Getting Unique Solution in Data Exchange

Nhung Ngo
supervised by Enrico Franconi
KRDB Research Center, Free University of Bolzano
Bolzano, Italy
ngo@inf.unibz.it

ABSTRACT

A schema mapping is a high-level specification in which the relationship between two database schemas is described. In data exchange, schema mappings are one-way mappings that describe which data can be brought from source data to target data. Therefore, given a source instance and a mapping, there might be more than one valid target instance. This fact causes many problems in query answering over target data for non-conjunctive queries. To make query answering feasible for all queries, we focus on a methodology for extending the original schema mapping to guarantee the uniqueness of target instance corresponding to a source instance. To this end, we introduce a theoretical framework where the problem is transformed to an abduction problem, namely, definability abduction. We apply the framework to relational data exchange setting and solve the problem by pointing out minimal solutions according to a specific semantic minimality criterion.

1. INTRODUCTION

1.1 Motivation

Data exchange deals with transforming data structured under one schema ("source schema") to data structured under another ("target schema"). This transformation must be done according to some specification called a schema mapping. Traditionally, schema mappings are written in the language of source-to-target tuple-generating-dependencies (s-t tgds) [10] to specify that if some positive facts hold in the source, then some other positive facts must hold in the target. In this context, given a source instance and a mapping, there might be more than one valid target instances (solutions) because the target instance might contain additional or unknown facts. Therefore, the problem of query answering over target schema becomes inherently complex. Given a query, its answer now is the certain answer - the set of all tuples appearring in the answer of the query over every valid target instance. Since there might be infinite number of suitable target instances, it is infeasible to compute the certain answer based on its definition. To deal with this problem, the classical data exchange framework uses the notion of universal solutions which are the most general valid target instances; and the problem of computing certain answer reduces to the problem of query answering over universal solutions by rewriting queries. This approach works well with positive relational queries but is not applicable for non-monotonic queries because of the following issues:

- There are queries which are not rewritable over universal solutions [3].
- Certain answer semantics gives non intuitive answers to queries with negation[15].
- Aggregative query answering is trivial under the certain answer semantics [1].

As far as we understand, there is no uniform approach to tackle the above issues.

1.2 Our Approach

The goal of this PhD research is to enrich the data exchange framework that allows general relational and aggregate queries by suggesting "reasonable" amendments to the initial mapping so that from a source instance the new extended mapping will then produce a *unique* target instance. In order to do that, we introduce and solve the novel problem of *definability abduction*, which aims at finding extensions to the initial schema mapping to guarantee the uniqueness of target instance, in a way where the intended meaning of the original mapping is minimally changed.

1.3 Research Questions

In this research, we focus on a framework to extend the schema mapping of a given data exchange problem so that the new extended mapping will then produce a *unique* materialized target instance from a given source instance. Our main research tasks can be summarized as follows:

1. Formalize the problem of extending a schema mapping by:
   - Defining the problem, its solutions and
   - Providing criteria to characterize "good" solutions
2. Tackle the problem in different data exchange settings by:
   - Finding solutions
   - Characterizing the language of extended schema mappings such that inexpensive and SQL-transformable ones are preferred
   - Considering query answering over the new data exchange setting with the extended schema mapping and comparing it with query answering over the classical data exchange framework

Proceedings of the VLDB Endowment, Vol. 6, No. 12
Copyright 2013 VLDB Endowment 2150-8097/13/10... $ 10.00.

- Studying the complexity of finding solutions

3. Compare our approach with different approaches in data exchange which aim to go beyond (union of) conjunctive queries

2. RELATED WORKS

2.1 Data Exchange

The problem of data exchange was formally defined in [10] as the problem of transforming data structured under a source schema into data structured under a target schema w.r.t a mapping consisting of dependencies. The main task of data exchange is materializing a valid target instance (called a solution) which satisfies all dependencies from a complete source data. Most of the results in the literature consider relational data exchange where mapping is a set of s-t tgds and source/target constraints are either tdgs or equality-generating dependencies. Obviously, these mappings lead to infinitely many solutions. Among them, *universal solutions* are good candidates to be materialized since they are the 'most general' solutions, , i.e. every other solutions can be homomorphically mapped to them. Fagin et. al. [10] showed that an universal solution (if exists) can be generated in polynomial time through the *chase* procedure if target dependencies are weakly acyclic tgds. Universal solutions possibly contain redundant data and therefore, the notion of a *core* solution which is the most 'optimal' universal solution was introduced in [11] and then raised an active research line about effectively generating the core in practice [21, 18].

