi pour toutes les instances de $\mathcal{I}'$ sont alors ajoutées à l'ensemble $\mathcal{T}$.

EXEMPLE 1. — Considérons l'exemple du tableau 1, soit $\mathcal{I}'$ un sous-ensemble d'instances tel que : $\mathcal{I}' = \{$*Ben_Affleck, Angelina_Jolie, Adam_West*$\}$. L'ensemble des transactions $\mathcal{T}$ serait alors :

$$\mathcal{T} = \{\{birthDate, residence\}, \{birthDate, citizenship\}, \{birthDate, citizenship, residence\}\}$$

□

L'objectif est alors de calculer les ensembles fréquents de propriétés co-occurrentes, dits patterns fréquents $\mathcal{FP}$, à partir du vecteur de transaction $\mathcal{T}$.

DÉFINITION 2 (Pattern). — *Soit $\mathcal{T}$ un ensemble de transactions. Un pattern $\hat{P}$ est une séquence de propriétés présentes dans une ou plusieurs transactions t de $\mathcal{T}$.*

Pour chaque pattern $\hat{P}$, soit $E(\hat{P})$ l'ensemble des items qui le composent et qui correspondent dans notre cas aux propriétés, et $T(\hat{P}) = \{t \in \mathcal{T} \mid E(\hat{P}) \subseteq E(t)\}$ l'ensemble des transactions correspondant. $E(\hat{P})$ désigne l' *expression* de $\hat{P}$, et $|T(\hat{P})|$ son *support*. Un pattern $\hat{P}$ est fréquent si $\frac{1}{|\mathcal{T}|}|T(\hat{P})| \geq \xi$, où ξ désigne un seuil spécifié par l'utilisateur.

EXEMPLE 3. — Considérons le tableau 2, soit $\hat{P} = \{birthDate, residence\}$ et $\xi = 60\ \%$. $\hat{P}$ est fréquent puisque son support qui vaut (66,7 %) est plus élevé que ξ. □

Pour trouver tous les patterns fréquents $\mathcal{FP}$, nous avons utilisé, comme expliqué précédemment, l'algorithme FP-growth pour l'extraction d'itemsets fréquents. Cependant, et à cause de la taille du vecteur de transactions, l'algorithme FP-growth risque de générer un très grand ensemble $\mathcal{FP}$. Mais comme notre but est de mesurer à quel point une transaction (ou la description d'une instance) est *complète* au regard d'un ensemble de propriétés, nous privilégions *l'expression* du pattern (donc les items qu'il contient) plutôt que son *support*.

Concernant le calcul de la complétude, nous devons choisir un seul pattern qui servira de schéma de référence à la source. Ce pattern devra préserver le juste équilibre entre la fréquence et l'expressivité. Dans l'extraction des itemsets, le concept d'itemset fréquent « Maximal » fournit un tel pattern. Par conséquent, pour réduire l'ensemble $\mathcal{FP}$, nous générons un sous-ensemble contenant uniquement les patterns maximaux.

DÉFINITION 4 ($\mathcal{MFP}$). — *Soit $\hat{P}$ un pattern fréquent. $\hat{P}$ est dit maximal si aucun sous-ensemble pouvant être composé à partir de ses propriétés n'est fréquent. Nous définissons l'ensemble des Patterns Fréquents Maximaux $\mathcal{MFP}$ comme suit :*

$$\mathcal{MFP} = \{\hat{P} \in \mathcal{FP} \mid \forall \hat{P}' \supsetneq \hat{P} : \frac{|T(\hat{P}')|}{|\mathcal{T}|} < \xi\}$$

EXEMPLE 5. — En considéront le tableau 2, soit $\xi = 60\%$ et l'ensemble des patterns fréquents $\mathcal{FP}$ = {{*birthDate*},{*residence*},{*citizenship*}, {*birthDate,residence*} {*birthDate,citizenship*}}. L'ensemble $\mathcal{MFP}$ est alors :

$$\mathcal{MFP} = \{\{\textit{birthDate,residence}\}, \{\textit{birthDate,citizenship}\}\}$$

□

4.2. Calcul de la complétude

Lors de cette phase, l'objectif est d'identifier un unique pattern qui servira lors du calcul de la complétude. Lors de la phase précédente, l'ensemble $\mathcal{MFP}$ contient le plus souvent non pas un seul, mais plusieurs patterns candidats. Dans ce cas, notre stratégie consiste à réduire l'ensemble $\mathcal{MFP}$ à un seul pattern. Cependant, les propriétés composant $\mathcal{MFP}$ ne sont pas toutes d'égales importances. Afin de tenir compte de ce fait, nous considérons deux aspects qui sont ; la fréquence d'apparition d'un item (d'une propriété) dans $\mathcal{MFP}$, et le support de chaque $\hat{P} \in \mathcal{FP}$ contenant cet item.

Le résultat est alors un pattern fréquent *pondéré* que nous notons $\hat{P}_w$. Ce pattern sera considéré comme schéma de référence lors du calcul de la complétude.

DÉFINITION 6 (Pattern fréquent pondéré). — *Soit $\mathcal{MFP}$ un ensemble de patterns fréquents maximaux. Un pattern fréquent pondéré $\hat{P}_w$ est un ensemble de couples $< p, w(p) >$ composé à partir d'une propriété p et d'un poids w qui lui est associé. Le poids w est calculé comme suit :*

$$\forall p \in \bigcup_{i=1}^{n} E(\hat{P}_i) : w(p) = \frac{1}{|\mathcal{MFP}|} \sum_{i=1}^{n} \delta(p,i).\frac{|T(\hat{P}_i)|}{|\mathcal{T}|} \quad (1)$$

tel que: p est un $singleton$, $\hat{P}_i \in \mathcal{MFP}$ et

$$\delta(p,i) = \begin{cases} 1 & \textit{if } p \in E(\hat{P}_i) \\ 0 & \textit{sinon} \end{cases}$$

L'équation (1) tient compte, comme nous l'avons mentionné précédemment, de la fréquence d'une propriété p en vérifiant, via le paramètre δ, sa présence dans l'itemset de chaque pattern fréquent maximal. Comme le calcul du poids tient compte du nombre de transactions dans $\mathcal{T}$ (afin d'avoir une valeur de *support relative*), sa valeur sera comprise entre 0 et 1.

