Découverte de cardinalité maximale contextuelle dans les bases de connaissances

E. A. Sidi Aly[1,2] M. L. Diakité[1] A. Giacometti[2] B. Markhoff[2] A. Soulet[2]

[1] Département Mathématiques et Informatique - Université de Nouakchott Al Aasriya (Mauritanie)
[2] Laboratoire d'Informatique Fondamentale et Appliquée de Tours - Université de Tours (France)

arbi2fr@yahoo.fr, diakite@una.mr, prenom.nom@univ-tours.fr

3 juillet 2018

Résumé

Les bases de connaissances du web sémantique doivent être enrichies par des informations utiles aux applications de fouille, de recherche d'information, de question-réponse, etc. En effet, leur génération à partir de plateformes collaboratives ou d'intégration de sources diverses produit des manques d'information, d'une part, et des erreurs ou incohérences d'autre part. Heureusement, leur volume important permet d'en induire des contraintes vraisemblables. Tel est l'objet de l'algorithme présenté dans cet article, qui extrait des règles de cardinalité maximale à partir d'une base de connaissances. L'enrichissement de la base par ces nouveaux axiomes permet d'y trouver plus de faits, positifs ou négatifs, ce qui rend plus pertinentes les évaluations de la qualité des règles générées par des algorithmes de fouille classiques. Les expérimentations menées sur une partie de DBpedia et sur l'ensemble d'une base de connaissances numismatiques démontrent la faisabilité de l'approche et la pertinence des contraintes extraites.

Mots Clef

Découverte de cardinalité, base de connaissances.

Abstract

The big semantic web knowledge bases have to be enriched for applications in data mining, information retrieval, question answering, etc. Indeed, their generation from collaborative platforms or integration of various sources leads to lack of information on the one hand, and inconsistencies on the other hand. Fortunately, their volume makes it possible to induce probable constraints. This is the aim of the algorithm presented in this article, which extracts maximum cardinality rules from a knowledge base. Adding these new axioms to the knowledge base allows applications to find more facts, positive or negative, which makes more relevant the evaluations of the quality of the rules generated by traditional datamining algorithms. Experiments conducted on part of DBpedia and on an entire numismatic knowledge base demonstrate the feasibility of the approach and the relevance of the discovered contextual constraints.

Keywords

Cardinality discovery, knowledge base.

1 Introduction

Nous considérons de grandes bases de connaissances du web, construites par des algorithmes de recherche d'information à partir de plateformes collaboratives (e.g., DBpedia [2]) et/ou d'intégration de sources diverses. Pour en désigner les éléments, nous utilisons les termes *concept*, *rôle* et *individu* au sens des logiques de description.

Contexte et motivations En représentation des connaissances les restrictions numériques précisant le nombre d'occurrences d'un rôle sont particulièrement utiles [3]. Parmi elles, les contraintes de cardinalité maximale permettent de savoir quand toutes les assertions sur un rôle donné pour un individu donné existent dans la base. C'est utile pour qualifier les réponses aux requêtes sur une base de connaissances, c'est-à-dire les compléter par des informations précises sur leur qualité en terme de *rappel* par rapport à une réalité [13, 17].
Il est illusoire d'espérer des ajouts manuels de telles contraintes d'intégrité dans de grandes bases de connaissances [1], qui soient correctes et suffisantes. Aussi, des techniques de type rétro-ingénierie [14] applicables sur ces grandes bases doivent être considérées, afin de les rechercher systématiquement. Des propositions existent déjà pour trouver des contraintes de clés [1, 11, 15, 16] dans des données RDF. Mais à notre connaissance, il n'y a pas encore de travaux sur l'extraction de contraintes de cardinalité maximale dans les bases de connaissances.

Challenge L'extraction de contraintes de cardinalité à partir des données existantes est connue comme un problème important de la rétro-ingéniérie des bases de données relationnelles [14, 18]. Par rapport au cadre des bases de données traditionnelles, ce problème est bien plus complexe pour les bases de connaissances du web.
Tout d'abord, ces bases de connaissances contiennent généralement des **données incohérentes**, que ce soient

1. [5] présente néanmoins un outil pour le faire sur Wikidata.

des assertions fausses ou des assertions dupliquées. De ce fait, la cardinalité maximale observée pour un rôle donné ne saurait être considérée comme sa cardinalité maximale la plus vraisemblable. Par exemple, il est vraisemblable qu'une personne ait au plus une année de naissance et deux parents. Pourtant dans DBpedia (voir les rôles `dbo:birthYear` et `dbo:parent` dans la table 1), certaines personnes ont 5 années de naissance ou 6 parents ! Ces quelques assertions incohérentes ne doivent pas influencer la caractérisation des cardinalités maximales.

Ensuite, ces bases de connaissances sont souvent **incomplètes pour un rôle donné**. Pour cette raison, la cardinalité la plus observée n'est pas forcément la cardinalité maximale. Typiquement, la plupart des personnes décrites dans DBpedia n'ont qu'un seul parent renseigné (voir le rôle `dbo:parent` dans la table 1). Toutefois, certaines en ont plus et ceci n'est pas une anomalie, il faut en tenir compte : la cardinalité maximale du rôle `dbo:parent` pour une personne ne doit pas être sous-estimée (ici à 1) au vu de l'ensemble des cardinalités observées.

Enfin, des travaux récents sur la détection de contraintes de clefs dans les bases de connaissances [16] ont montré que de nombreuses contraintes intéressantes ne sont **valides que sur une partie** d'une base de connaissances. Par exemple, s'il semble difficile de déterminer une cardinalité maximale pour le nombre de nationalités d'une personne en général, comme certains états limitent le nombre de nationalités à 1 il est possible de détecter cette limite pour les ressortissants de tels états. Il est donc essentiel non seulement de détecter des cardinalités maximales, mais aussi d'identifier *les contextes* dans lesquels de telles contraintes peuvent être détectées.

Contributions Etant donnée une distribution de cardinalités $(n_i)_{i\geq 1}$ observées dans une base de connaissances $\mathcal{K}$ pour un rôle R dans un contexte C, nous commençons par proposer **une méthode de calcul d'une cardinalité maximale vraisemblable**, en calculant une estimation du taux de cohérence *réel* que la cardinalité i soit maximale. Cette estimation, notée τ_i, est calculée en prenant en compte tous les individus pour lesquels le rôle R est complet. Son calcul est détaillé et justifié dans la section 4.2. Pour être statistiquement valide, une version corrigée de cette estimation du taux de cohérence, notée $\widetilde{\tau}_i$, est également introduite. Des exemples d'estimations de taux cohérence, corrigés ou non, sont représentés dans la table 1 pour les rôles `dbo:birthYear`, `dbo:parent` et `dbo:nationality` en considérant le concept `dbo:Person` comme contexte.

