Annals of the University of Craiova, Mathematics and Computer Science Series
Volume 37(3), 2010, Pages 29–34
ISSN: 1223-6934

A Semiotics Approach of Abduction Ontology Query

Cristian KEVORCHIAN

Abstract. Very important functionalities of ontology management system including ontology mapping, ontology learning, debugging reasoning with inconsistent ontologies, and (structured and unstructured) query answering are important research challenge. Our scenarios is based on semiotic fundamentals of abduction and abduction as computation applied to ontology query.

2010 Mathematics Subject Classification. Primary 68T27; Secondary 68T30.
Key words and phrases. abduction,description logic,semiotics, ontology query, abductiv graph.

1. Semiotic Preliminaires

Abduction as concept was introduced by C.S. Pierce as a logic inference process dedicated to hypothesis forming that provide observed phenomena explanation and involve two stages[9]: the selection and formation of plausible hypotheses. From an epistemological point of view abduction is the ”first stage” of scientific inquiries and of any interpretative process.As a premises identification process, the foundation of interpretative reconstruction of causes is as important as the inventive construction of theories.

In a knowledge acquisition context this process involves the interpolation mapping of perception-action relation based on a reasoning tasks sequence consisting in: induction(assumed to be a general principle that subsumes many facts), abduction(assumed to explain some facts) and deduction(applied to a general principle to infer some facts)[11]. According to C. S. Pierce's[9] approach, the abduction was the fundamental form of logical inference, as a rearrangement of Aristotelian syllogism. In this context, we can say that abduction gather deduction and induction provide a *double-loop learning* in a cognitive system. Solomon Marcus[7] considers that ”...abduction in science (and, to some extent, beyond it too) has its roots in two basic changes that occurred in its evolution in the twentieth century. The first change began one hundred years ago and concerned the decline of the predictive function of science and the rise of its explanatory function; we look less and less for deterministic laws and more and more for explanatory hypothetical-cognitive models and metaphors, which become in this way the main aim of the abduction process. It is well known that prediction does not always imply explanation and explanation may not lead to prediction. The second change occurred in the last decade of the twentieth century and it is related to the increasing length and complexity of the scientific proofs and to the need to improve our choice among various possible hypotheses (conjectures, guesses) to be tested as possible truth”. Popper's approach to the ”Logic of Science” says that the growth of knowledge is due to a procedure of ”trial and error”. On the one hand, the ability to

---

Received June 25, 2010. Revision received August 14, 2010.

solve problems is "a creative ability to produce new guesses, and more new guesses". The task related to the logic of science, on the other hand, is the critical discussion of the methods and the logical criteria of hypothesis falsification and elimination. The Pierce's abducting inference denies the possibility to draw a sharp borderline between "context of discovery" and "context of justification".

The last decade of the twentieth century has witnessed important research in ontological engineering which provides a framework based on Formal Ontology, a branch of the ontological research initiated by Husserl, who followed Kant's line of thought. According to Asuncion Gomez-Perez[2] formal ontology is the systematic, formal, axiomatic development of the logic of all forms and modes of being. It studies the formal properties and the classification of the entities of the world (physical objects, events, etc). The quality of ontologies is crucial for obtaining good results in semantic Web and their construction, integration and evolution depends on the availability of well defined semantics and power reasoning engines. There was a contradiction between the expressive power, as well as the reasoning efficiency provided by Description Logics (DL) and the expressivity of large knowledge bases, required by a good quality ontology. During the past few years, this contradiction has been reduced, thanks to important research.

The latest semantic technology helps industrial DL systems to become more expressive.

DL adds important features, such as numeric-valued functions, to Aristotle's monadic predicates and can be accepted as a framework for reasoning implementation over ontology. More precisely, DL are logics serving for formal descriptions of concepts and roles (relations). These logics were created to formalize semantic networks and frame-based systems. Semantically, they are based on first order logic(FOL), but their language is conceived in order to support practical, modeling and to have also good computational properties. The research in DL is focused on the various DL constructs that are usable for real world applications and on their impact on the complexity of reasoning. Knowledge representation based on DL consists of two components: TBox and ABox. The TBox describes terminology, which is the ontology in the form of concepts and roles definitions, while the ABox contains assertions about individuals using the terms from the ontology. Concepts describe sets of individuals, roles describe relations between individuals. When we use a representation language in a knowledge context, by example description logic (DL), we hope to obtain an attached ontology with low level incoherence. If abduction not allow us to explain inconsistent concepts, than we can use abduction to improve the quality of ontology.

