Reaching definability via Abduction

Evgeny Sherkhonov

thesis is done at Free Univesity of Bozen-Bolzano
TU Dresden
Supervisors: Prof. Enrico Franconi, Prof. Steffen Hölldobler

February 21, 2012

Background

- Query answering under constraints
- Definability
- Abduction
- Data exchange

Definability Abduction

- Problem formalization
- Definability abduction in data exchange
- Definability abduction in $\mathcal{ALC}$

Conclusion and future work

Data access under constraints

There are different types of constraints.

- Ontologies
  They provide conceptual view of the data

- Schema mappings
  They provide the specification how different schemas interact

Our assumptions

- Conceptual schema has a richer vocabulary than the data stores

  $\rightsquigarrow$ Standard DB technologies are not applicable

- DBox (constraints with exact views): Complete information of only some terms is available (from databases)

  $\rightsquigarrow$ Query answering is hard in general.

How to answer queries under constraints?

Common approach: Query rewriting

- Given Q over $\sigma(KB, DB)$.
- Rewrite Q into Q', which is over $\sigma(DB)$, such that $answer(Q) = answer(Q')$.
- Answer Q' using SQL.

Depends on KB and Q:

- KB is expressed in *DL-Lite* and Q is a $(U)CQ$.
- KB is expressed in FOL and Q is *implicitly definable from* $\sigma(DB)$.

Example

- KB:

$$\begin{aligned} Researcher(x) &\rightarrow MSc(x) \vee PhD(x) \\ MSc(x) &\rightarrow Researcher(x) \\ PhD(x) &\rightarrow Researcher(x) \\ MSc(x) &\rightarrow \neg PhD(x) \end{aligned}$$

- DB:

$$\begin{aligned} Researcher &= \{Leonard, Sheldon, Howard\} \\ PhD &= \{Leonard, Sheldon\} \end{aligned}$$

$Q(x) = MSc(x)$ is *implicitly definable from Researcher* and PhD.
Answer $MSc = \{Howard\}$

Definability

Definition 1 (Implicit definability)

φ is *implicitly definable from* $\mathcal{P}$ under KB if
$\forall I, J \in M(KB): \; D^I = D^J$ it holds that

$$\cdot^I|_{\mathcal{P}} = \cdot^J|_{\mathcal{P}} \;\Rightarrow\; \varphi^I \equiv \varphi^J$$

I.e. a formula is definable if its truth value solely depends on the domain and the extensions of predicates in $\mathcal{P}$.

Query rewriting framework

- Check consistency of KB and DB;
- Check implicit definability of Q from $\mathcal{P}_{DB}$ under KB;
- Compute Craig's interpolant (a.k.a rewriting);
- If the rewriting is domain independent, execute in SQL.

What is Abduction?

- “the action of forcibly taking someone away against their will” [Oxford dictionary]

What is Abduction?

- Type of reasoning for deriving *explanations* to facts.

Definition 2 (Abductive problem)

A pair $\langle \Sigma, q \rangle$ such that $\Sigma \not\models q$

Definition 3

α is a *solution* if $\Sigma \cup \{\alpha\} \models q$

- *consistent* if $\Sigma \cup \{\alpha\}$ is consistent,
- *relevant* if $\alpha \not\models q$,
- *conservative* if $\sigma(\alpha) \subseteq \sigma(\Sigma, q)$.

Other restrictions

- Syntactic restriction
- Preference criteria:
  - *minimality:* $(\alpha \models \beta \Rightarrow \beta \models \alpha)$
  - Σ-*minimality:* $(\Sigma \cup \alpha \models \beta \Rightarrow \Sigma \cup \beta \models \alpha)$
  - *basicness*: no relevant solution for $\langle \Sigma, \alpha \rangle$

Data exchange

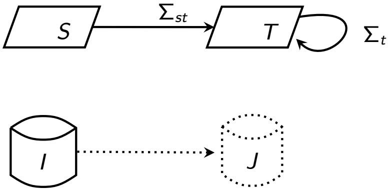

Figure: Data exchange problem.