Etant donnée une arborescence de concepts constituant les contextes candidats, nous proposons ensuite **un algorithme d'exploration systématique d'un ensemble de contraintes contextuelles** pour les rôles desquels nous recherchons les cardinalités maximales. Cet algorithme, décrit en section 4.3, vise à limiter les calculs en élaguant un maximum des contraintes possibles.

Enfin nous présentons et analysons des résultats expérimentaux obtenus sur une base de connaissances résultant d'un processus d'intégration de 5 bases de données numismatiques [6].

`dbo:Person` / `dbo:birthYear`			
i	n_i	τ_i	$\widetilde{\tau}_i$
5	1	1.0	0.0
4	2	0.667	0.0
3	4	0.571	0.0
2	91	0.928	0.775
1	**159841**	**0.999**	**0.996**
`dbo:Person` / `dbo:parent`			
i	n_i	τ_i	$\widetilde{\tau}_i$
6	1	1.0	0.0
4	9	0.9	0.420
3	75	0.882	0.718
2	**9392**	**0.991**	**0.975**
1	10643	0.529	0.518
`dbo:Person` / `dbo:nationality`			
i	n_i	τ_i	$\widetilde{\tau}_i$
8	2	1,000	0,000
6	1	0,333	0,000
5	1	0,250	0,000
4	13	0,765	0,397
3	167	0,908	0,796
2	3 263	0,947	0,921
1	123 386	0,973	0,969

TABLE 1 – Distributions de cardinalités de rôles de personnes dans DBpedia (i est la cardinalité ; n_i le nombre d'individus étant i fois sujets du rôle considéré ; τ_i est une estimation fréquentielle du taux de cohérence réel ; $\widetilde{\tau}_i$ en est une version corrigée s'appuyant sur la borne de Hoeffding)

2 Etat de l'art

Notre algorithme vise à augmenter la connaissance sur les données contenues dans les grandes bases de connaissances du web, en termes de validité comme en termes de complétude par rapport à la réalité représentée. Il permet d'enrichir la partie schéma (TBox en logiques de description) de ces bases pour mieux utiliser leur partie données (ABox). Plusieurs travaux récents vont dans ce sens [1, 11, 15, 16, 10] et d'autres s'en rapprochent [7, 13, 17] mais ciblent des individus (assertions de la ABox) plutôt que des concepts (assertions de la TBox).

Dans [17], une technique de fouille de textes de Wikipedia pour ajouter des précisions sur le degré de complétude des informations dans Wikidata est décrite. Notre proposition est complémentaire puisque notre algorithme traite les données déjà contenues dans les bases de connaissances. Mais surtout, il ne caractérise pas les rôles par rapport à des *individus* précis mais à des *concepts* définis (au sens des logiques de description). Les auteurs de [7, 13] présentent également des propositions pour déterminer quand est-ce qu'un rôle particulier (comme `dbo:parent`) manque pour un individu particulier (comme *Obama*). Plus générale, notre proposition consiste à calculer les cardinalités maximales vraisemblables des rôles relativement à des concepts

définissant des contextes : elle enrichit donc la partie schéma.

Ce sont des clés au sens des bases de données, donc des contraintes au niveau du schéma, qui sont recherchées dans [1, 11, 15, 16]. L'idée est de trouver des axiomes indiquant que tout individu d'un certain concept doit posséder une valeur unique pour un rôlé donné R. Cela constitue donc une cardinalité maximale du rôle R pour le concept C. Egalement très proches de nos travaux, dans [10], les auteurs proposent de déterminer automatiquement quels rôles devraient être obligatoirement renseignés pour un concept donné de la base de connaissances. Pour cela ils comparent la densité du rôle pour les individus de ce concept par rapport à sa densité pour les individus d'autres concepts, qui lui sont liés dans la hiérarchie des concepts. Notre proposition s'appuie sur d'autres critères pour calculer la cardinalité maximale du rôle pour un contexte (notion plus générale que seulement les concepts de la base). Elle peut être adaptée au calcul de la cardinalité minimale, auquel cas elle trouverait, entre autres, quels rôles ont une cardinalité minimale au moins supérieure à 1 pour un concept donné, soit plus d'information que seulement savoir si le rôle devrait exister ou pas.

Ces différentes sortes d'information supplémentaire sur la qualité des données de la base de connaissances, en termes de validité comme en termes de complétude par rapport à la réalité représentée, permettent d'améliorer le fonctionnement des applications qui les utilisent, en réduisant le flou de l'hypothèse du monde ouvert. Ainsi pour améliorer la mesure de qualité de règles issues de processus de fouille dans les bases de connaissances du web sémantique, une *hypothèse de complétude partielle* est définie et utilisée dans [8, 7] : cette règle stipule que si un rôle est renseigné pour un individu, alors les informations concernant ce rôle pour cet individu sont considérées complètes. Si on peut noter que cette hypothèse est contredite par l'observation de DBpedia (voir l'extrait fourni dans la table 1), elle rend tout de même plus précis le calcul de la confiance associée aux résultats de fouille. Ces auteurs ont démontré le besoin pour la fouille de ce qu'ils appellent des *oracles de complétude*, et proposé un certain nombre d'heuristiques pour en définir, comme par exemple la popularité des individus (qui augmente les chances que les faits renseignés sur eux soient complets), etc.

La fouille de données est loin d'être le seul domaine qui bénéficie d'axiomes tels que ceux découverts par notre algorithme, par exemple, s'appuyant sur des travaux de référence en base de données, les auteurs de [4, 12] et plus récemment [9] proposent de caractériser les réponses obtenues par des requêtes, en fonction des informations connues concernant le degré de complétude de la base de connaissances interrogée, par rapport à la réalité représentée.