2. Ontology Querying

According to Gruber's approach ontology knowledge is organized based five kinds of components: classes or concepts(a family of entities within a domain), relations(the interaction between concepts), functions(a special case of relations in which the n-th element of the relationship is unique for the n-1 preceding elements), axioms(used to model sentences that are always true- used in an ontology to constrain values of classes) and instances(represent specific individual elements). The classes and relations are organized in taxonomies.

A knowledge base $\mathcal{K} = (T, A)$ model over DL contains two parts: $TBox$ (TerminologicalBox) and $ABox$ (AssertionalBox). Several assertions about concepts and roles belong to the first-while the second consists of specific facts about a particular object obtained from a concept or a particular pair of objects belonging to a particular role. A query to $\mathcal{K}$ is a lambda expression

$$\lambda\overline{x}P_1 \wedge P_2 \wedge ... \wedge P_m \tag{1}$$

where $P_i(i = 1..m)$ are predicates of form C(x) or R(x,y),concept and respectively relation in T and each x and y appear in $\overline{x} = (x_1, x_2, ...x_n)$. We can rewrite the functional form of query over $\mathcal{K}$ with following form[6]:

$$Q(x_1, x_2, ..., x_m) \leftarrow BodyOfQuery \tag{2}$$

where $BodyOfQuery$ has the following expression:

$$\exists x_{m+1}, ..., \exists x_n : (\bigwedge C(x_i) \wedge (\bigwedge R(x_i, x_j)) \tag{3}$$

in which predicate names are associated both concepts and roles in the DL.Let $\{a_1, a_2, ..., a_n\} \subseteq A$ based on define an associate to Q answers space with n^k elements and general form[5]:

$$\theta_i = \{(x_1, a_{i_1}), (x_2, a_{i_2})...(x_k, a_{i_k})\} \tag{4}$$

in which variables of the query are linked with constant symbols a_{i_j}. We can view this link as a substitution $\theta_i = \{(x_1/a_{i_1}), x_2/a_{i_2})...x_k/a_{i_k})\}$ an answer to query Q and we can rewrite as

$$\mathcal{K} \models BodyOfQuery\theta \tag{5}$$

Assume that DL supports nominals-a concept defined by finite enumeration of its elements. Therefor the answer $\theta_i = \{(x_1/a_{i_1}), x_2/a_{i_2})...x_k/a_{i_k})\}$ is a family of paired elements(variables x_j and nominal $\{a_{i_j}\}$) and

$$\mathcal{K} \models \forall x_1, ..., \forall x_k : (x_1\{a_{i_1}\} \wedge ... \wedge x_k\{a_{i_k}\}) \tag{6}$$

Nominals allow us to add the dependencies type inclusion as $\{a_i\} \subseteq \mathcal{C}$ or $\{a_i\} \subseteq \exists\mathcal{R}\{a_j\}$ to $TBox$ which correspond to $ABox$ assertion $\mathcal{C}(a_i), \mathcal{R}(a_i, a_j) \in A$ and $ABox$ is reduced to a nominals collection. Considering all concept names from $TBox$ gather with attached relations obtain a taxonomy. More precisely, the term taxonomy means entities classification focused on terms or objects, in a hierarchical structure according to the sub/super class paradigm. There is only one type of relationship relating these entities, namely the $ISA-relationship$. If we reduce the types of relationships in an ontology to the type ISA dedicated to represent the concepts, the ontology will be equivalent to a taxonomy.

This approach involves a rich description for the components $x_i(i = 1..k)$ of the k-tuple answer[10], based on $DBox$(**D**escription**Box**) of $\mathcal{K}$ to replace the $ABox$ as a description base.More precisely a $DBox$, D is a set of assertions $A(a)$ and $R(a, b)$ where $A \in \sum_D(C) \subseteq N_C$, $R \in \sum_D(R) \subseteq N_R$ and $a, b \in \sum_D(I) \subseteq N_I$ is a family of individuals. The signature $\sum_D \subseteq \sum$, where $\sum = (N_I, N_C, N_R)$ is a signature over a DL, where N_I is a family of individuals names, N_C is the family of atomic concepts name and N_R is the family of roles.

An interpretation $\mathfrak{I} = (\Delta^{\mathfrak{I}}, \cdot^{\mathfrak{I}})$ is a model of D, $\mathfrak{I} \models D$ iff $a^{\mathfrak{I}} = a$ for every $DBox$ individual $a \in \sum_I$ and for every concept or role name $P \in \sum_D(P)$ and for every $u \in \Delta^{\mathfrak{I}}$ and respectively $(u, v) \in \Delta^{\mathfrak{I}} \times \Delta^{\mathfrak{I}}$ iff $P(u) \in D$ and respectively $(u, v) \in P^{\mathfrak{I}}, P(u, v) \in D$[10][8]. We can say that every model of D extension attached to $DBox$ predicate, identified by:

$$\sum\nolimits_D(P) = \sum\nolimits_D(C) \bigcup \sum\nolimits_D(R) \tag{7}$$

is given by the content of $DBox$ and is the same in every model. We must mention that the domain $\triangle^{\mathfrak{I}}$ of the model D is not fixed.

