Gestion de gros volumes de données RDF

Xin HUANG

LIPADE Université Paris Descartes

June 23, 2014

Sommaire

Contexte

1. Augmentation considérable des données du Web, RDF
2. Données provenant de multiple sources autonomes, donc
3. Hétérogènes : sémantique différentes des termes, métadonnées et descriptions
4. Inconsistantes : violation de règles d'intégrité
5. Incertaines : faits et règles parfois incertains

Objectif

1. Collecte des informations pertinentes, cohérentes et complètes.
2. Développer des techniques et approches efficaces et adaptées au Big Data :
   - Récupérer une information cohérente par agrégation et réconciliation de données provenant de sources différentes.
   - Raisonnement en présence de données hétérogènes, inconsistance et incertaines.

Exemple

Considérons les deux sources RDF suivantes :

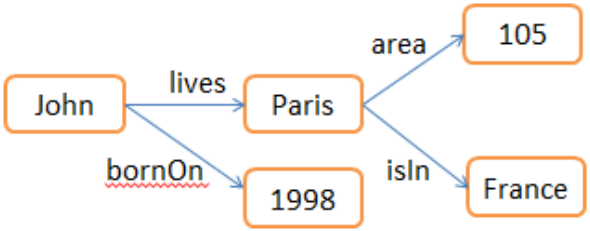


Figure : Source 1

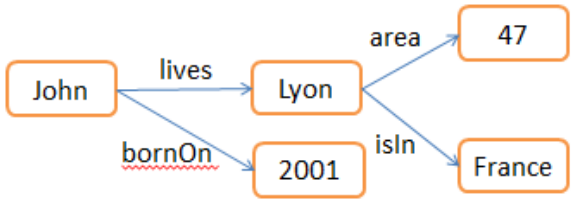


Figure : Source 2

- Règles d'integrité :
- Résultat : $bornOn(John, 1998), bornOn(John, 2001)$?!

Exemple

L'évaluation de requêtes doit opérer sur :

1. Données implicites : Inclure les faits implicites en résultat.
   Exemple : John vit-il en France ?
   Inférence à base règles comme :
   $$lives(P, F) \leftarrow lives(P, V) \wedge isIn(V, F)$$
2. Inconsistance dans les données : ne pas inclure des faits inconsistants dans un même résultat.
   Exemple : quelle est la date de naissance de Jonh ?
   $$bornOn(John, 1998), bornOn(John, 2001).$$
   en présence de la règle de cohérence :
   $bornOn(p, d1) \wedge bornOn(p, d2) \rightarrow d1 = d2$

Stockage de RDF

Stockage centralisé

- En RDBMs(Big triple tables, Property tables, Partition verticale)
- RDF Store native(RDF-3X, Hexastore)

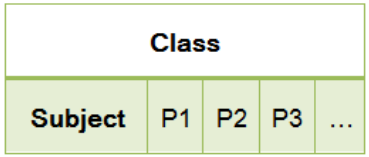


Figure : Property tables

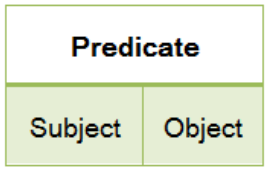


Figure : Partition verticale

Stockage de RDF

Stockage distribué

- RDF textefiles in Hadoop HDFS
- RDF in Hbase

```
Row Key: Subject URI1 {
   Column Family: VALUE {
      Column: (predict1, object1)
      Column: (predict2, object2)
      Column: (predict3, object3)
   }
}
```

Figure 2. S_PO data row

Interrogation de gros volumes de RDF

Hadoop

Hadoop est un framework open source basé sur Java qui permet le stockage et le traitement de gros volume de données

Map-reduce

Map-reduce est un modèle de programmation massivement parallèle adapté au traitement de très grandes quantités de données.

Interrogation de gros volumes de RDF

MapReduce framework

- Une interrogation SPARQL est exécutée dans une série de MapReduce Jobs itératives pour effectuer les jointures entre les patterns.
- Algorithme pour déterminer le nombre de Jobs nécessaires pour répondre à une requête SPARQL.

Articles pour ce problème

1. Storage and Retrieval of large RDF graphe using hadoop and MapReduce. Husain and all, in LNCS, CloudCom 2009.
2. Scalable RDF Store Based on Hbase and MapReduce. Sun and all, in ICACTE 2010.
3. Scalable Sparql querying of large RDF graphes. Huang and all, in VLDB 2011.
4. Heuristic-based query processing for large RDF graphs using cloud computing. Husain and all, in TKDE 2011.