3 Préliminaires

3.1 Bases de connaissances

Dans ce papier, nous considérons des *bases de connaissances* $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ où $\mathcal{T}$ et $\mathcal{A}$ sont respectivement les TBox et ABox de $\mathcal{K}$. $\mathcal{T}$ désigne un ensemble d'axiomes terminologiques définis à partir des concepts et rôles atomiques de $\mathcal{K}$, alors que $\mathcal{A}$ désigne l'ensemble des assertions ou faits de $\mathcal{K}$. Plus précisément, $\mathcal{A}$ contient des expressions de la forme $C(a)$ et $R(a, b)$ où C est un *concept*, R est un *rôle*, et a, b sont des *individus*.
Dans le cas de la base de connaissances DBpedia, `dbo:Country` et `dbo:Person` sont des exemples de concepts atomiques et `dbo:nationality` est un exemple de rôle atomique de sa TBox. Par ailleurs, `dbo:Country`$(Mauritania)$ et `dbo:nationality`$(Arby, Mauritania)$ sont des exemples de faits ou assertions de sa ABox. Le premier indique que $Mauritania$ est un pays, alors que le second indique que $Arby$ est de nationalité mauritanienne.

Les logiques de description permettent de définir des axiomes pour enrichir la TBox d'une base de connaissances. Par exemple, la relation d'inclusion $\sqsubseteq$ permet d'indiquer qu'un concept C_1 est inclus dans un concept C_2, noté $C_1 \sqsubseteq C_2$. Plus précisément, une base de connaissances $\mathcal{K}$ implique l'axiome $C_1 \sqsubseteq C_2$ si pour toute interprétation $\mathcal{I}$ de $\mathcal{K}$, $C_1^{\mathcal{I}} \subseteq C_2^{\mathcal{I}}$. Par exemple, les axiomes $\exists$`dbo:nationality`$.\top \sqsubseteq$ `dbo:Person` et $\exists$`dbo:nationality`$^{-}.\top \sqsubseteq$ `dbo:Country` indiquent respectivement que le domaine du rôle `dbo:nationality` est inclus dans le concept `dbo:Person`, et que le co-domaine du rôle `dbo:nationality` est inclus dans le concept `dbo:Country`.

3.2 Contraintes contextuelles de cardinalité maximale

Soit R un rôle d'une base de connaissances $\mathcal{K} = (\mathcal{T}, \mathcal{A})$. On considère généralement que ce rôle satisfait dans $\mathcal{K}$ une contrainte de cardinalité maximale M si pour tout sujet s, le nombre d'objets o tels que $R(s, o)$ soit présent dans $\mathcal{K}$ (directement présent dans sa ABox $\mathcal{A}$ ou inférable à partir de ses TBox $\mathcal{T}$ et ABox $\mathcal{A}$) est inférieur ou égal à M.
En logique de description, une telle contrainte peut se représenter par un axiome de la forme $sqsubseteq$ en utilisant le constructeur de restriction de cardinalité $(\leq MR)$. En effet, en terme logique, une base de connaissances $\mathcal{K}$ implique l'axiome $\exists R.\top \sqsubseteq (\leq MR)$ si pour toute interprétation $\mathcal{I}$ de $\mathcal{K}$, $\{x : (\exists y)((x, y) \in R^{\mathcal{I}})\} \subseteq \{x : \#\{y : (x, y) \in R^{\mathcal{I}}\} \leq M\}$ où $\#E$ représente la cardinalité d'un ensemble E.
Plus précisément, dans ce papier, nous cherchons à identifier des contraintes *contextuelles* de cardinalité maximale, à savoir des contraintes qui ne sont pas nécessairement vérifiées par tous les sujets s d'un rôle R, mais par tous les sujets instances d'un concept, qu'il soit atomique ou composé, déjà défini dans $\mathcal{K}$ ou pas. Cette notion est introduite

formellement dans la définition suivante :

Définition 3.1 (Contrainte contextuelle)**.** *Etant donnés un rôle R, un concept atomique ou défini C et un entier M, une contrainte contextuelle de cardinalité maximale définie sur R est une expression γ de la forme : $C \sqsubseteq (\leq M\, R)$.*
Le concept C est appelé le contexte de la contrainte γ. La contrainte γ est satisfaite dans une base de connaissances $\mathcal{K}$ si et seulement si pour toute interprétation $\mathcal{I}$ de $\mathcal{K}$, $C^{\mathcal{I}} \subseteq \{x : \#\{y : (x, y) \in R^{\mathcal{I}}\} \leq M\}$.

Par exemple, la contrainte contextuelle $(\texttt{dbo:Person}) \sqsubseteq (\leq 5\, \texttt{dbo:nationality})$ indique que toutes les personnes ont au plus 5 nationalités, alors que la contrainte contextuelle $(\texttt{dbo:Person} \sqcap \exists \texttt{dbo:nationality}.\{China\}) \sqsubseteq (\leq 1\, \texttt{dbo:nationality})$ indique que toutes les personnes de nationalité chinoise ont au plus une nationalité.

Dans ce travail, on cherche à extraire des contraintes contextuelles de cardinalité maximale qui soient les plus générales possibles.

Définition 3.2 (Contrainte contextuelle minimale)**.** *Soient deux contraintes contextuelles de cardinalité maximale $\gamma_1 : C_1 \sqsubseteq (\leq M_1\, R)$ et $\gamma_2 : C_2 \sqsubseteq (\leq M_2\, R)$ définies sur R.*
La contrainte γ_1 est dite plus générale que la contrainte γ_2 si $C_2 \sqsubset C_1$ et $M_1 \leq M_2$. Etant donné un ensemble de contraintes Γ définies sur R, une contrainte $\gamma_1 \in \Gamma$ est dite minimale dans Γ s'il n'existe aucune contrainte γ_2 dans Γ plus générale que γ_1.

Par exemple, la contrainte contextuelle $(\texttt{dbo:Person}) \sqsubseteq (\leq 2\, \texttt{dbo:nationality})$ est plus générale que la contrainte contextuelle $(\texttt{dbo:Person} \sqcap \exists \texttt{dbo:nationality}.\{China\}) \sqsubseteq (\leq 5\, \texttt{dbo:nationality})$ car $(\texttt{dbo:Person} \sqcap \exists \texttt{dbo:nationality}.\{China\}) \sqsubseteq \texttt{dbo:Person})$ et $2 \leq 5$.