We must mention that whenever a concept is used as an parameter for a query, is not possible to have all predefined names, because not all of them are available at the time of the system generate.

Semantic query needs, complete information from sources of knowledge where available information is often incomplete. In order to solve such incomplete reasoning problem, we try embedded default logic into the description logic knowledge base, meanwhile it prioritized the default rules, which preferred more specific default rules over more general ones. Then, an original incomplete query could be transformed into a complete query relative to the extended knowledge base, by checking default satisfiability of complex concept according to the query.

3. Query Based Abduction

Generally speaking, abduction is formalized as $\Sigma \cup \Delta \models \Gamma$ where background knowledge (Σ), and observations (Δ) are given and explanations (Δ) are to be computed. According to classical logic, this kind of reasoning is ***non secuitur***[1] inference, affirming the consecvent. In [4] different constrains for abduction are presented, where Γ is a knowledge base, and A, B are formulas:

Consistency	$\Gamma + A \not\models \bot$
Minimality	A is a "minimal explanation for B"
Relevance	$A \not\models B$
Explanatoriliness	$\Gamma \not\models B, A \not\models B$

However "we suggest that datalog and its extensions by integrity constrains and abduction should be considered as a starting point for a thorough revision and replacement of established but deficient classical foundations of ISE[2] by a new paraconsistent fundament". If we extend Decker's point of view at enterprise ontologies, then we will obtain tools based on abduction for disambiguation and interpretation of process flows at the enterprise level.

Let L be a DL over a knowledge base $\Gamma \subseteq L$ and Φ an assertion over Γ so that $\Gamma \cup \Phi)$ is consistent. The solution for the knowledge base abduction problems, (Γ, Φ) is a finit set:

$$S = \Phi_i | i \leq n \tag{8}$$

of assertions from TBox or ABox, with

$$\Gamma \cup S \models \Phi \tag{9}$$

All solutions are denoted by $Sol_{\mathrm{K}}(\Phi)$. It is interesting to study the relation between the set of solutions and family of query over K. Let's consider following computational abduction process according to obtain $\alpha_1, \alpha_2, ..., \alpha_n$ concepts which added to K current version make an earlier failed query φ a succeed after all via abduction reasoning process over K.Practically, we have[1]:

- α is an abductive(explanation) query φ given Γ
- $\Gamma \not\models \varphi$ while $\alpha + \Gamma \models \varphi$

Bazed on results from [5] reasoning should start from the abductive query to search solutions in the knowledge base. Proof strategies should allow us to obtain the background knowledge, such that those parts of the knowledge base that cannot contribute

---

[1] An inference or conclusion that does not follow from the premises or evidence

[2] Information System Engeneering

to solving the problem are not considered and thus do not introduce extra computational tasks for reasoning. Practically will build a refutation proof for the negated abductive query, attached to knowledge base.

Let a abduction problem $ABox$ (K, Φ). Consider that the query is generated from the knowledge base. In this situation exist a refutation proof for $\mathrm{K} \models \Phi$. There is a closed tableau tree or a resolution deduction of an empty clause, initiated by $\neg\Phi$, with K as the set of premises:

- 1. **tableau**:There is a tableau T such that:
  (a) the root of T contains all clauses $\tau(\mathrm{K}) \cup \tau(\neg\Phi)$
  (b) T was initiated by expansion of some clause $Cl_{init} \in \mathbf{R}(\tau(\neg\Phi))$
  (c) $A_{FOL} = \neg L(t)|L(t) \in Cl$ where Cl is the set of the leaves of all the open branches of T.
- 2. **resolution** There is a sequence of resolution inference steps with a resulting set of resolvents R, where:
  (a)the initial set of clauses which comprise all $\tau(\mathrm{K}) \cup \tau(\neg\Phi)$
  (b)$\mathbf{R}(\tau(\neg\Phi))$ was the set-of-support for that sequence (c)$A_{FOL} = \neg L(t)|L(t) \in Cl$ where Cl is a resolvent in R

The selection of the initial clause is a relative difficult problem. In the tableau building, we will generate alternative proofs.If the proof succeeds will have no a typical abductive problem, with no additional formula needed to query. In other cases, we can identify formulas by analyzing the possible partial proofs. Observe, that at every stage of a proof it is possible to construct a formula A that forces its completion by closing all open branches of the tableau or enabling derivation of an empty clause through resolution. Such a formula would be complete the proofs of the query, and thus it could be seen as a solution to the translated problem (K, Φ), as $\tau(\mathrm{K}) \cup A \cup \tau(\neg\Phi) \models \perp$ and therefore $K \cup A \models \Phi$ Both computing proceses are sound and complete for FOL.

