Deriving Axioms Across Ontologies [*]

Dejing Dou
Computer and Information Science
University of Oregon
Eugene, Oregon 97403

dou@cs.uoregon.edu

Drew McDermott
Computer Science Department
Yale University
New Haven, Connecticut 06520

drew.mcdermott@yale.edu

ABSTRACT

Ontologies play a key role in agent communication and the emerging Semantic Web. Axioms are an important component of ontologies to describe the relationships among the concepts. The current research on ontology mapping and ontology translation mainly focuses on how to map and translate the vocabularies and associated data instances from one ontology to another. However, when extending one ontology using axioms from another, we must confront the problem of translating axioms. In this paper, we show that simple symbol replacement will not solve the problem of axiom translation because of *asymmetry of translation*. Instead we extend our inferential ontology translation for facts (ground atomic formulas) and queries to cover axioms using a method we call *axiom derivation*.

Keywords

Ontologies, the Semantic Web, Agent Communication, Ontology Translation

1. INTRODUCTION

Ontologies, which can be defined as the formal specification of a vocabulary of concepts and the relationships among them, play a key role in both agent communication and the emerging Semantic Web. We can distinguish several different *semantic heterogeneity* problems among multiple agents and web services, which may use different ontologies.

1. *Ontology mapping:* Finding correspondences between the symbols of two ontologies; the correspondences are often equivalences and subset-superset (or subproperty-superproperty) relationships.

2. *Ontology translation:* Translating a dataset (assertions) or a query expressed using one ontology into a form that uses a different ontology.

3. *Theory translation:* Translating the axioms of one ontology into the vocabulary of another, preserving their validity.

Much previous work (see a survey in [4]) has been in the first category; ours [2] has been mainly in the second. In this paper we focus on the third, in which the translation approach for simple assertions by replacing the symbols of one vocabulary with the corresponding symbols from another will not work for axioms in general. For example, in one genealogy ontology G_1, husband and married are two concepts (properties) and there may be a first-order logic (FOL) axiom to describe their relationship:

$$(\forall x, y)\mathsf{husband}(x, y) \rightarrow \mathsf{married}(x, y) \qquad (1)$$

where x and y are universal variables to represent man and woman respectively. Suppose another genealogy ontology G_2 has male_partner and in_marriage as corresponding (mapping) properties. Some facts (assertions) expressed in the language of G_1, can simply be translated into the language of G_2 by replacing corresponding properties:

$$\mathsf{husband}(\mathsf{John}, \mathsf{Mary}) \rightsquigarrow \mathsf{male\_partner}(\mathsf{John}, \mathsf{Mary})$$

$$\mathsf{married}(\mathsf{John}, \mathsf{Mary}) \rightsquigarrow \mathsf{in\_marriage}(\mathsf{John}, \mathsf{Mary})$$

We use a symbol $\rightsquigarrow$ to represent the process of translation. The translations are correct in terms of the semantics of G_1 and G_2, where husband can be thought of as a special kind of male_partner. However, if we use the same technique to translate the axiom (1) of G_1 to G_2, we will get:

$$(\forall x, y)\mathsf{male\_partner}(x, y) \rightarrow \mathsf{in\_marriage}(x, y) \qquad (2)$$

It is obvious that (2) is not always true since a man is a partner of a woman doesn't mean that he must be in a marriage with her; (2) is not a valid axiom in G_2.

What is the reason that the above translation for facts is correct, and the translation from (1) to (2) is not correct? Can we find a formal method for translating axioms from a source ontology to a target ontology that guarantees that the translations are also true in the target ontology?

In this paper, we will give a formal characterization of ontology translation that unifies the translation of facts, queries and axioms in one inferential framework (section 2). We call the translation of axioms from one ontology to another ontology *axiom derivation*. We point out that it is the *asymmetry of translation* that makes axiom derivation difficult. In section 3, we will elaborate how to derive axioms across different ontologies as the extension of our previous work for translating datasets and queries.