La notion de minimalité a pour objectif de ne pas extraire de contraintes contextuelles qui soient redondantes. Intuitivement, considérons les deux contraintes γ_1 et γ_2 introduites dans la définition précédente, et supposons que γ_1 soit plus générale que γ_2. Etant donnée une base de connaissances $\mathcal{K}$ dans laquelle les contraintes γ_1 et γ_2 sont satisfaites, soit une instance s de C_2 dans $\mathcal{K}$. D'après γ_2, nous savons que pour toute interprétation $\mathcal{I}$ de $\mathcal{K}$, $\#\{o : (s, o) \in R^{\mathcal{I}}\} \leq M_2$. Mais comme γ_1 est plus générale que γ_2, nous savons par définition que $C_2 \sqsubseteq C_1$. Il en découle que s est aussi une instance de C_1 dans $\mathcal{K}$, et d'après γ_1, que pour tout interprétation $\mathcal{I}$ de $\mathcal{K}$, $\#\{o : (s, o) \in R^{\mathcal{I}}\} \leq M_1$, ce qui est une contrainte plus forte que $\#\{o : (s, o) \in R^{\mathcal{I}}\} \leq M_2$. En effet, par définition de la minimalité, nous savons que $M_1 \leq M_2$. Par rapport à la contrainte γ_1, la contrainte γ_2 est donc inutile car redondante, i.e. elle ne permet pas de déduire d'information supplémentaire.

Le problème traité dans ce papier est alors le suivant : **étant donnés une base de connaissances $\mathcal{K}$, un rôle R et une hiérarchie de concepts $(\mathcal{C}, \sqsubseteq)$, nous cherchons à découvrir l'ensemble des contraintes contextuelles de cardinalité maximale de la forme $C \sqsubseteq (\leq M\, R)$ avec $C \in \mathcal{C}$, qui soient satisfaites sur $\mathcal{K}$ et minimales dans $\mathcal{C}$.**

En pratique, une base de connaissances telle que DBpedia est très incomplète (par exemple, de nombreuses personnes ont seulement un parent), et elle comporte de nombreuses incohérences (par exemple, des personnes peuvent avoir jusqu'à 5 parents). Pour ces raisons, étant donnée une base de connaissances $\mathcal{K}$, il n'est pas pertinent de chercher à extraire des contraintes de cardinalité qui soient *parfaitement* satisfaites dans $\mathcal{K}$, mais les contraintes :

- *les plus probables et suffisamment probables* par rapport à un seuil donné, de manière à prendre en compte et tolérer les incohérences, et
- *suffisamment certaines* par rapport à un degré de confiance, pour ne pas extraire des contraintes qui soient remises en cause régulièrement par l'ajout de nouveaux faits dans la base de connaissances.

Nous détaillons dans la section suivante comment évaluer la probabilité qu'une contrainte soit satisfaite dans une base de connaissances $\mathcal{K}$ et comment mesurer la certitude que cette contrainte soit réelle.

4 Extraction de contraintes contextuelles de cardinalité maximale

Pour résoudre le problème énoncé précédemment, nous commençons par le reformuler en introduisant la notion de taux de cohérence dans la section 4.1, puis nous décrivons dans la section 4.2 comment détecter une cardinalité maximale pour un rôle R dans un contexte C. Ensuite, étant donné un ensemble de contextes candidats $\mathcal{C}$, nous montrons dans la section 4.3 comment explorer efficacement l'ensemble des contraintes contextuelles possibles.

4.1 Taux de cohérence

Etant donnée une base de connaissances $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, supposons que i soit la cardinalité maximale du rôle R dans le contexte C. Soit s un individu de C dans $\mathcal{K}$, complet pour le rôle R dans $\mathcal{K}$ (dans le sens où tous les faits $R(s, o)$ possibles représentant le monde réel sont dans $\mathcal{A}$ ou inférables). Dans le cas où il existe exactement i faits dans $\mathcal{K}$ de la forme $R(s, o)$, cela renforce l'hypothèse que i soit la cardinalité maximale de R dans le contexte C. Inversement, s'il existe plus de i faits dans $\mathcal{K}$ de la forme $R(s, o)$, cela affaiblit l'hypothèse que i soit la cardinalité maximale de R dans le contexte C. Ainsi dans le tableau 1, pour la classe `dbo:Person`, les individus comportant au moins 3 assertions pour le rôle `dbo:parent` affaiblissent l'hypothèse que la cardinalité maximale soit 2 mais ils restent peu nombreux au regard des 9 392 individus qui ont exactement 2 parents.
En suivant ce raisonnement, nous introduisons la notion de

taux de cohérence pour évaluer si une cardinalité i pour le rôle R dans le contexte C a des chances d'être maximale :

Définition 4.1 (Taux de cohérence)**.** *Etant donnée une base de connaissances $\mathcal{K}$, le taux de cohérence de la cardinalité i pour le rôle R dans le contexte C est le ratio :*

$$\tau_i^{C,R}(\mathcal{K}) = \frac{n_i^{C,R}}{n_{\geq i}^{C,R}}$$

où $n_i^{C,R}$ (resp. $n_{\geq i}^{C,R}$) représente le nombre de sujets s tels que i faits $R(s,o)$ (resp. i faits ou plus) appartiennent à $\mathcal{K}$ dans le contexte C.

Par exemple, dans le tableau 1, $n_{\geq 2}^{\texttt{dbo:Person},\texttt{dbo:parent}}$ est égal à 9477 ($9477 = 9392+75+9+1$). De cette manière, le taux de cohérence $\tau_2^{\texttt{dbo:Person},\texttt{dbo:parent}}(\mathcal{K})$ est de 0,991 (i.e., $9392/9477$). Par la suite, quand le contexte et la relation sont clairs, nous pouvons les omettre dans les notations. Dans ce cas, n_i et τ_i désignent respectivement les termes $n_i^{C,R}$ et $\tau_i^{C,R}$.

Maintenant nous allons formaliser le lien entre le taux de cohérence et la notion de contrainte maximale. Originellement introduit dans [13], $\mathcal{K}^* = (\mathcal{T}^*, \mathcal{A}^*)$ désigne une hypothétique base de connaissances idéale qui contiendrait tous les axiomes et toutes les assertions du monde réel. Comme $\mathcal{K}^*$est correcte et complète, **le taux de cohérence au sein de $\mathcal{K}^*$, noté $\tau_M^{C,R}(\mathcal{K}^*)$, est égal à 1 si et seulement si $C \sqsubseteq (\leq M\ R)$ appartient à $\mathcal{T}^*$.**

En pratique, le taux de cohérence mesuré dans une base de connaissances est différent du taux de cohérence réel : $\tau_i(\mathcal{K}) \neq \tau_i(\mathcal{K}^*)$. Par exemple, le taux de cohérence $\tau_2(\mathcal{K})$ pour le rôle `dbo:parent` du tableau 1 est égal à 0,991 alors que le taux de cohérence réel est égal à 1. Plus grave, on a $\tau_6^{\texttt{dbo:Person},\texttt{dbo:parent}}(\mathcal{K}) = 1$! Le taux de cohérence sur $\mathcal{K}$ est donc une estimation peu fiable du taux de cohérence réel sur $\mathcal{K}^*$.