Interrogation de gros volumes de RDF

Exemple d'exécution

Listing 1.1. LUBM Query 12

```
SELECT ?X WHERE {                                                    1
?X rdf:type ub:Chair .                                               2
?Y rdf:type ub:Department .                                          3
?X ub:worksFor ?Y .                                                  4
?Y ub:subOrganizationOf <http://www.University0.edu> }               5
```

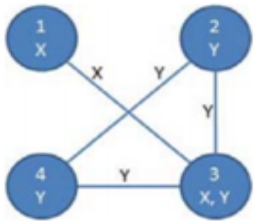


Fig. 1. Graph for Query 12 in Iteration 1

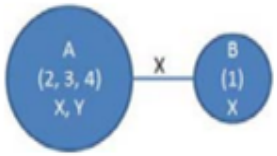


Fig. 2. Graph for Query 12 in Iteration 2

Interrogation de gros volumes de RDF

Optimisation

- Partition de données
- Plan d'exécution

Raisonnement

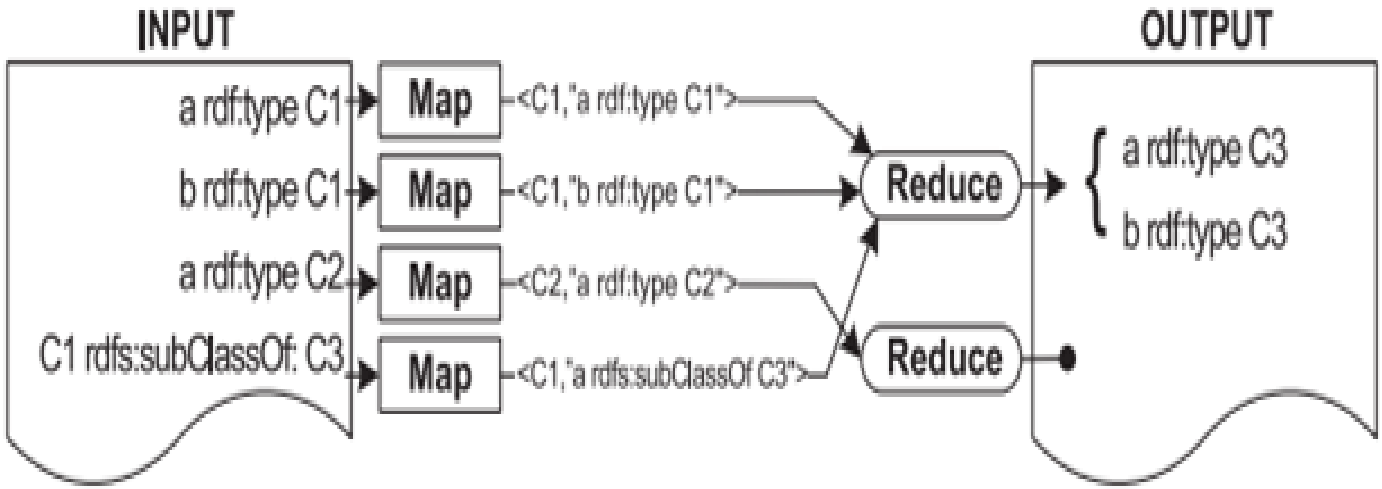
Deux modes d'inférence

1. Inférence en chaîne avant [WebPIE 2011]
   Consiste à appliquer toutes les règles sur l'ensemble des triplets RDF jusqu'à ce que aucun nouveau triplet ne peut être dérivé.
2. Inférence en chaîne arrière [QueryPIE 2011]
   Les règles applicables sont déterminées à l'évaluation des requêtes par le mécanisme de réécriture de requêtes.