EXEMPLE 7. — Soit $\mathcal{MFP}$ = {{*birthDate,residence*}, {*birthDate,citizenship*}} dans lequel chacun des itemsets a une valeur de *support* égale à 67 %. L'ensemble des patterns fréquents pondérés $\hat{P}_w$ est alors:

$$\hat{P}_w = \{\{birthDate, 0, 67\}, \{residence, 0, 33\}, \{citizenship, 0, 33\}\}$$

□

Une fois le pattern pondéré $\hat{P}_w$ ainsi calculé, nous procédons, pour chaque transaction, à une comparaison entre ses propriétés et les propriétés pondérées de $\hat{P}_w$. Nous obtenons ainsi une indication sur la complétude de chaque transaction $t \in \mathcal{T}$.

DÉFINITION 8 (Complétude $\mathcal{CP}$). — *Soit $\mathcal{I}'$ un sous-ensemble d'instances, $\mathcal{T}$ l'ensemble de transactions construites à partir de $\mathcal{I}'$, et $\hat{P}_w$ l'ensemble de propriétés fréquentes pondérées. La complétude de $\mathcal{I}'$ correspond à la complétude de son vecteur de transactions $\mathcal{T}$. Cette complétude est calculée comme la moyenne des complétudes de chacune des transactions au regard de $\hat{P}_w$. Par conséquent, nous définissons la complétude $\mathcal{CP}$ d'un sous-ensemble d'instances $\mathcal{I}'$ comme suit :*

$$\mathcal{CP}(\mathcal{I}') = \frac{1}{|\mathcal{T}|} \sum_{k=1}^{|\mathcal{T}|} \sum_{j=1}^{n} \frac{w(p_j \cap E(t_k))}{\mathcal{W}} \quad (2)$$

tel que :

$$w(\emptyset) = 0,\ \mathcal{W} = \sum_{j=1}^{n} w(p_j),\ p_j \in \bigcup_{j=1}^{n} E(\hat{P}_j),\ \textit{et } \hat{P}_j \in \mathcal{MFP}$$

Il est à noter que les poids des propriétés nous permettent de calculer la complétude d'une transaction selon un contexte qui, dans notre cas, correspond aux propriétés utilisées dans l'ensemble des transactions correspondant aux données considérées. L'algorithme 1 montre le pseudo-code permettant de calculer $\mathcal{CP}(\mathcal{I}')$.

EXEMPLE 9. — Soit $\xi = 60$ %. La complétude du sous-ensemble d'instances du tableau 1 au regard de $\hat{P}_w$ = {{birthDate,0,67}, {residence,0,33},{citizenship,0,33}}, est calculé par:

$$\mathcal{CP}(\mathcal{I}') = (2 * (0,33 + 0,67) + (0,33 + 0,33 + 0,67))/1,33 * 3 = 0,83$$

Cette valeur correspond à la complétude moyenne de l'ensemble de données au regard du schéma inféré $\hat{P}_w$. □

Algorithme 1 : Calcul de la complétude

Données : $\mathcal{D}, \mathcal{I}', \xi$
Résultat : $\mathcal{CP}(\mathcal{I}')$
pour chaque $i \in \mathcal{I}'$ **faire**
  $t_i = \left|p_1 \quad p_2 \quad \dots \quad p_n\right|$;
  $\mathcal{T} = \mathcal{T} + t_i$;
fin
// Extraction du schéma de la source de données
$\mathcal{MFP} = Maximal(\textit{FP-growth}(\mathcal{T}, \xi))$;
$\hat{P}_w =< nil, nil >$;
pour chaque $\hat{P} \in \mathcal{MFP}$ **faire**
  pour chaque $p \in \hat{P}$ **faire**
    // En utilisant l'équation 1
    $w(p_i) = CalculateWeight(p, \mathcal{MFP}, \mathcal{T})$;
    $\hat{P}_w.put(p_i, w(p_i))$;
  fin
fin
// En utilisant l'équation 2
retourner $CP(\mathcal{I}') = CalculateCompleteness(\mathcal{I}', \mathcal{T}, \hat{P}_w)$;

5. Évaluation empirique

Cette section est consacrée à un ensemble d'expérimentations dont le but est d'évaluer notre approche en faisant varier les paramètres sous-jacents. La méthode d'évaluation analyse selon deux points de vue le comportement de la métrique de complétude. Le premier concerne l'impact du nombre d'instances alors que le second est lié au seuil ξ dont la valeur est fixée par l'utilisateur. Les expérimentations utilisent les données issues de deux sources de données réelles qui sont accessibles à un grand nombre d'utilisateurs, qui sont bien connues et qui sont largement utilisées. La première, DBpedia, est une base de connaissances issue d'un effort communautaire dont

les données sont dérivées de Wikipédia. Elle contient actuellement 4,58 millions d'entités. La seconde source, Freebase (avant qu'elle migre vers Wikidata), inclut approximativement 47,3 millions de topics et 2,9 millions de faits.