Another important problem in data exchange is query answering over the target schema. As it was mentioned in Introduction, in data exchange the *certain answer* semantics is considered. It has been shown that in the relational setting, the certain answer of a (union of) conjunctive query can be obtained simply by evaluating it over a universal solution and dropping tuples containing null values.

2.2 Closed World Data Exchange

Problems with the certain answer semantic in data exchange were pointed out in literature [3, 10, 1]. Libkin [16] proposed a notion of Close World Assumption (CWA) solution to overcome some of those problems such as: non rewritability and trivial semantic for queries with negation. Based on the intuition of Close World Assumption, a solution is a CWA one if: (i) it contains only "justified" atoms inferred from the source instance using the given mapping; (ii) justifications for atoms are optimal to prevent excessive null values; (iii) it does not have new facts compared to what can be inferred from the source data. Query answering was then considered only over CWA solutions. It was shown that under this semantics the mentioned problems disappear. Complexity results on this work can be summarized as follows:

- There is data exchange setting such that it is undecidable whether a given source instance has a CWA-solution.
- In weakly acyclic data exchange settings, CWA-solutions can be computed in polynomial time.
- Under some restriction, the problem of evaluation of first-order queries under the CWA semantic has co-NP data complexity.

Motivated by some scenarios where the CWA-solution-based semantic does not give intuitive answers, Libkin et. al. [17] introduced a combination approach where users are allowed to control which positions of atoms in the target may be considered as *open*, and which positions may be considered as *closed*.

To deal with *aggregative queries*, Afrati and Kolaitis [1] proposed a strict version of the CWA semantic. In this work, the certain answer of a query is evaluated under the set of all endomorphic images of the canonical universal solution. The range semantics of an aggregate query is the greatest lower bound and the least upper bound of the values that the query takes over such images. It was shown that there are polynomial-time algorithms for computing the range semantics of every scalar aggregation query.

Inspired by semantics for deductive databases, Hernich also introduced a generalized version of CWA-solution namely GCWA*-solution [15] that is basically unions of inclusion-minimal solutions. He claimed that in comparison with other CWA-semantics, query answering with GCWA*-semantic is more intuitive and invariant under logically equivalent schema mappings. On the other hand, there are data exchange settings and Boolean queries for which query evaluation under the semantic is undecidable.

2.3 Knowledge Base Exchange

Knowledge base exchange is a new line of research focusing on exchanging incomplete data specified by knowledge bases. This is the problem of generating target knowledge base corresponding to a source knowledge base with respect to a mapping. The problem was firstly introduced in [6] where relational knowledge bases were considered and then studied in [5, 4] where description logic (DL) knowledge bases were examined. In the DL setting, they argued that standard universal solutions in which TBox is empty may lead to exponentially large target ABoxes; and therefore should not be used. To overcome this problem, a weaker notion of (universal) Q-solution, where Q is a query language, was introduced and union of conjunctive queries (UCQ) was the first considered language.

One could realize that the knowledge base exchange theory has to cope with similar issues in data exchange such as many possible solutions and only some classes of queries are applicable. Therefore, we believe that the approach to get the definability of target predicates also can be applied in knowledge base exchange to have a unique target solution.

3. CONTRIBUTION

3.1 Definability Abductive Problem in Data Exchange

Firstly, we provided a theoretical framework to transform the problem of extending a schema mapping in data exchange into an *abduction problem* based on the idea of *definability*.

How can we precisely characterize the case when the target is uniquely defined given a source instance? This semantical property is given by the notion of *implicit definability* [8, 19]. Intuitively, we say that a predicate p is implicitly definable from a set of predicates P under a theory Σ once the extensions of the predicates from P are fixed in a model of Σ then we are certain that the extension of p is fixed as well. This is exactly what we need in the case of data exchange. That is, whenever we have a source instance, we want the schema mapping Σ to ensure that the target instance is uniquely identified. In other words, each target predicate needs to be implicitly definable from the source predicates under the schema mapping. From the intuition, one can verify this property by checking whether or not

$$\Sigma \cup \widetilde{\Sigma} \models \forall \bar{x}. p(\bar{x}) \leftrightarrow \tilde{p}(\bar{x}),$$

where $\widetilde{\Sigma}$ is obtained from Σ by replacing all source predicates with new predicates with the same arity and $\tilde{p}$ is a new predicate with the same arity as p .