Inférence en chaînage avant

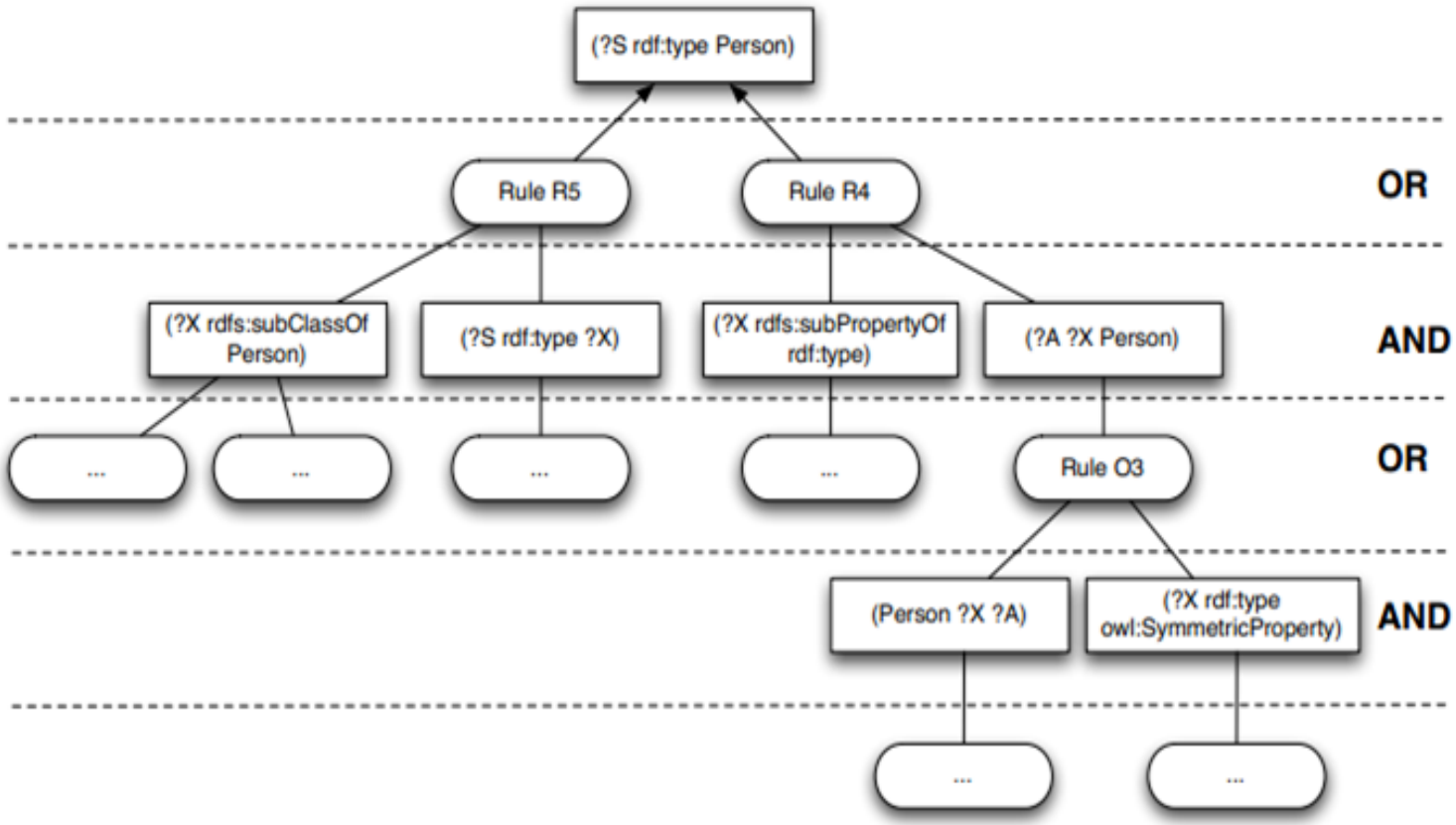
Application en MapReduce de la règle suivante :

9 s rdf:type x & x rdfs:subClassOf y $\Rightarrow$ s rdf:type y



Inférence arrière

Exemple de "and-or reasoning tree"



Données RDF incertaines

Contexte

1. Techniques automatiques d'extraction de bases de connaissances en RDF.
2. Incertitude liées à la complexité du monde réel.

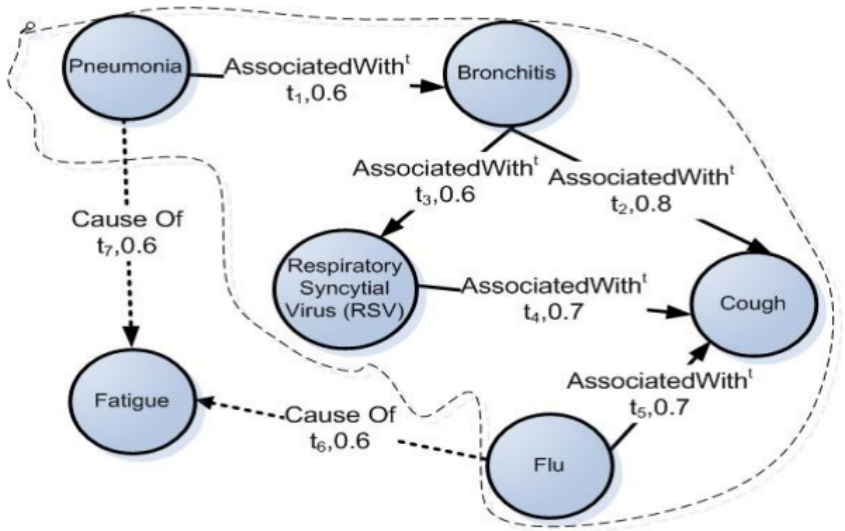


Figure : Exemple d'un graphe RDF incertain

Données RDF incertaines : quelques travaux existants

1. Evaluation de requête
   Query evaluation on probabilistic RDF Databases. Hai HUANG and all, in WISE 2009.
2. Inférence basée sur des règles
   Query-time Reasoning in Uncertain RDF Knowledge Bases with Soft and Hard Rules. Nakashole and all, in VLDS 2012.

Perspectives

1. Les techniques existantes sont limitées face aux importants volumes de données hétérogènes, inconsistantes et incertaines.
2. Ma thèse consiste à proposer des techniques efficaces pour la gestion de gros volumes de données RDF.

Merci de votre attention!