4.2 Détection d'une contrainte

L'estimation $\tau_i(\mathcal{K})$ de $\tau_i(\mathcal{K}^*)$ doit être corrigée pour être statistiquement valide. Pour ce faire, nous proposons d'utiliser l'inégalité de Hoeffding qui a l'avantage d'être vraie pour toute distribution. En terme de probabilité, si X est une variable aléatoire indiquant pour un sujet s tiré aléatoirement, le nombre de faits $R(s,o)$ appartenant à $\mathcal{K}$, alors τ_i est une estimation fréquentielle de la probabilité conditionnelle $P(X = i \ / \ X \geq i)$. Etant donné un niveau de confiance $1-\delta$, l'inégalité de Hoeffding stipule que $\tau_i(\mathcal{K}^*)$ est compris entre $\tau_i(\mathcal{K}) - \epsilon_i$ et $\tau_i(\mathcal{K}) + \epsilon_i$ où $\epsilon_i = \sqrt{\frac{\log(1/\delta)}{2n_{\geq i}}}$. Dans ce contexte, afin de prendre des décisions les plus sûres, nous proposons d'utiliser la borne inférieure de l'intervalle de confiance $[\tau_i - \epsilon_i, \tau_i + \epsilon_i]$. Plus formellement, on a la propriété suivante :

Propriété 4.1 (Minoration)**.** *Etant données une base de connaissances $\mathcal{K}$ et une confiance $1-\delta$, le taux de cohérence réel $\tau_i(\mathcal{K}^*)$ de la cardinalité i pour le rôle R dans le contexte C est supérieur à $\widetilde{\tau}_i(\mathcal{K})$:*

$$\tau_i(\mathcal{K}^*) \geq \widetilde{\tau}_i(\mathcal{K})$$

où $\widetilde{\tau}_i(\mathcal{K})$ est le taux de cohérence pessimiste défini par :

$$\widetilde{\tau}_i(\mathcal{K}) = \max\left\{\frac{n_i}{n_{\geq i}} - \sqrt{\frac{\log(1/\delta)}{2n_{\geq i}}}; 0\right\}$$

Cette propriété nous munit d'un outil efficace pour approximer le taux de cohérence réel. Il survient alors la difficulté de choisir la cardinalité maximale une fois que l'on dispose pour chaque cardinalité i du taux de cohérence pessimiste $\widetilde{\tau}_i(\mathcal{K})$ (pour un rôle R dans le contexte C).

Plus précisément, **étant donnés un seuil minimal de cohérence min_τ et un niveau de confiance $1-\delta$, nous considérons que M est la cardinalité maximale de R dans le contexte C si et seulement si $\widetilde{\tau}_M \geq min_\tau$ et $M = \arg\max_{i \geq 1} \widetilde{\tau}_i(\mathcal{K})$.**

Quelques exemples d'estimations $\widetilde{\tau}_i$ et de détection de cardinalités maximales contextuelles sont donnés dans la table 1. Dans les 3 exemples, on a considéré `dbo:Person` comme contexte, et on a cherché à détecter la cardinalité maximale contextuelle de trois rôles : `dbo:birthYear`, `dbo:parent` et `dbo:nationality`. Intuitivement, pour les deux premiers rôles, on souhaiterait détecter des cardinalités maximales respectives de 1 et 2. Pour un niveau de confiance $1-\delta = 99\%$ et un seuil $min_\tau = 0.97$, on constate que les cardinalités maximales supposées sont effectivement détectées (cf. lignes en gras dans la table 1). De manière intéressante, avec ces mêmes seuils, aucune cardinalité n'est détectée pour `dbo:nationality`.

4.3 Exploration de l'espace de recherche

Etant donnés une base de connaissances $\mathcal{K}$, un rôle R, une arborescence de concepts $(\mathcal{C}, \sqsubseteq)$, un degré de confiance δ et un seuil minimal de cohérence min_τ, nous cherchons à découvrir l'ensemble des contraintes contextuelles de cardinalité maximale de la forme $C \sqsubseteq (\leq M\ R)$ avec $C \in \mathcal{C}$, qui soient minimales et suffisamment certaines sur $\mathcal{K}$. En pratique, notons que l'arborescence $(\mathcal{C}, \sqsubseteq)$ peut être une arborescence déjà existante dans la TBox de la base de connaissances, ou une arborescence construite dans une phase préalable de préparation des données (voir la section 5.1).

Dans un tel cadre, il y a potentiellement un très grand nombre de contraintes contextuelles à considérer, évaluer et comparer. Néanmoins, il est possible de réduire la taille de l'espace de recherche à explorer en se basant sur les propriétés 4.2 et 4.3 énoncées ci-après. Tout d'abord, la propriété 4.2 montre qu'une contrainte $C \sqsubseteq (\leq M\ R)$ ne peut pas être suffisamment certaine si le contexte C contient trop peu d'individus dans $\mathcal{K}$, car alors l'intervalle de confiance du taux de cohérence calculé grâce à l'inégalité de Hoeffding est très large et sa borne inférieure ne peut être supérieure au seuil min_τ imposé.

Propriété 4.2 (Nombre minimal d'observations)**.** *Etant donnés une base de connaissances $\mathcal{K}$, une contrainte contextuelle de cardinalité maximale $C \sqsubseteq (\leq M\ R)$ et un seuil min_τ, le taux de cohérence $\widetilde{\tau}_M(\mathcal{K})$ que M soit la cardinalité maximale de R dans C ne peut être supérieur à min_τ que si $|C| \geq \frac{log(1/\delta)}{2(1-min_\tau)^2}$.*

Par ailleurs, supposons qu'une contrainte γ définie par $C \sqsubseteq (\leq M\ R)$ avec $M = 1$ ait été détectée comme suffisamment certaine au cours de l'exploration. Alors, d'après la propriété 4.3, il n'est pas nécessaire d'explorer les contraintes γ' définies par $C' \sqsubseteq (\leq M'\ R)$ où C' est plus spécifique que C. Cette propriété découle directement de la définition 3.2 de la minimalité.