Obviously, in case of the data exchange setting that does not guarantee a unique solution, the entailment does not hold for some target predicates. This leads us to the idea of adding some constraints to Σ to have the entailment for every target predicate. Indeed, this purpose coincides with the meaning of an abductive reasoning task [20, 2] in which an explanation of a given fact (the above definability entailment) must be found based on theory Σ. We call this problem as *definability abduction*.

Definition 1. Given a set of predicates S, a predicate p and a theory Σ, the tuple $(\mathcal{S}, p, \Sigma)$ is said to be a *definability abductive problem* if it holds that

$$\Sigma \cup \widetilde{\Sigma} \not\models \forall \bar{x}.p(\bar{x}) \leftrightarrow \tilde{p}(\bar{x}),$$

where $\tilde{p}$ is a new predicate with the same arity as p and $\widetilde{\Sigma}$ is obtained from Σ by replacing all predicates that are not in $\mathcal{S}$ with new predicates with the same arity.

Definition 2. A set of sentences Δ is called a *solution* to $(\mathcal{S}, p, \Sigma)$ if

$$(\Sigma \cup \Delta) \cup (\widetilde{\Sigma} \cup \widetilde{\Delta}) \models \forall \bar{x}.p(\bar{x}) \leftrightarrow \tilde{p}(\bar{x}),$$

where $\widetilde{\Delta}$ is obtained from Δ by replacing all predicates that are not in $\mathcal{S}$ with new predicates with the same arity.

As in the classical abduction, definability abduction may have many solutions. Among them, we prefer those solutions that are less informative with respect to Σ than other solutions. Put it in other words, those solutions which have more common models with Σ than others are preferred.

Definition 3. An abductive solution Δ is called Σ*-minimal* if for every Δ' such that Δ' is a solution, it holds that

$$\Sigma \cup \Delta \models \Delta' \Rightarrow \Sigma \cup \Delta' \models \Delta.$$

Besides, Beth [8] also pointed out that implicit definability implies explicit definability in first order logic. It means that once we find a solution for the definability abduction problem, we can construct the explicit definitions of the target predicates and therefore can generate the unique target instance of a given source instance based on those definitions.

3.2 A case study: Relational Data Exchange

At the early stage of the research, we applied the framework to deal with a data exchange setting where schema mappings contain only tgds and source/target constraints are ignored. We firstly observed that under this setting, the definability of the target predicates does not exists and therefore their related abduction problems need to be solved.

THEOREM 1. *Given a relational data exchange setting that contains a source schema $\mathcal{S}$, a target schema $\mathcal{T}$ and a schema mapping Σ, for every $p \in \mathcal{T}$, $(\mathcal{S}, p, \Sigma)$ is a definability abductive problem.*

Next, we specified the language to extend schema mapping namely *sts fragment* which allows s-t tgds and t-s $CQ-to-UCQ^{=}$ dependencies. This language is reasonable because s-t tgds would specify what additional information should be brought from source to target while t-s CQ-to-$UCQ^{=}$ dependencies say what information is enough in the target. Obviously, this is also the language of new mappings and possible to transform to some SQL scripts.

Within this setting, instead of using abduction techniques to find minimal solutions, we actually came up with some intuitive solutions and proved that they are minimal. More precisely, solutions we obtained in this setting are as follow:

1. *Schema mapping containing only full tgds*: Full tgds are tdgs without existential quantifier and therefore we can always assume that such a mapping has the following form:

$$\Sigma = \bigcup_{p_i \in T} \bigcup_{j=1}^{n_i} \{\varphi_j^{p_i}(\bar{x}, \bar{z_j}) \to p_i(\bar{x})\}, \qquad (*)$$

   i.e. it consists of Horn clauses. The assumption is valid since a set of full tgds is logically equivalent to a set of Horn clauses, due to decomposition of the conjunction of the consequents in tgds.

   In this case, one can extend the schema mapping to have a unique solution based on the following theorem.