5.1. Description des sources de données

Pour évaluer la robustesse de l'approche, nous analyserons son comportement vis-à-vis de la nature des données. Pour ce faire, nous avons considéré quelques catégories de différentes natures à partir de deux sources de données qui sont DBpedia et Freebase. Dans DBpedia, nous avons analysé la complétude des instances des catégories $C_1 = \{$*Populated Place*, *Organisation*, *Actor*, *Athlete*$\}$, et dans Freebase les catégories $C_2 = \{$*Citytown*, *Organization*, *Football Player*, *Museum*$\}$ qui se rapprochent le plus. Rappelons que notre objectif n'est pas de calculer une valeur absolue ou une valeur globale de complétude de la source qui, d'un point de vue pratique, sont des objectifs souvent irréalistes. D'ailleurs, calculer la valeur de complétude de toute la source a peu d'intérêt puisque les requêtes des utilisateurs portent le plus souvent sur des sous-ensembles d'instances. De plus, la source de donnée reste intéressante dès l'instant où la valeur de complétude est satisfaisante pour le sous-ensemble sur lequel porte l'intérêt de l'utilisateur même si une valeur globale sur l'ensemble de la source fournit des valeurs moins satisfaisantes. D'autre part, une valeur de complétude élevée supérieure ou égale à 99 % pourrait s'avérer peu intéressante si elle est associée à un schéma de taille insignifiante.

Dans la première étape, nous avons formulé des requêtes sur les sources de données (SPARQL pour DBpedia et MQL pour Freebase) permettant d'extraire les données des catégories sélectionnées. Nous avons ensuite construit l'ensemble $\mathcal{T}$ des transactions leur correspondant. Un vecteur de transactions est constitué de séquences de propriétés dérivées des instances appartenant à une même catégorie (ex. l'ensemble des instantes de *Actors* dans DBpedia). L'ensemble des transactions [7]est ensuite utilisé pour générer les patterns fréquents et pour calculer la complétude. Les expérimentations ont été exécutées sur du matériel Dell XPS 27 avec un processeur Intel Core i7-4770S et 16GB de DDR3 RAM. Le temps d'exécution de chaque expérimentation est négligeable (moins de 5 secondes).

5.2. Impact du nombre d'instances

Dans cette expérimentation nous comparons les valeurs de complétude que nous avons calculées pour chacune des catégories en faisant varier le nombre de transactions (entre 100 et 10 000). Nous avons également calculé les valeurs de la complétude en utilisant une approche « Brute Force » (BFA) que nous comparerons aux résultats de notre approche. Les résultats sont illustrés dans les figures 1 et 3.

7. Les itemsets utilisés dans ces expérimentations sont disponibles à: http://cedric.cnam.fr/~hamdif/upload/cpmining2/

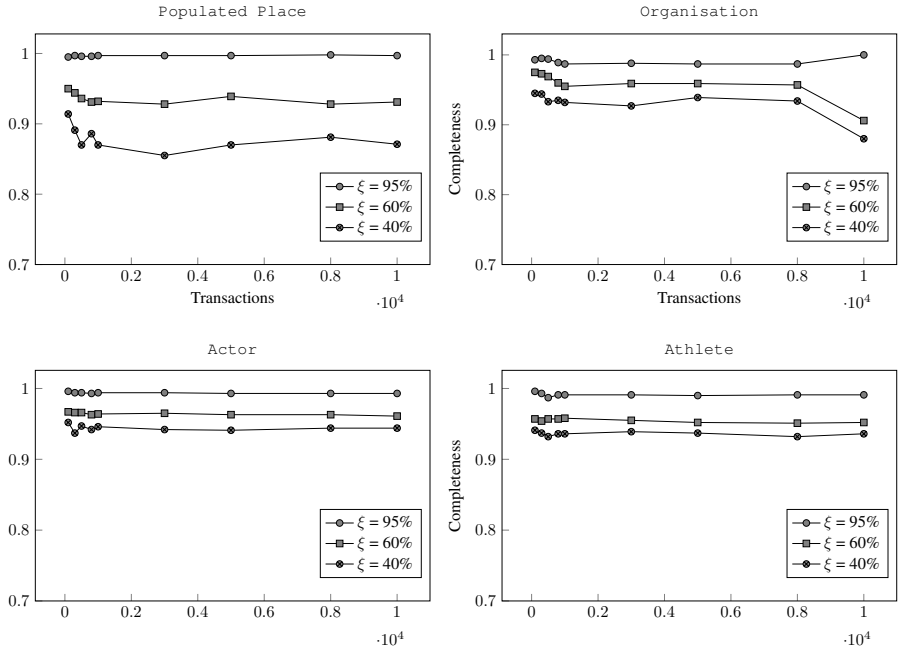

Figure 1. La complétude des catégories DBpedia lorsque le nombre de transactions et la valeur minimale du support ξ varient

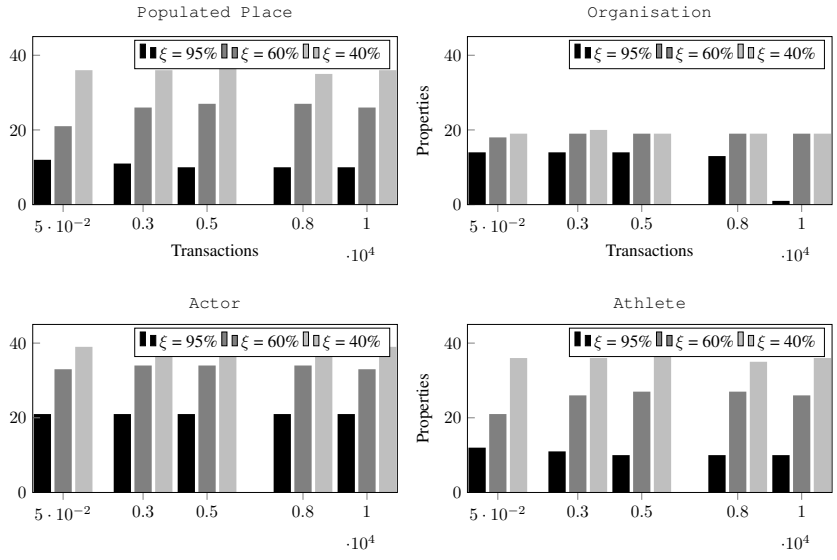

Figure 2. Le nombre de propriétés dans $\hat{P}_w$, pour chaque catégorie DBpedia lorsque le nombre de transactions et la valeur minimale du support ξ varient