2. FRAMEWORK

2.1 Merged Ontologies & Bridging Axioms

In this section we first briefly review our previous work [2] on ontology translation. We assume that in order to translate facts from one ontology to another there must be a *merged ontology* in which all the relevant symbols are allowed to interact. For example, consider the pair of genealogy ontologies G_1 and G_2. The mappings between them can be represented as first order axioms:

$$(\forall x, y)\mathsf{husband}(x, y) \rightarrow \mathsf{male\_partner}(x, y) \qquad (3)$$

$$(\forall x, y)\mathsf{married}(x, y) \leftrightarrow \mathsf{in\_marriage}(x, y) \qquad (4)$$

We call these axiomatic mapping rules *bridging axioms*. We have developed a strongly typed first-order logic language, *Web-PDDL*, to express bridging axioms. These axioms are embedded in a *merged ontology* complete with namespace declarations and type-equivalence rules.

2.2 Inferential Ontology Translation

We will use the symbol $\rightsquigarrow$ to indicate translation: $\alpha \rightsquigarrow \beta$ means that β is the translation of α. We call the ontology O_s that α uses the *source* ontology and O_t, the one β uses, the *target*. In the case of sets of assertions ("datasets"), we stipulate that the translation of α_d is simply the strongest set of assertions, β_d, in O_t entailed by α_d. A consequence of this stipulation is that

$$(KB; \alpha_d) \rightsquigarrow \beta_d \ only \ if \ (KB; \alpha_d) \models \beta_d$$

where we have added to the left-hand sides the symbol KB to refer to the merged ontology with bridging axioms.

If α_q is a query in O_s, its translation is a query β_q in O_t such that any answer (set of bindings) to β_q is also an answer to α_q. In other words:

$$(KB + DB_\beta; \alpha_q) \rightsquigarrow \beta_q \ only \ if \ (KB + DB_\beta; \theta(\beta_q)) \models \theta(\alpha_q)$$

for any substitution θ, where $KB + DB_\beta$ now includes both the merged ontology and the facts from the target data instances. The point is that β_q need not be (and seldom is) *equivalent* to α_q, in the sense that any answer to one is an answer to the other. All we need is that any answer to β_q be an answer to α_q. If we take O_s to be G_2 and O_t to be G_1 (section 2.1), the query $\mathsf{male\_partner}(?x, ?y)$ in G_2 will be translated into the query $\mathsf{husband}(?x, ?y)$ in G_1. But the set of all husbands is not equivalent to the set of all male_partners.

2.3 Asymmetry of Translation

Given our approach, translation exhibits certain asymmetries that one must be wary of. We will subscript the symbol $\rightsquigarrow$ with a "Q" to indicate the query translation ($\rightsquigarrow_Q$), and with a "D" (for "data") to indicate the assertion or dataset translation($\rightsquigarrow_D$). (We leave the subscript off in those cases where the context allows either reading.) The asymmetry is: if β_t is the translation of α_s: $(KB; \alpha_s) \rightsquigarrow \beta_t$ that doesn't mean α_s is the translation of β_t: $(KB; \beta_t) \rightsquigarrow \alpha_s$. Slightly less obviously, if $(KB; P) \rightsquigarrow Q$ we can't conclude $(KB; \neg P) \rightsquigarrow \neg Q$.

Instead (not surprisingly), negation ends up involving the same duality as query translation. Assume that R is an expression which can be derived from KB and $\neg P_s$ by inference. Using the deduction theorem in FOL and considering

that $\neg P_s \rightarrow R$ is an equivalent to $\neg R \rightarrow P_s$, we know that

$$
\begin{aligned}
(KB; \neg P_s) \vdash R \quad &\Leftrightarrow \quad KB \vdash (\neg P_s \rightarrow R) \\
&\Leftrightarrow \quad KB \vdash (\neg R \rightarrow P_s) \\
&\Leftrightarrow \quad (KB; \neg R) \vdash P_s
\end{aligned}
$$

This gives us a way to translate negations. We can think of P_s as a "ground query" ($\theta(P_s) = P_s$): Given P_s, try to find a Q'_t, which satisfies $(KB; Q'_t) \vdash P_s$. But this is just the problem of translating the query P_s: $(KB; P_s) \rightsquigarrow_Q$?.