- Data exchange problem:
  - Translate the data structured under S to the data under T in as precise as possible way.
  - Query answering over T must be consistent with the source information.
- Data exchange setting: $(S, T, \Sigma_{st}, \Sigma_t)$, where Σ_{st} is a *source to target schema mapping*, Σ_t is target constraints.

Schema mapping

Data exchange setting $(S, T, \Sigma_{st}, \Sigma_t)$
Schema mappings given by *dependencies*

- source to target $\mathcal{L}_1$-to-$\mathcal{L}_2$-dependency:

$$\varphi(\bar{x}, \bar{y}) \rightarrow \exists \bar{z}.\psi(\bar{x}, \bar{z}),$$

  where

  - φ is a $\mathcal{L}_1$-formula over S,
  - ψ is a $\mathcal{L}_2$-formula over T.

- Σ_{st} is expressed by source to target CQ-to-CQ dependencies,
- Σ_t is expressed by target to target CQ-to-CQ dependencies, plus equality generating dependencies over T.

$$\varphi(\bar{x}) \rightarrow x_i = x_j.$$

Data exchange

Example 4

$\Sigma_{st}: \; P(x,y) \rightarrow \exists z(Q(x,z) \wedge Q(z,y))$
$I = \{P(a,b)\}$

- $\{Q(a,b), Q(b,b)\}$,
- $\{Q(a,\bot), Q(\bot,b)\}$,
- $\{Q(a,\bot_i), Q(\bot_i,b) \mid 1 \leq i \leq n\}$.

- For a source instance I there might be many solutions. Which one to materialize?
  $\rightsquigarrow$ Universal solution (can be homomorphically embedded into all other solutions)
- What is the semantics of query answering?
  $\rightsquigarrow$ Certain answers

$$certain(Q,I) = \bigcap \{Q(J) \mid J \text{ is a solution}\}$$

Outline

What if a query is not definable?

- Assume Q is *not* definable from $\mathcal{P}$ under Σ.
- and we want to *make* it definable (Why? See later). How?

Definition 5 (Definability abductive problem)

A DAP is a tuple $\langle \Sigma, \mathcal{P}, Q \rangle$ such that

$$\Sigma \cup \widetilde{\Sigma} \not\models Q \leftrightarrow \widetilde{Q},$$

where $\widetilde{\cdot}$ is replacement of predicates other than from $\mathcal{P}$ by fresh ones.

Definability abduction

Definition 6

Δ is a solution to a DAP if

$$\Sigma \cup \Delta \cup \widetilde{\Sigma} \cup \widetilde{\Delta} \models Q \leftrightarrow \widetilde{Q}.$$

It is

- *consistent* if $\Sigma \cup \Delta$ is,
- *relevant* if $\Delta \cup \widetilde{\Delta} \not\models Q \leftrightarrow \widetilde{Q}$,
- *conservative* if $\sigma(\Delta) \subseteq \sigma(\Sigma, Q) \cup \{=\}$

Example

- Σ :

$$\begin{aligned}\forall x(s(x) \rightarrow g(x) \vee u(x)),\\ \forall x(g(x) \rightarrow s(x)),\\ \forall x(u(x) \rightarrow s(x)),\end{aligned}$$

- $\mathcal{P} = \{s, u\}$,
- $Q = g$.

Definability abductive solutions:

- $\forall x.g(x) \rightsquigarrow$ Irrelevant
- $\forall x.(g(x) \leftrightarrow \neg s(x)) \rightsquigarrow$ Inconsistent
- $\forall x(g(x) \rightarrow \neg u(x)) \rightsquigarrow$ Consistent, relevant

Constraints

Similarly to classical abduction the following has to be taken into account:

- Syntactic restriction
- Preference criterion

What are these restrictions?
It depends on particular instances.