Nous constatons, dans ces deux figures, que les valeurs de complétude obtenues par notre approche sont relativement stables vis-à-vis de la variation du nombre de transactions (ex. elles varient entre $0,92$ et $0,95$ pour la catégorie *Populated Place* de DBpedia avec $\xi = 60\%$). Concernant Freebase, les valeurs sont moins régulières que celles de DBpedia. Ceci signifie que la distribution des valeurs manquantes dans DBpedia est plus régulière qu'elle ne l'est dans Freebase. Ce constat et intéressant puisqu'il démontre, au moins pour les catégories observées (ex. *organization*), que les instances ont des descriptions divergentes et peu homogènes bien qu'elles soient du même type (appartenant à la même catégorie).

Les mêmes conclusions s'appliquent au nombre de propriétés comme le montrent les figures 2 et 4. Le nombre de propriétés dans $\hat{P}_w$ reste stable vis-à-vis de la taille des itemsets. Ces résultats attestent de la robustesse de la métrique de complétude vis-à-vis du nombre d'instances.

Revenons maintenant aux valeurs obtenues en appliquant l'approche « Brute Force ». Les valeurs que nous avons obtenues sont très faibles (moins de $0,26$ pour les catégories issues de DBpedia et Freebase) et tendent vers zéro lorsque le nombre de transactions augmente (nous ne les avons par conséquent pas incluses dans les figures). Ce résultat s'explique par le fait que les instances des catégories DBpedia et Freebase n'utilisent pas toutes les propriétés présentes dans les échantillons d'instances que nous avons extraits de ces deux sources. Ceci rejoint ce que nous avions déclaré au sujet de la non-pertinence de l'approche « Brute Force » dès l'instant où les propriétés ne sont pas toutes d'égale importance.

Cette expérimentation montre que, pour chaque catégorie d'un ensemble de données, notre métrique fournit des valeurs pertinentes pour la complétude au regard d'un schéma de référence ($\hat{P}_w$).

5.3. Impact du seuil ξ spécifié par l'utilisateur

Afin de mesurer l'impact de la valeur minimale du support, nous évaluons la corrélation entre ξ et la valeur de complétude et ξ et le nombre de propriétés dans $\hat{P}_w$. Nous utilisons pour cela le coefficient de corrélation des rangs de Spearman ρ.

Les valeurs obtenues pour les échantillons de notre expérimentation sont présentées dans le tableau 3.

Tableau 3. Coefficient de corrélation des rangs de Spearman ρ entre ξ et $\mathcal{CP}$, et ξ et $|\hat{P}_w|$

$\mathcal{CP}$	$\lvert\hat{P}_w\rvert$	$\mathcal{CP}$	$\lvert\hat{P}_w\rvert$	$\mathcal{CP}$	$\lvert\hat{P}_w\rvert$	$\mathcal{CP}$	$\lvert\hat{P}_w\rvert$
$PopulatedPlace$		$Organisation$		$Actor$		$Athlete$	
$0,94$	$-0,82$	$0,9$	$-0,68$	$0,94$	$-0,81$	$0,94$	$-0,82$
$Citytown$		$FootballPlayer$		$Museum$		$Organization$	
$0,81$	$-0,56$	$0,93$	$-0,73$	$0,82$	$-0,36$	$0,78$	$-0,81$

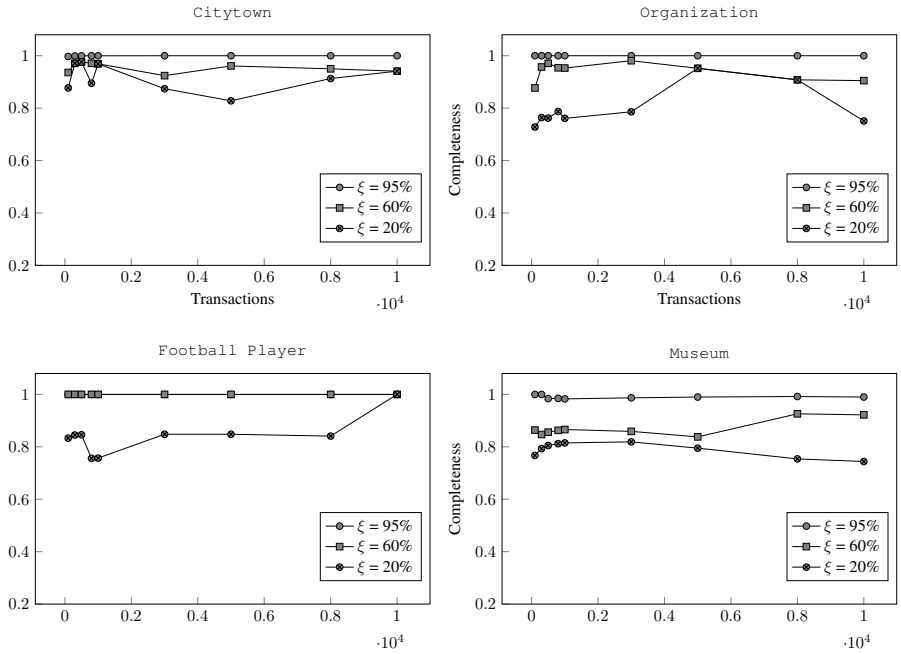

Figure 3. La complétude des catégories Freebase lorsque le nombre de transactions et la valeur minimale du support ξ varient