Therefore, if the query translation of P_s is Q'_t, $\neg Q'_t$ can be derived from KB and $\neg P_s$ by the data translation and vice versa:

$$
\begin{aligned}
(KB; P_s) \rightsquigarrow_Q Q'_t \quad &\Rightarrow \quad (KB; \neg P_s) \rightsquigarrow_D \neg Q'_t \\
(KB; P_s) \rightsquigarrow_D Q'_t \quad &\Rightarrow \quad (KB; \neg P_s) \rightsquigarrow_Q \neg Q'_t
\end{aligned}
$$

Axioms are usually more complex than a typical dataset element, and it would be useful if we could attack this complexity by translating the pieces of a complex formula and composing the results. For conjunctions and disjunctions, composition of translation is straightforward. Translating implications includes an algorithm for translating negations.

3. OUR APPROACH

3.1 Conditional Facts and ICF Axioms

To explain our approach to theory (axiom) translation, we first show how to translate *conditional facts* using our inference engine, OntoEngine [2]. A conditional fact is a formula of the form:

$$P_1 \wedge \cdots \wedge P_i \cdots \wedge P_n \rightarrow Q_1 \wedge \cdots \wedge Q_j \wedge \cdots \wedge Q_m$$

where all $P_i(1 \leq i \leq n)$ and $Q_j(1 \leq j \leq m)$ are ground atomic formulas (facts). All axioms, such as axioms 1, can be put in this form which we called *ICF (Implicative Conjunction Form)*, but of course axioms have quantified variables. It is unusual, but not unheard of, for people to need to express that some facts are true only if some other facts are also true:

$$
\begin{aligned}
precedes(deathof(Roosevelt), \ endof(WW2)) &\rightarrow \\
president(Truman, \ &endof(WW2))
\end{aligned}
$$

"If Roosevelt died before the end of World War 2, then Truman was president at the end of World War 2."

3.2 Conditional Fact Translation

Conditional fact translation is the translation of a conditional fact from the source ontology to the target ontology. It is not as simple as translating both the antecedent and conclusion in the source ontology to the antecedent and conclusion in the target ontology by the data translation with forward chaining. This is a typical example which we need to consider asymmetry of translation since the translation of implications actually includes the translation of negations and disjunctions, which has been discussed in section 2. For example, suppose we have a simple conditional fact:

$$@\mathsf{Onto1} : P(A, B) \rightarrow @\mathsf{Onto1} : Q(A, B)$$

where @Onto1:P means predicate P from Onto1. We want to translate this conditional fact to Onto2. Suppose @Onto1:P(A,B) can be translated to @Onto2:R(A,B) and @Onto1:Q(A,B) can

be translated to @Onto2:S(A,B), both by forward chaining. It doesn't mean that

$$@Onto2 : R(A, B) \rightarrow @Onto2 : S(A, B)$$

is true. However, if there exists @Onto2:R'(A,B) satisfies:

$$@Onto2 : R'(A, B) \rightarrow @Onto1 : P(A, B)$$

we know that

$$@Onto2 : R'(A, B) \rightarrow @Onto2 : S(A, B)$$

is a true statement in Onto2. To get this, we need to do query translation with backward chaining from @Onto1:P(A,B) to a corresponding expression in the target ontology, such as @Onto2:R'(A,B).

It should be obvious that this process yields a valid result, in the sense that the translated fact follows from the original fact and the axioms. If backward chaining from the antecedent fails to find any goals in the target ontology, then the antecedent of the translated conditional fact will be the empty, or false, making the translation itself equivalent to true — and hence useless.

3.3 Axiom Derivation

Axiom derivation for ICF axioms still can be thought of as an inference process, if we can transform the axioms to conditional facts and transform the conditional facts back to axioms. The idea is to substitute Skolem constants for the variables temporarily. (A similar technique was used in [3].) In general, axiom derivation can be broken into three steps:

From ICF axioms to conditional facts: we can use Universal Elimination and Existential Elimination to transform ICF axioms to conditional facts. For instance, suppose that we have an axiom in the source ontology O_s:

$$(\forall x, y)@O\_s : P(x, y) \rightarrow (\exists z)@O\_s : Q(x, z) \wedge @O\_s : R(z, y)$$