Propriété 4.3 (Contrainte minimale)**.** *Soient une base de connaissances $\mathcal{K}$ et une contrainte contextuelle de cardinalité maximale γ définie par $C \sqsubseteq (\leq M\ R)$ avec $M = 1$. Toute contrainte γ' définie par $C' \sqsubseteq (\leq M'\ R)$ avec $C' \sqsubseteq C$ et $M' \geq 1$ ne peut être minimale.*

L'algorithme 1 détaille notre fonction récursive d'exploration, la fonction $C3M$ (pour *Contextual Cardinality Constraint Mining*). Cette fonction prend en entrée une base de connaissances, un rôle à explorer, un contexte courant, une cardinalité maximale courante ($M = \infty$ si aucune cardinalité maximale n'a encore pu être détectée), et enfin, des seuils δ et min_τ. **Le démarrage de l'exploration d'une arborescence de racine $\top$ se fait en exécutant la fonction** $C3M(\mathcal{K}, R, \top, \infty, \delta, min_\tau)$.
Pour commencer, la fonction $C3M$ détermine si le nombre d'individus est suffisant dans le contexte C. Si ce n'est pas le cas, elle arrête l'exploration à la ligne 2 conformément à la propriété 4.2. Sinon, le taux de cohérence $\widetilde{\tau}_i$ est calculé pour chaque cardinalité i (lignes 4 à 6) et la ligne 7 retient la cardinalité maximale la plus probable. Si le taux de cohérence correspondant n'est pas supérieur au seuil min_τ, alors cela signifie qu'aucune cardinalité maximale n'a pu être détectée à ce niveau et i_M est fixé ligne 8 à ∞. Ensuite, si la cardinalité maximale détectée i_M est strictement inférieure à M (la cardinalité maximale détectée au niveau précédent), alors on dispose d'une nouvelle contrainte minimale de cardinalité maximale i_M et on l'ajoute à Γ, l'ensemble des contraintes recherchées. Finalement, conformément à la propriété 4.3, si i_M est égale à 1, il n'est pas nécessaire de poursuivre l'exploration en parcourant les contextes plus spécifiques de C. Sinon, la fonction $C3M$ est appelée récursivement à la ligne 12 pour tous les C' qui sont des sous-concepts directs de C.
Dans notre implémentation de la fonction $C3M$, nous avons appliqué une approche client-serveur où les distributions de cardinalité $n_i^{C,R}$ sont calculées par interrogation en SPARQL d'une base de connaissances localisée sur un serveur. Dans un tel cadre, la complexité de notre méthode en nombre de requêtes sur le serveur est en $\mathcal{O}(|\mathcal{C}|)$ où $|\mathcal{C}|$ représente le nombre de concepts dans l'arborescence $\mathcal{C}$ explorée. Dans le pire des cas, côté client, la complexité en nombre d'opérations est en $\mathcal{O}(|\mathcal{C}| \times i_{max})$ où i_{max} représente l'entier maximal pour lequel il existe au moins un sujet s tel que i_{max} faits $R(s, o)$ appartiennent à la base de connaissances $\mathcal{K}$, i.e. $i_{max} = \arg\max_{i \in \mathbb{N}}\{n_i^{\top,R} > 0\}$.

Algorithm 1 C3M

Input: Une base de connaissances $\mathcal{K}$, un rôle R, un contexte C, un entier M, un niveau de confiance δ et un seuil minimal de support min_τ
Output: Un ensemble Γ de contraintes contextuelles de cardinalité maximale

1: $\alpha := \frac{log(1/\delta)}{2(1-min_\tau)^2}$ et $n_{\geq 0}^{C,R} := |C|$
2: **if** $(n_{\geq 0}^{C,R} < \alpha)$ **then return** $\emptyset$
3: $\Gamma := \emptyset$ et $i_{max} := \arg\max_{i \in \mathbb{N}}\{n_i^{C,R} > 0\}$
4: **for all** $i \in [1..min\{M, i_{max}\}]$ **do**
5: $\quad \widetilde{\tau}_i := \max\left\{\frac{n_i^{C,R}}{n_{\geq i}^{C,R}} - \sqrt{\frac{\log(1/\delta)}{2n_{\geq i}^{C,R}}}; 0\right\}$
6: **end for**
7: $i_M := \arg\max_{i \in [1..min\{M, i_{max}\}]}\{\widetilde{\tau}_i\}$
8: **if** $(\widetilde{\tau}_{i_M} < min_\tau)$ **then** $i_M = \infty$
9: **if** $(i_M < M)$ **then** $\Gamma := \{C \sqsubseteq (\leq i_M\ R)\}$
10: **if** $(i_M > 1)$ **then**
11: $\quad$ **for all** $C' \in subClassOf(C)$ **do**
12: $\quad\quad \Gamma := \Gamma \cup C3M(\mathcal{K}, R, C', i_M, \delta, min_\tau)$
13: $\quad$ **end for**
14: **end if**
15: **return** Γ

5 Expérimentations

Outre les requêtes sur DBpedia (dont nous montrons des échantillons de réponses en table 1), qui ont été utilisées pour mettre au point la définition du taux de cohérence, nous avons expérimenté l'algorithme 1 sur un jeu de données mis à notre disposition par les auteurs de [6].

5.1 Données et protocole

Le jeu de données utilisé porte sur le domaine numismatique, il est le résultat d'un processus d'intégration mené dans le cadre du projet européen ARIADNE [2]. Ses auteurs ont utilisé le CIDOC-CRM [3] pour intégrer les contenus de 5 ressources construites par des institutions de différents pays européens. Il contient 3 123 998 triplets, dont les définitions de 114 classes et 373 rôles ou propriétés du CIDOC-CRM et d'ARIADNE. Il est stocké et interrogé avec le triplestore Blazegraph (v2.1.4), sur une machine virtuelle sous Linux avec 32 GB de mémoire virtuelle, sur un serveur ayant pour processeur un Dual Intel Xeon E5620 4 coeurs. L'algorithme 1 est implémenté en Java et utilise la bibliothèque de programmation pour RDF Jena [4].
La base porte sur des pièces de monnaies mais, par choix des intégrateurs, il n'existe pas de classe $Coin$. Les individus correspondant à des pièces sont des instances de `E22_Man_Made_Object` caractérisées par certains URIs (ex. `<http://nomisma.org/id/coin>`) comme va-