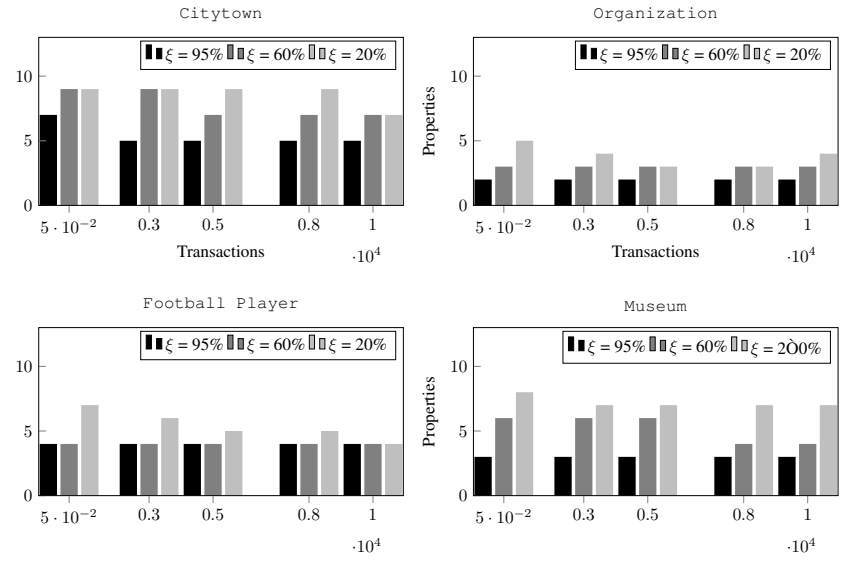

Figure 4. Le nombre de propriétés dans $\hat{P}_w$, pour chaque catégorie Freebase, lorsque le nombre de transactions et la valeur minimale du support ξ varient

Pour savoir si la valeur obtenue pour ρ est significative, où ρ suit une distribution statistique t avec un degré de liberté de $n-2$, telle que $t = \rho\sqrt{\frac{n-2}{1-\rho^2}}$ (où $n = 27$ est la taille de notre échantillon). Nous constatons donc une corrélation positive significative des rangs entre ξ et les valeurs de complétude au niveau $0,0005$ (niveau de confiance de $99{,}9$ %). Ces valeurs sont observées à la fois pour les catégories de DBpedia et de Freebase ce qui est très bon dans la région de rejet de l'hypothèse nulle. Nous pouvons donc conclure qu'il y a une corrélation positive élevée entre le seuil défini par l'utilisateur et la valeur de complétude calculée. En ce qui concerne $|\hat{P}_w|$, une corrélation négative avec ξ paraît évidente. En effet, la corrélation négative pour les catégories issues de DBpedia est statistiquement significative au niveau $0{,}005$. Cependant, lorsque nous analysons les valeurs de ρ pour les catégories issues de Freebase, la corrélation n'est significative qu'au niveau $0{,}05$ mais reste tout de même significative. Ceci est dû au fait que dans Freebase le nombre de propriétés varie peu lorsque l'on fait varier ξ surtout pour la catégorie *Football Player* comme le montre la figure 4.

5.4. Discussion

Les expérimentations, que nous avons menées, montrent que la métrique de complétude fournit des valeurs significatives et pertinentes au regard d'un pattern ou d'un schéma de référence extrait à partir des données.

La première observation issue des résultats de ces expérimentations concerne le comportement de la métrique de complétude. Les résultats obtenus démontrent la robustesse de la mesure vis-à-vis de la taille des échantillons et de la nature des données. La différence entre les valeurs obtenues pour DBpedia, comparées à celle de Freebase (plus stables pour DBpedia), pourrait s'expliquer par le fait que DBpedia dispose d'une ontologie sous-jacente qui a été créée à partir des informations (infoboxes) les plus communément utilisées dans Wikipedia. Dans Freebase cependant, les données sont importées depuis une grande variété de sources produisant probablement des descriptions de données plus hétérogènes. De plus, nous avons constaté de meilleures valeurs lorsque les données concernent des catégories populaires, telles que *Actors*, qui suscitent un effort de publication supérieur à celui consacré à des catégories telles que *organization*.

La deuxième observation concerne le seuil ξ défini par l'utilisateur. Ce paramètre est directement lié à l'expressivité souhaitée du schéma inféré. Les expérimentations mettent en évidence une corrélation élevée avec la complétude au regard du schéma inféré. Ceci signifie que notre approche est capable de fournir à l'utilisateur un schéma de référence répondant à ses exigences. Une telle caractéristique est très intéressante et pourrait avoir diverses applications pratiques. Elle pourrait par exemple être utilisée pour développer une méthode d'assistance à l'expression de requêtes sur des données RDF lorsque le schéma de ces données est absent. Ce schéma, en plus de servir de canevas à l'expression des requêtes, permet d'assurer un degré de complétude des réponses obtenues qui de fait pourrait être contrôlé par l'utilisateur.

Il est également à noter qu'en plus d'assurer une bonne valeur de complétude, le schéma inféré garantit une bonne expressivité puisqu'il contient un nombre intéressant de propriétés (autour de 40 propriétés pour certaines catégories de DBpedia). Dans le cas Freebase, les patterns inférés sont moins expressifs et ne dépassent pas les 12 propriétés pour les échantillons utilisés dans l'expérimentation. Pour conclure, nous pouvons dire que, plus la description des instances est homogène, plus les valeurs de complétude sont stables et plus expressifs seront les schémas inférés.