- In data exchange: dependencies.
- In $\mathcal{ALC}$: concept inclusions.

DAP in data exchange

Why we need definability in data exchange?

- Odd anomalies of certain answering semantics.
  Consider $\mathcal{M} = (\{P\}, \{P'\}, \Sigma)$ with Σ:

  $$\forall x, y(P(x, y) \to P'(x, y)).$$

  a source instance $I = \{P(a, a)\}$ and

  $$Q(x) = \forall y(P'(x, y) \to P'(y, x)).$$

  We expect the answer $\{a\}$.
  However, $certain_{\mathcal{M}}(I, Q) = \emptyset$!
- Note if we add $\forall x(P'(x, y) \to P(x, y))$ to Σ, then the target instance is fully defined. $\rightsquigarrow$ Q will be answered correctly.

Non rewritability

- Consider $\mathcal{M} = (\{G, R\}, \{G', R'\}, \Sigma)$ with

$$\Sigma = \{G(x, y) \to G'(x, y), R(x, y) \to R'(x, y)\}.$$

  Then

$$Q(x) = R'(x) \vee \exists y \exists z (R'(y) \wedge G'(y, z) \wedge \neg R'(z))$$

  is not FO rewritable over a universal solution!

- If we add $G'(x, y) \to G(x, y), R'(x, y) \to R(x, y)$ to Σ, then the target instance is fully defined and Q can be answered correctly.

Target is not definable from source

- Observe, the target schema is not implicitly definable from the source schema.
- Can we amend the schema mappings Σ such that T becomes definable from S?
- Any data exchange setting $= (S, T, \Sigma)$ is a *definability abductive problem* with the DAP query $\bigwedge_{q \in T} q(\bar{x}_q)$
- What is the syntactic restriction?
  Target-to-source dependencies $\rightsquigarrow$ tableau and resolution techniques are hardly applicable
- Preference criterion?
  Σ-minimality: Δ_1 is minimal if $\Sigma \cup \Delta_1 \models \Delta_2 \Rightarrow \Sigma \cup \Delta_2 \models \Delta_1$
  Thus, we concentrate on finding minimal solutions only

Σ_{st} is full, $\Sigma_t = \emptyset$

Shape of solutions.

- *CQ*-to-*CQ* solutions.
  - There is a data exchange setting which does not admit any relevant consistent *CQ*-to-*CQ* DAP solution.
- *CQ*-to-$CQ^=$ solutions.
  - Minimal relevant consistent *CQ*-to-$CQ^=$ DAP solutions are among $\Delta_j = \{p_i(\bar{x}) \to \exists \bar{y}.\varphi_i^j(\bar{x}, \bar{y}) \mid 1 \leq i \leq n\}$, $1 \leq j \leq k_i$
  - problems: difficult to find a minimal one; there might be source instances for which there is no data exchange solution under $\Sigma_{st} \cup \Delta$.

CQ-to-$UCQ^{=}$ solutions

- $\Sigma = \{\varphi_i^j(\bar{x}, \bar{y}) \to p_i(\bar{x}) \mid 1 \leq j \leq k_i, 1 \leq i \leq n\}$,
- There is a unique minimal t-s CQ-to-$UCQ^{=}$ solution:

$$\Delta = \{p_i(x) \to \bigvee_{1 \leq i \leq n} \exists \bar{z}_j \varphi_i^j(\bar{x}, \bar{z}_j)\}.$$

- The problem is gone.

Embedded schema mappings

Now consider the case of embedded schema mappings.

- There is a pure embedded data exchange setting which does not admit relevant consistent t-s CQ-to-$(U)CQ$ solutions.
  Example: $p(x) \to \exists y.q(x,y)$

How to get definability of T from S in this case?