We can substitute the universal quantified variables with constants (e.g., Atx and Bty) and substitute the existential quantified variables with uniquified Skolem terms (e.g., Skz01):

$$@O\_s : P(Atx, Bty) \rightarrow$$
$$@O\_s : Q(Atx, Skz01) \wedge @O\_s : R(Skz01, Bty)$$

Conditional facts translation: suppose that the target ontology is O_t and we already have the merged ontology of O_s and O_t. The conditional fact in O_s can be translated to O_t. By backward chaining from @O_s:P(Atx,Bty) and forward chaining from @O_s:Q(Atx,Skz01) and @O_s:R(Skz01,Bty), we may finally get a conditional fact in O_t:

$$@O\_t : S'(Atx, Ctc) \wedge @O\_t : T'(Ctc, Bty) \rightarrow$$
$$@O\_t : U(Atx, Skz01) \wedge @O\_t : V(Skz01, Skd02)$$
$$\wedge @O\_t : W(Bty, Skz01)$$

where Ctc is a constant and Skd02 is a new generated Skolem term by forward chaining.

From conditional facts to ICF axioms: we can use Universal Generalization [3] and Existential Introduction to transform conditional facts back to ICF axioms.

For example, considering the above translated conditional fact in O_t, we can use Universal Generalization to replace all constants which have substituted universal variables with universal variables. Atx, Bty and Ctc can be replaced by x, y and c. We also can use Existential Introduction to replace

all Skolem terms with existential variables. Skz01 and Skd02 can be replaced by z and d. Therefore, the generated ICF axiom looks thus:

$$(\forall x, y, c)@O\_t : S'(x, c) \wedge @O\_t : T'(c, y) \rightarrow$$
$$(\exists z, d)@O\_t : U(x, z) \wedge @O\_t : V(z, d) \wedge @O\_t : W(y, z)$$

It's not so obvious that this procedure works, but we have proved that it does. The detail proof can be found in a long version[1] of our paper.

3.4 Axiom Derivation for Time Ontologies

We have evaluated our axiom derivation algorithm in some real application scenarios, in which complex ontologies always have large sets of axioms. There are several time ontologies in current popular knowledge bases and (semantic) web resources, such as Cyc time [2], SUMO time and OWL-Time (formerly DAML-Time) [3]. They describe the temporal concepts, such as Instant, Interval and durationOf, and their relationships, which are represented as sets of logic axioms (e.g. OWL-Time has around 180 first order axioms.)

Researchers have manually built some mappings among the concepts of some time ontologies, but have not talked about how to represent the axioms in different ontologies. The axioms in the Cyc time ontology can be automatically translated to the OWL-Time ontology with our algorithm.

4. RELATED WORK AND CONCLUSION

To the best of our knowledge, the only other work on axiom (theory) translation is [1]. This work presents a formalism for knowledge translation based on the theory of contexts [3]. The authors define knowledge translation in terms of truth, and propose, as we do, using a theorem prover to perform the translations. However, the paper does not say exactly *how* the theorem-proving process would work.

Based on the inferential ontology translation framework, our paper has pointed out the asymmetry of translation and described an algorithm for theory translation from one ontology to another, which we call axiom derivation. Although our algorithm is provably correct, practical application requires further work on problems of incompleteness and redundancy, because our algorithm does not by itself guarantee that the axioms we produce are sufficient, nor does it avoid producing axioms that are already present in the target ontology.

5. REFERENCES

[1] S. Buvac and R. Fikes. A Declarative Formalization of Knowledge Translation. In *Proceedings of the ACM CIKM conference*, 1995.

[2] D. Dou, D. Mcdermott, and P. Qi. Ontology translation on the semantic web. *Journal of Data Semantics*, II:35–57, 2005.

[3] J. McCarthy and S. Buvac. Formalizing context (expanded notes). In A. Aliseda, R. van Glabbeek, and D. Westerstahl, editors, *Computing Natural Language*. University of Chicago Press, 1997.

[4] N. F. Noy. Semantic integration: A survey of ontology-based approaches. *SIGMOD Record*, 33(4):65–70, 2004.