Enfin, une fois que l'on aura inféré le schéma de référence pour une source de données, et que l'on aura calculé la complétude de la source au regard de ce schéma, les propriétés composant ce schéma constituent de bons candidats pour interconnecter cette source à d'autres sources externes. Ceci est dû au fait que les valeurs de complétude qu'assure le schéma se propagent à travers les propriétés qui le composent en assurant une certaine complétude des liens d'interconnexion pour les sources externes.

6. État de l'art

Durant les deux dernières décennies, un important effort a été fourni par les chercheurs pour proposer des solutions à la gestion de la qualité de l'information. Cet effort, qui a porté aussi bien sur des aspects théoriques que pratiques, a conduit à la proposition de plusieurs méthodes, modèles et solutions permettant de gérer la qualité des systèmes d'information traditionnels et de leurs données (Lee *et al*., 2002 ; Batini *et al*., 2009 ; Berti-Equille *et al*., 2011).

Dans le domaine du web de données, la problématique de la qualité est récente, mais suscite néanmoins un intérêt grandissant. De nombreux chercheurs ont soulevé le fait qu'il ne suffit pas de rendre les données accessibles au plus grand nombre d'utilisateurs surtout lorsque ces utilisateurs sont des entreprises ou des organisations gouvernementales. En effet, dans ce cas, les données sont utilisées à des fins commerciales, de recherche ou pour assurer la sécurité des pays et des citoyens. Par conséquent, la crédibilité des sources de données devient primordiale et exige l'assurance ou au moins la connaissance de la qualité des données qu'elles contiennent.

Les travaux existant sur ce sujet peuvent être classés selon les 4 catégories issues du modèle TDQM - Total Data Quality Management - (Wang, Strong, 1996) que sont : *La qualité intrinsèque* (exactitude, réputation, crédibilité et provenance), *la qualité Représentationnelle* (facilité de compréhension, représentation cohérente, représentation concise et facilité d'interprétation), *l'Accessibilité* (facilité d'accès et accès sécurisé) et *la qualité Contextuelle* (volume de données, pertinence, complétude et fraicheur).

La qualité intrinsèque s'appuie, pour son évaluation, sur les caractéristiques internes des données et est indépendante de leur environnement et de leur usage. Ce qui importe c'est que les données soient exactes (correctes et précises) et cohérentes entre elles. Fürber et Hepp (Fürber, Hepp, 2011) distinguent deux types d'exactitude dites syntaxique et sémantique. L'exactitude syntaxique passe souvent par la défini-

tion de règles sur les valeurs et les formats autorisés pour les données. L'exactitude sémantique, plus difficile à mesurer, peut être vérifiée par la définition de règles de dépendance entre les propriétés. D'autres auteurs reportent le problème de l'exactitude sur la confiance accordée à la source et concentrent les efforts sur la qualification de la provenance (Omitola *et al.*, 2010 ; Markovic *et al.*, 2012 ; Hartig, 2008 ; Golbeck, 2006).

La qualité représentationnelle concerne la qualité externe. Elle considère les facteurs qui ont un impact direct sur la perception de l'utilisateur et sur son comportement vis-à-vis des données. La concision, par exemple, a un impact non seulement sur la perception, mais également sur les performances. On peut citer des travaux qui se sont intéressés à la facilité de compréhension (Mendes, Bizer *et al.*, 2012) ou à la concision (Zaveri *et al.*, 2013). Dans (Fürber, Hepp, 2011), les auteurs proposent de gérer la concision à travers des règles d'unicité. La facilité de compréhension passe plutôt par des recommandations encourageant l'usage de vocabulaires et de formats connus.

L'accessibilité signifie que la donnée est disponible et peut être extraite facilement et rapidement (Pipino *et al.*, 2002). C'est une dimension de la qualité très importante pour le web de données qui a fait l'objet de contributions récentes. Dans (Hogan *et al.*, 2010) les auteurs présentent les erreurs communes, commises lors de la publication de données RDF, qui impactent directement l'accessibilité de ces données. Les auteurs proposent des solutions pour la détection de liens morts ou pour vérifier la disponibilité des données surtout dans le cas de données interconnectées.

Enfin, la qualité contextuelle met l'accent sur le fait que l'évaluation de la qualité ne peut être considérée indépendamment de son contexte de production et d'usage. Cette dimension est d'autant plus importante que le contexte est très changeant dans le web de données. Par exemple, les auteurs dans (Hartig, Zhao, 2009) montrent comment peut-on exploiter la provenance des données pour évaluer leur fraicheur. Pour évaluer la pertinence des données lors de l'interrogation, les auteurs dans (Chen, Garcia, 2010) proposent des métriques qui évaluent l'adéquation du volume de données aux exigences du processus d'exploration des relations entre sources. La pertinence peut également être évaluée en passant par un processus de notation ou de classement des données (Eastman, Jansen, 2003 ; Herzig, Tran, 2012).

La complétude est une autre dimension de la qualité contextuelle qui est reconnue comme difficile à évaluer. Une donnée incomplète occulte une partie de la réalité et peut conduire à des décisions et/ou des interprétations erronées. Cependant, il est rare de rencontrer des sources de données réelles sans valeurs manquantes. L'incomplétude pourrait provenir de défauts de compréhension, d'erreurs d'interprétation ou de l'inadéquation des traitements ou du stockage conduisant à l'altération ou à la perte de données. Dans ce cas, le rôle de l'évaluation de la complétude est de réduire le biais induit par les données manquantes en permettant à l'utilisateur d'intégrer la connaissance de l'incomplétude lors de l'exploitation des données.