   THEOREM 2. *Let us consider a data exchange setting with a source schema $\mathcal{S}$, a target schema $\mathcal{T}$ and a schema mapping Σ specified by full s-t dependencies. Then for every $p \in \mathcal{T}$, $\Delta = \{p(\bar{x}) \to \vee_j \exists \bar{z_j} \varphi_j^p(\bar{x}, \bar{z_j})\}$ is a Σ-minimal solution to the corresponding definability abductive problem of p .*

   Intuitively, to make a target predicate definable, one just needs to add a sentence saying that the extension of the predicate is nothing except the extensions of the source formulas appearing in the head of its related mapping rules. This idea is identical to the idea of circumscription [22] which aims to minimizes the extension of predicates according to a theory.

   Example 1. Consider a data exchange setting where $\mathcal{S} = \{Manager(\cdot), Employee(\cdot)\}$, $\mathcal{T} = \{Staff(\cdot)\}$, and Σ is the set of following full-tgds:

$$Manager(x) \to Staff(x)$$
$$Employee(x) \to Staff(x)$$

   Then $\Delta = \{Staff(x) \to (Manager(x) \vee Employee(x))\}$ is a $\Sigma - minimal$ solution to $(\mathcal{S}, Staff, \Sigma)$

2. *Schema mapping containing only embedded tgds*: Embedded tgds are tdgs with existential quantifiers. The problem becomes more complicated in this case because intuitively existential values are not necessarily bounded by some predicates. Therefore, we showed that in order to obtain definability of a target predicate, its definition should be explicitly given in the extending theory.

   THEOREM 3. *Let us consider a data exchange setting with a source schema $\mathcal{S}$, a target schema $\mathcal{T}$ and a schema mapping Σ that is a set of embedded s-t tgds. Then for every $p \in \mathcal{T}$, $\Delta = \{p_s \leftrightarrow p\}$ is a Σ-minimal solution to the corresponding definability abductive problem of p where p_s is a fresh source predicate that does not appear in Σ.*

   Example 2. Consider a data exchange setting where $\mathcal{S} = \{Person(\cdot), Phone(\cdot, \cdot)\}$, $\mathcal{T} = \{Contact(\cdot, \cdot)\}$, and $\Sigma = \{Person(x) \to \exists y Contact(x, y)\}$.

   Then $\Delta = \{Contact(x, y) \leftrightarrow Phone(x, y)\}$ is a $\Sigma - minimal$ solution to $(\mathcal{S}, Contact, \Sigma)$.

3. *Schema mapping containing tgds in general*: We proved that in the general case, solutions are obtained by combining solutions from the above cases. We also provided a polynomial algorithm to generate such combined solutions.

Besides, the above solutions give us explicitly the definitions of target predicates under extended schema mappings. Based on the syntax of sts fragment, obviously target predicates are (union of) conjunctive queries over source predicates. Therefore, it is possible to transform these definitions to SQL scripts for generating the unique solution of a source instance.

Some of the results we have obtained can be found in [14].

4. FUTURE WORKS

We have considered the problem of gaining definability of target predicates over source predicates in data exchange. We have defined the problem as an abduction task and approached it by stating a minimal criterion and a syntax restriction for its solutions.

The current framework works without source and target dependencies. By adding source dependencies only, we believe there is no change in finding minimal solutions. However, target dependencies may affect current proposed minimal solutions. Therefore, finding conditions to guarantee the minimal solutions under target dependencies is an interesting open challenge.

Beside the relational database setting, in future, we also would like to explore the definability abductive problem in some fragments of description logic where the problem coincide with the TBox abduction problem. At the first stage, we have focused on DL-Lite TBox.

Regarding to the language of minimal solution, we proposed sts fragment. In general, the fragment is undecidable. However, by adding some conditions related to the problem such as: heads of formulas in solutions are always target atoms, each target atom appears in only one formula, we hope that we can investigate the complexity of the language or at least can check whether or not a solution satisfies target dependencies.

The inversion of schema mappings has been considered as one of the basic operators in data management. The inversion problem is the problem of finding an inverse of a given schema mapping such that one can generate source instances from target instances. Recently, there are related several theories which have been proposed such as: Fagin-inverse [13], quasi-inverse [12] and maximum recovery [7]. Among them, the definition of inversion as maximum recovery of Arenas et. al. [7] is the most general definition. We realized that in case of full tdgs, the definition of target predicates coincides with the maximum recovery of the schema mapping. Therefore, we intend to investigate the connection in case of embedded tgds and more general schema mapping languages.