2. http://ariadne-infrastructure.eu/
3. http://www.cidoc-crm.org/
4. http://jena.apache.org

leur objet de certains rôles (ex. P2_has_type). Plusieurs rôles et plusieurs URIs sont utilisés pour cela, aussi nous avons décidé de construire notre propre arborescence d'exploration de la façon suivante :

Au **premier niveau**, notre arborescence contient tous les concepts C_i de la base, soit 114 concepts ($i \in [1..114]$). Tous ces concepts sont des sous-concepts du **concept racine** $\top$ **au niveau zéro**, i.e. pour tout i, nous avons $C_i \sqsubseteq \top$. Au **deuxième niveau** notre arborescence contient tous les concepts C_i^j définis par $C_i^j := C_i \sqcap (\exists R_j.\top)$ où C_i ($i \in [1..114]$) et R_j ($j \in [1..373]$) sont respectivement des concepts et rôles de la base. A ce niveau, 42 522 concepts C_i^j sont ainsi définis. Enfin, au **troisième niveau**, notre arborescence contient tous les concepts $C_i^{j,k}$ définis par $C_i^{j,k} := C_i \sqcap (\exists R_j.\{a_k\})$ où C_i ($i \in [1..114]$) et R_j ($j \in [1..373]$) sont respectivement des classes et rôles de la base, et a_k est un individu du co-domaine de R_j, i.e. $a_k \in (\exists R_j^{-1}.\top)$. Grâce à ce dernier niveau, il est possible de considérer des contextes à la manière de notre exemple jouet où dbo:Person $\sqcap \exists$dbo:nationality.$\{China\}$. Notons finalement que pour tout i, j, k, nous avons $C_i^{j,k} \sqsubseteq C_i^j \sqsubseteq C_i$. Globalement, cette arborescence comporte 3 160 357 concepts, donc pour les 373 rôles de la base de connaissances cela représente plus d'un milliard de contraintes contextuelles possibles (exactement 1 178 813 161 contraintes). Néanmoins, comme nous le verrons dans la section suivante, l'utilisation des propriétés 4.2 et 4.3 permet d'élaguer une grande partie de l'espace de recherche.

5.2 Résultats

Tous les résultats présentés dans cette section ont été obtenus avec un **seuil minimal de confiance** $1 - \delta = 0,99\%$ (pour des contraintes les plus certaines possibles) et un **seuil minimal de cohérence** $min_\tau = 0,95$ (pour des contraintes suffisamment probables). Ce seuil a été défini expérimentalement. Sur des bases de connaissances de plus grande taille comme DBpedia, un seuil plus élevé est préférable. Néanmoins, l'approche est relativement peu sensible aux seuils (i.e., l'ensemble des contraintes trouvées est stable).

Analyse quantitative. Avec ces paramètres, la propriété 4.2 nous indique qu'une contrainte $C \sqsubseteq (\leq M\ R)$ ne peut être suffisamment certaine si son contexte C contient moins de $\alpha = \frac{log(1/\delta)}{2(1-min_\tau)^2} = 922$ instances. Ainsi, l'utilisation de la propriété 4.2 permet de n'explorer que 16 641 contraintes, soit moins de $0,002\%$ des plus de 1 milliard de contraintes possibles. Qui plus est, notre expérience montre que la propriété 4.3 permet de réduire encore de $82,5\%$ la taille de l'espace de recherche à explorer. Au final, avec les seuils choisis notre algorithme ne cherche à détecter une cardinalité maximale que pour 2 909 contextes possibles, avec un temps de calcul complet de moins de 50 minutes.

La table 2 donne une vue globale et quantitative du résultat de l'exploration réalisée. Sur les 2 909 contraintes contextuelles possibles, notre algorithme a détecté au total 887 contraintes de cardinalité maximale, 595 d'entre elles étant des contraintes minimales. Sur cet exemple, le critère de minimalité permet donc de réduire de près de 67% le nombre de contraintes retournées. On constate que les contraintes les plus nombreuses sont des cardinalités maximales avec $M = 1$, ce qui correspond à des contraintes où pour un rôle donné R, tout sujet s est en relation avec au plus un objet o. Néanmoins de très nombreuses contraintes sont trouvées avec des cardinalités maximales $M \in \{2, 3\}$ (37% des contraintes minimales détectées). On note également que si des contraintes de cardinalités maximales sont détectées dès le niveau 0 (65 contraintes avec un contexte $C \equiv \top$), la recherche de contraintes contextuelles est particulièrement pertinente. Il faut en effet noter que les contraintes les plus nombreuses sont trouvées au niveau 3 (75% des contraintes détectées), sachant que par construction de notre arborescence, c'est à ce niveau que sont caractérisées les pièces de monnaie.

	Niveau dans l'arborescence				
M	**0** $\top$	**1** C_i	**2** C_i^j	**3** $C_i^{j,k}$	**Total**
1	60	28	10	222	320
2	3	6	9	90	108
3	0	7	14	92	113
4	1	0	8	20	29
5	1	0	0	16	17
6	0	0	0	8	8
Total	**65**	**41**	**41**	**448**	**595**

TABLE 2 – Répartition par niveau et cardinalité maximale M des contraintes minimales détectées

Analyse qualitative. Tout d'abord, dès le niveau 0, notre méthode permet de retrouver des contraintes fonctionnelles attendues, par exemple, pour les 3 rôles du CIDOC-CRM P1_is_identified_by, P52_has_current_owner et P50_has_current_keeper, indiquant que si un sujet décrit dans la base possède plus d'un identifiant, un propriétaire ou un conservateur, alors on peut en déduire que ces identifiants (respectivement, propriétaires et conservateurs) sont identiques. Concernant le rôle P45_consists_of du CIDOC-CRM (permettant de décrire les matériaux constitutifs d'un objet), il est intéressant de noter qu'une cardinalité maximale de 2 est détectée dès le niveau 1 pour la classe E22_Man_Made_Object. La base de connaissances décrit notamment des médailles constituées d'or et de pierre précieuse (telle l'agate). Pour ce même rôle, une cardinalité maximale de 1 est détectée au niveau 3 pour les pièces de monnaie. Cette information est notamment représentée par la contrainte E22_Man_Made_Object $\sqcap$ $\exists$P2_has_type.{<http://nomisma.org/id/coin>} $\sqsubseteq$ (≤ 1 P45_consists_of). Cette contrainte est détectée bien qu'à certaines pièces la relation P45_consists_of associe deux matériaux ; mais c'est rare (et le plus souvent il

s'agit du même matériau dans deux langues différentes). Un même type de contrainte (avec $M = 1$) est trouvée au niveau 3 pour tous les contextes décrivant des pièces, concernant le rôle `P62_depicts` (ce qui est dépeint sur l'objet). C'est raisonnable car dans le cas d'une pièce de monnaie, on trouve le plus souvent une seule représentation figurative (sur une des deux faces de la pièce), alors qu'une telle contrainte n'est pas valide pour d'autres objets.