Une partie des travaux de recherche dans cette direction adresse le problème de la complétude des réponses aux requêtes. Dans (Darari *et al.*, 2013) les auteurs ont

introduit un framework permettant de spécifier des déclarations liées à la complétude de données RDF. Fürber et Hepp (Fürber, Hepp, 2011) distinguent la complétude du schéma de la complétude de la population ou des données. Dans le premier cas, la complétude mesure le degré d'adéquation du schéma et des propriétés qui le composent avec les données que l'on souhaite décrire. Dans le second cas, le but est de mesurer à quel point les données dont on dispose sont complètes, comparées à une population supposée complète de ces données. Une vision similaire est également proposée dans (Mendes, Mühleisen, Bizer, 2012) à travers les concepts de complétude intentionnelle et extensionnelle . Ces définitions et les métriques qui s'y rattachent sont très proches de celles qui ont été définies dans le contexte des bases de données relationnelles où le schéma ou l'intention correspondent au schéma de la base de données et la population ou l'extension représentent ses instances supposées être l'image du « monde réel ».

À noter que la limite d'une telle vision est le fait que l'on fasse l'hypothèse du monde fermé. Dans ce cas on suppose disposer d'une source de données de référence dont la description et la population sont complètes et pourraient, de ce fait, servir de référence à l'évaluation de la complétude d'une autre source supposée représenter les mêmes données. Dans notre approche cependant, où nous nous intéressons aussi à la complétude du schéma, nous ne faisons pas l'hypothèse d'un schéma de référence. Nous proposons de calculer la complétude intrinsèque à la source de données en exploitant à la fois son intention (les propriétés qui la décrivent) et son extension (ses instances). L'hypothèse sous-jacente est que, comme les sources sont alimentées par diverses personnes et/ou sont issues de diverses sources, le schéma le plus fréquent est celui qui se rapprocherait le plus de ce que l'on pourrait qualifier de « consensuel ».

7. Conclusion

Dans cet article nous avons présenté une approche permettant l'évaluation d'une source de données RDF. Cette approche est un processus en deux phases dont la première infère, depuis les données, un schéma composé des propriétés les plus pertinentes pour cette source. La seconde phase calcule la complétude de cette source vis-à-vis de ce schéma. L'originalité de l'approche réside dans le fait qu'elle ne requiert ni l'existence d'un schéma de description de la source ni celui d'une source de donnée de référence.

Les propriétés composant le schéma inféré sont obtenues par l'application de l'algorithme FP-growth avec une hypothèse sous-jacente stipulant que plus fréquentes sont les propriétés dans les données plus pertinent sera le schéma qu'elles composent. Cette solution propose un compromis entre une approche basée sur un schéma de référence, difficilement applicable en pratique, et l'approche « Brute Force » qui conduit à des résultats peu pertinents, car ils ne reflètent pas la réalité des données.

L'approche proposée a été évaluée sur deux sources de données réelles et largement utilisées qui sont DBpedia et Freebase. Plusieurs expérimentations ont été menées afin d'évaluer la robustesse de la mesure vis-à-vis de divers paramètres tels que la

taille des échantillons, la nature des données ainsi que le seuil spécifié par l'utilisateur. Nous avons également analysé l'impact qu'a la variation de ces facteurs sur la taille du schéma inféré puisque l'expressivité du schéma est également importante.

L'analyse de nos mesures a révélé certains résultats intéressants et a permis de mettre en exergue les caractéristiques des sources utilisées et certains comportements des communautés qui les maintiennent. Les résultats ont montré par exemple que lorsque les sources de données sont décrites de manière hétérogène, les schémas inférés comprennent peu de propriétés comme c'est le cas pour la catégorie *Organization*.

Les travaux présentés ici suggèrent plusieurs directions de recherche futures comme l'amélioration de la complétude des sources en concentrant l'effort d'enrichissement sur les propriétés les plus pertinentes que suggère le schéma inféré. Nous envisageons également d'exploiter le schéma et les valeurs de complétude pour assister l'expression de requêtes sur les sources de données avec en plus la possibilité d'exprimer des exigences de complétude des résultats retournés.

Bibliographie

Ballou D. P., Pazer H. L. (2003). Modeling completeness versus consistency tradeoffs in information decision contexts. *Knowledge and Data Engineering, IEEE Transactions on*, vol. 15, n° 1, p. 240–243.

Batini C., Cappiello C., Francalanci C., Maurino A. (2009). Methodologies for data quality assessment and improvement. *ACM Computing Surveys (CSUR)*, vol. 41, n° 3, p. 16.

Bechhofer S., Buchan I., De Roure D., Missier P., Ainsworth J., Bhagat J. *et al*. (2013). Why linked data is not enough for scientists. *Future Generation Computer Systems*, vol. 29, n° 2, p. 599–611.

Berti-Equille L., Comyn-Wattiau I., Cosquer M., Kedad Z., Nugier S., Peralta V. *et al*. (2011). Assessment and analysis of information quality: a multidimensional model and case studies. *IJIQ*, vol. 2, n° 4, p. 300–323.

Chen P., Garcia W. (2010). Hypothesis generation and data quality assessment through association mining. In F. Sun, Y. Wang, J. Lu, B. Zhang, W. Kinsner, L. A. Zadeh (Eds.), *Proceedings of the 9th IEEE international conference on cognitive informatics, ICCI 2010, july 7-9, 2010, beijing, china*, p. 659–666. IEEE.

Codd E. F. (1986). Missing information (applicable and inapplicable) in relational databases. *SIGMOD Record*, vol. 15, n° 4, p. 53–78.

Darari F., Nutt W., Pirrò G., Razniewski S. (2013). Completeness statements about RDF data sources and their use for query answering. In H. Alani *et al*. (Eds.), *The semantic web - ISWC 2013 - 12th international semantic web conference, sydney, nsw, australia, october 21-25, 2013, proceedings, part I*, vol. 8218, p. 66–83. Springer.