An approach based on abductive graph,$\mathcal{G} = (V, E)$, whose vertices are terms (variables,individual names, Skolem terms) and edges are labeled with role names[5]:

- 1. If A_{FOL} is derived from a clause Cl,such that Cl is the initial clause expanded on the tableau or one of the clauses in the initial set-of-support then $\mathcal{G} = (V, E)$ is the *abductive graph* associated with the proof of $A_{FOL}\ iff\ V = \{a, b|r(a, b) \in E\}$ and $E = \{r(a, b)|r(a, b) \in \mathcal{A}\} \cup \mu(Cl)$ where $\mathcal{A}$ is a $ABox$ in K
- 2. If A_{FOL} is obtained by connecting a clause Cl to an abductive proof involving the existence of an MGU σ and $\mathcal{G}' = (V', E')$ is the abductive graph associated with the proof of $A_{FOL}\ iff\ V = \{a, b|r(a, b) \in E\}$ and $E = \sigma E' \cup \sigma\mu(Cl)$

With every abductive proof and its FOL-base will associate a single graph, which describes the relationships between all the terms occurring in the proof. The graph is initiated at the start of the proof, by including all role assertions occurring in the ABox of the problem, and later it evolves along the construction of the proof. Each inference step it is extended with new edges and vertices present in the modal core of the connected clause, under the substitution applied to the proof at that step.

4. Conclusion

Query abduction is a interesting form of abductive reasoning over DL ontology. Some scenarios to implement this type of reasoning over DL ontology was analyzed from a computational semiotic perspective. Practically the an abductive query over ontology is obtained through automated proof technic. Syntactic structure of the input not modify the abductive reasoning and implicitly abductive query. Will be necessary to obtain more expressive language which involves extra transformation rules, covering additional constructorsin DL axiomatic structure and will have to revisions at the definitions of an admissible abductive graph and extracting procedure. Skolemization which is no included in DL reasoner an interesting challenge.

References

[1] A. Atocha. Abductive Reasoning. In *Synthese Library volume 330*,Springer, New York (2006), 96–132.

[2] G. Asuncion, F. Mariano and O. Corcho. Ontological Engineering, Springer-Verlag, London, 2004.

[3] H. Decker, A Case for Paraconsistent Logic as a Foundation of Future Information Systems *CAISE '05*, Porto, Portugal (2005), 451–461.

[4] C. Elsenbroich, O. Kutz and U. Sattler. A Case for Abductive Reasoning over Ontologies. OWL: Experiences and Direction (OWLED), Athens (2006). `http://www.webont.org/owled/2006/acceptedLong/submission_25.pdf`

[5] S. Klarman, U. Endriss and S. Schlobach. ABox Abduction in the Description Logic ALC, *Journal of Automated Reasoning*, (February 2010).

[6] D. Lembo and M. Ruzzi. Consistent Query Answering over Description Logic Ontologies, *Web Reasoning and Rule Systems First International Conference*, Innsbruck, Austria, June 7-8 (2007), 194–208.

[7] S. Marcus, Abduction: the double change, *Semiotica* (2005), 287–298.

[8] J. Mei, L. Ma and Y. Pan, Ontology Query Answering on Databases,*The Semantic Web - ISWC 2006 5th International Semantic Web Conference Proceedings* Athens, GA, USA, November 5-9, 2006, Springer Berlin, Heidelberg, (2006).

[9] C.S. Peirce, *Philosophical Writings of Peirce*, Dover Publications Inc., New York, 1955.

[10] I. Seylan, E. Franconi and J. de Bruijn. Effective Query Rewriting with Ontologies over DBoxes, . *Proceedings of the 21st International Joint Conference on Artifical Intelligence*, Pasadena, California, USA, (2009), 923–929.

[11] J.F. Sowa, Research Trends inSCience, *Proceedings of Technology and Mathematics Education(J.R. Chunawala ed.)*, Mumbay, Homi Bhabha (2006).

(Cristian KEVORCHIAN) IT Department, Parliament of Romania-Legislative Council, 13
Septembrie Street 1-3, Bucuresti, 71000, Romania
E-mail address: `cristi@clr.ro`