Au passage, l'étude de l'ensemble des contraintes extraites par notre méthode a mis en évidence des redondances dans la base, sans doute du fait des choix d'intégration. Dans une phase de post-traitement, la connaissance d'axiomes tel que $\exists$`P2_has_type`.$\{\ldots coin\}$ $\sqsubseteq$ $\exists$`Thing_has_type_Concept`.$\{\ldots moneta\}$ pourrait réduire encore le nombre de contraintes extraites.

6 Conclusion

Nos expérimentations démontrent la faisabilité d'une exploration systématique d'une base de connaissances, à la recherche de contraintes contextuelles de cardinalité maximale, grâce à l'algorithme que nous proposons dans cet article : dans le cas étudié, cela prend moins d'une heure pour une base de connaissances contenant plus de 3 millions de triplets, décrits par une centaine de concepts et plus de 300 rôles. Les propriétés utilisées par notre algorithme font que seules 595 contraintes ont été obtenues, ce qui reste analysable manuellement. Cela nous a permis de vérifier que ces contraintes sont bien pertinentes dans le contexte de la base étudiée. De plus, nos expérimentations démontrent l'importance du contexte dans cette découverte de contraintes.

Il s'agit à notre connaissance de la première proposition de calcul de contraintes contextuelles de cardinalité maximale dans une base de connaissances du web sémantique. Ces grandes bases de connaissances, reflet d'une intelligence collective, sont générées à partir de l'expertise limitée de nombreux contributeurs et souffrent encore, tantôt de lacunes dans les informations, tantôt d'incohérences. Utiliser leurs contenus courants afin de mieux caractériser les connaissances représentées est donc très utile, comme montré dans l'état de l'art : cela permet aux applications qui exploitent ces grandes bases de connaissances de produire des résultats plus fiables.

Nous avons donc pour perspective d'exploiter les contraintes extraites pour calculer la confiance associée à des règles découvertes dans la base de connaissances ainsi enrichie. Mais avant cela, nous travaillons sur des post-traitements pour réduire encore le nombre de contraintes présentées en résultat. Pour cela, nous explorons le potentiel des raisonnements possibles sur la TBox, en particulier comment les relations de subsomption entre classes peuvent éliminer des redondances dans les ensembles de contraintes extraites.

Références

[1] Atencia, M., David, J., Scharffe, F. : Keys and pseudo-keys detection for web datasets cleansing and interlinking. In : EKAW. pp. 144–153. Springer (2012)

[2] Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z. : Dbpedia : A nucleus for a web of open data. In : The semantic web, pp. 722–735. Springer (2007)

[3] Baader, F., Sattler, U. : Expressive number restrictions in description logics. Journal of Logic and Computation 9(3), 319–350 (1999)

[4] Darari, F., Nutt, W., Pirrò, G., Razniewski, S. : Completeness statements about rdf data sources and their use for query answering. In : ISWC. pp. 66–83. Springer Berlin Heidelberg (2013)

[5] Darari, F., Razniewski, S., Prasojo, R.E., Nutt, W. : Enabling Fine-Grained RDF Data Completeness Assessment. In : Web Engineering. pp. 170–187. Springer International Publishing, Cham (2016)

[6] Felicetti, A., Gerth, P., Meghini, C., Theodoridou, M. : Integrating heterogeneous coin datasets in the context of archaeological research. In : EMF-CRM@ICTPDL. pp. 13–27. CEUR-WS.org (2015)

[7] Galárraga, L., Razniewski, S., Amarilli, A., Suchanek, F.M. : Predicting completeness in knowledge bases. In : WSDM. pp. 375–383. ACM (2017)

[8] Galárraga, L.A., Teflioudi, C., Hose, K., Suchanek, F. : Amie : Association rule mining under incomplete evidence in ontological knowledge bases. In : WWW. pp. 413–422. ACM (2013)

[9] Galárraga, L., Hose, K., Razniewski, S. : Enabling Completeness-aware Querying in SPARQL. In : Proceedings of WebDB. pp. 19–22. ACM (2017)

[10] Lajus, J., Suchanek, F.M. : Are All People Married ? Determining Obligatory Attributes in Knowledge Bases . In : WWW (2018)

[11] Pernelle, N., Saïs, F., Symeonidou, D. : An automatic key discovery approach for data linking. Web Semantics : Science, Services and Agents on the World Wide Web 23, 16–30 (2013)

[12] Razniewski, S., Korn, F., Nutt, W., Srivastava, D. : Identifying the extent of completeness of query answers over partially complete databases. In : SIGMOD. pp. 561–576. ACM (2015)

[13] Razniewski, S., Suchanek, F., Nutt, W. : But what do we actually know ? In : 5th Workshop on Automated Knowledge Base Construction. pp. 40–44 (2016)

[14] Soutou, C. : Relational database reverse engineering : algorithms to extract cardinality constraints. Data & Knowledge Engineering 28(2), 161–207 (1998)

[15] Symeonidou, D., Armant, V., Pernelle, N., Saïs, F. : Sakey : Scalable almost key discovery in RDF data. In : ISWC. pp. 33–49. Springer (2014)

[16] Symeonidou, D., Galárraga, L., Pernelle, N., Saïs, F., Suchanek, F. : Vickey : Mining conditional keys on knowledge bases. In : ISWC. pp. 661–677. Springer (2017)

[17] Tanon, T.P., Stepanova, D., Razniewski, S., Mirza, P., Weikum, G. : Completeness-aware rule learning from knowledge graphs. In : ISWC. pp. 507–525. Springer (2017)

[18] Yeh, D., Li, Y., Chu, W. : Extracting entity-relationship diagram from a table-based legacy database. Journal of Systems and Software 81(5), 764–771 (2008)