We also plan to implement a data exchange tool which suggests users change their schema mapping for the uniqueness of target instances. This tool can be either an independent tool or integrated into some data exchange tools such as Clio [9].

5. REFERENCES

[1] F. N. Afrati and P. G. Kolaitis. Answering aggregate queries in data exchange. In *PODS*, pages 129–138, 2008.

[2] A. Aliseda-Llera. *Seeking explanations: abduction in logic, philosophy of science and artificial intelligence*. PhD thesis, Stanford, CA, USA, 1998. UMI Order No. GAX98-10072.

[3] M. Arenas, P. Barceló, R. Fagin, and L. Libkin. Locally consistent transformations and query answering in data exchange. In *PODS*, pages 229–240, 2004.

[4] M. Arenas, E. Botoeva, D. Calvanese, V. Ryzhikov, and E. Sherkhonov. Exchanging description logic knowledge bases. In *KR*, 2012.

[5] M. Arenas, E. Botoeva, D. Calvanese, V. Ryzhikov, and E. Sherkhonov. Representability in dl lite r knowledge base exchange. In *Description Logics*, 2012.

[6] M. Arenas, J. Pérez, and J. Reutter. Data exchange beyond complete data. In *Proceedings of the thirtieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, PODS '11, pages 83–94, New York, NY, USA, 2011. ACM.

[7] M. Arenas, J. Pérez, and C. Riveros. The recovery of a schema mapping: Bringing exchanged data back. *ACM Trans. Database Syst.*, 34(4):22:1–22:48, Dec. 2009.

[8] E. Beth. On Padoa's method in the theory of definition. *Indagationes Mathematicae*, 15:330–339, 1953.

[9] R. Fagin, L. M. Haas, M. A. Hernández, R. J. Miller, L. Popa, and Y. Velegrakis. Clio: Schema mapping creation and data exchange. In *Conceptual Modeling: Foundations and Applications*, pages 198–236, 2009.

[10] R. Fagin, P. G. Kolaitis, R. J. Miller, and L. Popa. Data exchange: semantics and query answering. *Theor. Comput. Sci.*, 336(1):89–124, 2005.

[11] R. Fagin, P. G. Kolaitis, and L. Popa. Data exchange: getting to the core. *ACM Trans. Database Syst.*, 30(1):174–210, Mar. 2005.

[12] R. Fagin, P. G. Kolaitis, L. Popa, and W. C. Tan. Quasi-inverses of schema mappings. *ACM Trans. Database Syst.*, 33(2), 2008.

[13] R. Fagin and A. Nash. The structure of inverses in schema mappings. *J. ACM*, 57(6):31:1–31:57, Nov. 2010.

[14] E. Franconi, N. Ngo, and E. Sherkhonov. The definability abduction problem for data exchange - (abstract). In *RR*, pages 217–220, 2012.

[15] A. Hernich. Answering non-monotonic queries in relational data exchange. In *ICDT*, pages 143–154, 2010.

[16] L. Libkin. Data exchange and incomplete information. In *Proceedings of the twenty-fifth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, PODS '06, pages 60–69, New York, NY, USA, 2006. ACM.

[17] L. Libkin and C. Sirangelo. Data exchange and schema mappings in open and closed worlds. In *Proceedings of the twenty-seventh ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, PODS '08, pages 139–148, New York, NY, USA, 2008. ACM.

[18] G. Mecca, P. Papotti, and S. Raunich. Core schema mappings: Scalable core computations in data exchange. *Inf. Syst.*, 37(7):677–711, 2012.

[19] A. Nash, L. Segoufin, and V. Vianu. Views and queries: Determinacy and rewriting. *ACM Trans. Database Syst.*, 35:21:1–21:41, July 2010.

[20] G. Paul. Approaches to abductive reasoning: An overview. *AI Review*, 7:109–152, 1993.

[21] R. Pichler and V. Savenkov. Towards practical feasibility of core computation in data exchange. In *LPAR*, pages 62–78, 2008.

[22] R. Reiter. Circumscription implies predicate completion (sometimes). In *AAAI*, pages 418–420, 1982.