- Equate existential variables with universal variables:
  $q(x,y) \to p(x) \wedge x = y \rightsquigarrow$ not intuitive
- Introduce new source predicates which give values for existential variables:
  $q_s(x,y) \leftrightarrow q(x,y)$,
  it will imply the source dependency: $p(x) \to \exists y.q_s(x,y)$
  $\rightsquigarrow$ conservativeness criterion is sacrificed

These solutions are minimal!

Adding source and target constraints

- CQ-to-$(U)CQ^=$ solutions remain to be solutions with added source and target constraints,
- Source constraints do not influence minimality,
- Target constraints do influence minimality
  $\rightsquigarrow$ one has to find minimal solutions taking into account the target constraints

CWA-solutions

CWA-solutions were introduced to solve similar odd behavior of certain answers semantics.

- $\mathcal{M} = (S, T, \Sigma)$ full schema mapping,
- I source instance and
- Δ a minimal *CQ*-to-$UCQ^{=}$ solution. Then

J is a *CWA*−solution for I under Σ iff J is a solution for I under $\Sigma \cup \Delta$.

$\rightsquigarrow$ *DAP* solution provides formalization of meta-assumptions about *CWA* by means of schema mappings.

DAP in $\mathcal{ALC}$

Definition 7

DAP: $\langle \mathcal{T}, \mathcal{P}, C \rangle$.
A TBox $\mathcal{T}_A$ is a solution:

$$C \equiv_{\mathcal{T} \cup \mathcal{T}_A \cup \widetilde{\mathcal{T}} \cup \widetilde{\mathcal{T}}_A} \widetilde{C},$$

- We show how we can generate solutions to a DAP for $\mathcal{ALC}$.

Algorithm

- Construct a complete tableau for $\langle C \sqcap \dot{\neg} \widetilde{C}, \mathcal{T} \cup \widetilde{\mathcal{T}} \rangle$.
- If closed, then definable. Otherwise let $\mathcal{B}$ be an open branch.
  - If $\{x : E, x : F\} \subseteq \mathcal{B}$ and $\sigma(E), \sigma(F) \subseteq \sigma(\mathcal{T}, C)$, then $E \sqsubseteq \dot{\neg} F \in closure(\mathcal{B})$.
  - If $\{x : E, x : F\} \subseteq \mathcal{B}$ and $\sigma(E), \sigma(F) \subseteq \sigma(\widetilde{\mathcal{T}}, \widetilde{C})$, then $\widetilde{E} \sqsubseteq \dot{\neg} \widetilde{F} \in closure(\mathcal{B})$,
- A $\vdash_{\mathcal{T}}$-solution is an element of $\bigotimes_{\mathcal{B} \in \Gamma_{\mathcal{T}}} closure(\mathcal{B})$
- Generates general concept inclusions $E \sqsubseteq F$, where E and F are from sub-concept closure of $\mathcal{T}$ and C.
- Algorithm is sound: Every $\vdash_{\mathcal{T}}$-solution is a DAP solution.
- Alas, it is incomplete.

Summary

- We have introduced a new problem of gaining definability of a formula from particular set of predicates. This problem arises in the context of query rewriting under general constraints.
- This problem is abductive.
- We have applied it to the problem of data exchange, where there is a need to have the target to be definable from the source.
  - The problem has good solutions of the form t-s CQ-to-$UCQ^{=}$ dependencies for full schema mappings.
  - Embedded schema mappings are bad knowledge bases for definability abduction. Non-conservative solutions can be found though.
- We have compared DAP solutions with recoveries and CWA-solutions.
- We have presented a sound algorithm for DAP in $\mathcal{ALC}$.

Future work

- Complete algorithms for solution generation.
- Explore other scenarios when definability is needed.
- Try other preference criteria.
- Minimal solutions in the presence of target constraints in data exchange.

Thank you!

Bad theories

- $\Sigma = \{r \rightarrow w, \neg r\}$
- $q = w$

Then $\alpha = \neg r \rightarrow w$ is the most reasonable explanation, but still bad.
Therefore, the algorithms might not generate good solutions if the knowledge base is bad.