Eastman C. M., Jansen B. J. (2003). Coverage, relevance, and ranking: The impact of query operators on web search engine results. *ACM Transactions on Information Systems (TOIS)*, vol. 21, n° 4, p. 383–411.

Fürber C., Hepp M. (2011). Swiqa-a semantic web information quality assessment framework. In *Ecis*, vol. 15, p. 19.

Golbeck J. (2006). Combining provenance with trust in social networks for semantic web content filtering. In L. Moreau, I. T. Foster (Eds.), *Provenance and annotation of data, international provenance and annotation workshop, IPAW 2006, chicago, il, usa, may 3-5, 2006, revised selected papers*, vol. 4145, p. 101–108. Springer.

Gouda K., Zaki M. J. (2001). Efficiently mining maximal frequent itemsets. In *Proceedings of the 2001 ieee international conference on data mining*, p. 163–170. Washington, DC, USA, IEEE Computer Society.

Grahne G., Zhu J. (2003). Efficiently using prefix-trees in mining frequent itemsets. In B. Goethals, M. J. Zaki (Eds.), *FIMI '03, frequent itemset mining implementations, proceedings of the ICDM 2003 workshop on frequent itemset mining implementations, 19 december 2003, melbourne, florida, USA*, vol. 90. CEUR-WS.org.

Han J., Pei J., Yin Y. (2000). Mining frequent patterns without candidate generation. In W. Chen, J. F. Naughton, P. A. Bernstein (Eds.), *Proceedings of the 2000 ACM SIGMOD international conference on management of data, may 16-18, 2000, dallas, texas, USA.*, p. 1–12. ACM.

Han J., Pei J., Yin Y., Mao R. (2004, janvier). Mining frequent patterns without candidate generation: A frequent-pattern tree approach. *Data Min. Knowl. Discov.*, vol. 8, n° 1, p. 53–87.

Hartig O. (2008). Trustworthiness of data on the web. In *Proceedings of the sti berlin & csw phd workshop.*

Hartig O., Zhao J. (2009). Using web data provenance for quality assessment. In J. Freire, P. Missier, S. S. Sahoo (Eds.), *Proceedings of the first international workshop on the role of semantic web in provenance management (SWPM 2009), collocated with the 8th international semantic web conference (iswc-2009), washington dc, usa, october 25, 2009*, vol. 526. CEUR-WS.org.

Herzig D. M., Tran T. (2012). Heterogeneous web data search using relevance-based on the fly data integration. In A. Mille, F. L. Gandon, J. Misselis, M. Rabinovich, S. Staab (Eds.), *Proceedings of the 21st world wide web conference 2012, WWW 2012, lyon, france, april 16-20, 2012*, p. 141–150. ACM.

Hogan A., Harth A., Passant A., Decker S., Polleres A. (2010). Weaving the pedantic web. In C. Bizer, T. Heath, T. Berners-Lee, M. Hausenblas (Eds.), *Proceedings of the WWW2010 workshop on linked data on the web, LDOW 2010, raleigh, usa, april 27, 2010*, vol. 628. CEUR-WS.org.

Institute M. G., Chui M., Manyika J., Bughin J., Dobbs R., Roxburgh C. *et al*. (2012). *The social economy: Unlocking value and productivity through social technologies*. McKinsey Global Institute.

Jr. R. J. B. (1998). Efficiently mining long patterns from databases. In L. M. Haas, A. Tiwary (Eds.), *SIGMOD 1998, proceedings ACM SIGMOD international conference on management of data, june 2-4, 1998, seattle, washington, USA.*, p. 85–93. ACM Press.

Lee Y. W., Strong D. M., Kahn B. K., Wang R. Y. (2002). Aimq: a methodology for information quality assessment. *Information & management*, vol. 40, n° 2, p. 133–146.

Markovic M., Edwards P., Corsar D., Pan J. Z. (2012). The crowd and the web of linked data: A provenance perspective. In *Wisdom of the crowd, papers from the 2012 AAAI spring symposium, palo alto, california, usa, march 26-28, 2012.*

Mendes P. N., Bizer C., Young Y., Miklos Z., Calbimonte J., Moraru A. (2012). *Conceptual model and best practices for high-quality metadata.* Deliverable 2.1 of PlanetData, FP7 project 257641 (2012).

Mendes P. N., Mühleisen H., Bizer C. (2012). Sieve: linked data quality assessment and fusion. In *Proceedings of the 2012 joint edbt/icdt workshops*, p. 116–123.

Naumann F., Freytag J.-C., Leser U. (2004). Completeness of integrated information sources. *Information Systems*, vol. 29, n° 7, p. 583–615.

Omitola T., Gibbins N., Shadbolt N. (2010, February). Provenance in Linked Data Integration. In S. Auer, S. Decker, M. Hauswirth (Eds.), *Proc. of Linked Data in the Future Internet at the Future Internet Assembly, Ghent 16/17 Dec 2010*, vol. 700.

Pipino L. L., Lee Y. W., Wang R. Y. (2002). Data quality assessment. *Communications of the ACM*, vol. 45, n° 4, p. 211–218.

Samwald M., Jentzsch A., Bouton C., Kallesøe C. S., Willighagen E., Hajagos J. *et al*. (2011). Linked open drug data for pharmaceutical research and development. *Journal of cheminformatics*, vol. 3, n° 1, p. 19.

Wang R. Y., Strong D. M. (1996). Beyond accuracy: What data quality means to data consumers. *Journal of management information systems*, p. 5–33.

Zaveri A., Rula A., Maurino A., Pietrobon R., Lehmann J., Auer S. *et al*. (2013). Quality assessment methodologies for linked open data. *Submitted to Semantic Web Journal*.