Abducing Rules with Predicate Invention

Katsumi Inoue[1], Koichi Furukawa[2], and Ikuo Kobayashi[2]

[1] National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan
ki@nii.ac.jp
[2] SFC Research Institute, Keio University
5322 Endo, Fujisawa 252-8520, Japan
{furukawa,ikuokoba}@sfc.keio.ac.jp

Abstract. This paper addresses discovery of unknown relations from incomplete network data using abduction. Given a network information such as causal relations and metabolic pathways, we want to infer missing links and nodes in the network to account for observations. This is implemented in SOLAR, an automated deduction system for consequence finding, using first-order representation of algebraic relations and full-clausal ground formulas of network information. Abduction by SOLAR is powerful enough to infer unknown rules and to realize predicate invention by inferring unknown causes. In particular, we point out the importance of existentially quantified formulas to express hypotheses including new variables representing missing nodes to be introduced.

1 Introduction

Abduction has been applied to scientific discovery, in particular to biological domains, e.g., [15, 8, 16, 9]. Scientific knowledge is often structured in a network form, in which arcs and nodes have important meanings in applications. For example, in biological domains, a sequence of signalings or biochemical reactions constitutes a *pathway*, which specifies a mechanism to explain how genes or cells carry out their functions. However, information of biological networks in public domain such as KEGG [6] is generally *incomplete*. To deal with incompleteness of pathway databases, we need to predict the status of relations which is consistent with the status of nodes [16, 17], or augment missing arcs between nodes to explain observations [15, 8]. These problems are characterized by an abductive problem called *theory completion* [10] or *graph completion* [15], in which unknown status of nodes or missing arcs are augmented to account for observations.

Many current abductive methods have several limitations. Sometimes only single missing arcs can be found so that those explanations with more than one missing arc cannot be obtained. In another case, multiple observations are often given at once rather than they are input sequentially, which may also request abductive systems to infer *multiple missing arcs*. A more difficult case is to abduce *missing nodes* and arcs connecting from/to these unknown nodes.

Similar problems are encountered in analysis of human bodies. *Skill science* is a new discipline in which methods to achieve hard tasks and skills are explored.

Presented at the 19th International Conference on Inductive Logic Programming (ILP'09).

For example, cello playing requires a lot of skills but humans are often unaware of precise reasons why one performance is better than another. The problem of physical skill discovery is also represented in abduction, which provides a way to suggest players how to well perform hard tasks [7]. However, the current system in [7] cannot explain the principle behind an *empirical rule* such as "one can increase the sound volume if she keeps her arm shut during bowing" [1]. To formulate this explanation, we need to *abduce rules* which can explain *multiple goals* simultaneously and often involve *invention of new predicates* [11].

In this paper, we provide a simple and powerful method which infers missing nodes as well as missing arcs by abduction. Using SOLAR [12], we show how to realize an advanced form of rule abduction with predicate invention. SOLAR is a state-of-the-art inference system based on *SOL resolution* [2], and has been used to realize both abductive and inductive inference [3], yet we propose a different way of SOLAR utilization in this paper. An interesting feature in our abductive system is to infer rules. In other words, *induction is realized by abduction* (in part). This is contrasted with the fact that abduction is realized as a special case of induction in the framework of CF-induction [3]. The merit of our rule abduction lies in the fact that both enumeration of rule-form hypotheses and predicate invention are easily realized by SOLAR in a meta-level abduction. As an application of the proposed abductive system, we investigate how to discover *knack* in performing skillful tasks [1].

2 Hypothesis Generation by SOLAR

The logical framework of hypothesis generation in abduction can be expressed as follows. Let B be a set of clauses, which represents the *background knowledge*, and O be a set of literals, which represents *observations* (or *goals*). Also let Γ be a set of literals representing the set of *abducibles*, which are candidate assumptions to be added to B for explaining O. Given B, O and Γ, the hypothesis-generation problem is to find a set H of literals (called a *hypothesis*) such that

$$B \cup H \models O, \tag{1}$$

$$B \cup H \text{ is consistent, and} \tag{2}$$

$$H \text{ is a set of instances of literals from } \Gamma. \tag{3}$$

In this case, H is also called an *explanation* of O (with respect to B and Γ). An explanation H of O is *minimal* if no proper subset of H satisfies the above three conditions. We often introduce additional conditions of hypotheses such as the maximum number of literals in each explanation. A hypothesis is *ground* if it is a set of ground literals.

Given the observations O, each explanation H of O can be computed by the principle of *inverse entailment* [2], which converts the equation (1) to

$$B \cup \{\neg O\} \models \neg H, \tag{4}$$

where $\neg O = \bigvee_{L \in O} \neg L$ and $\neg H = \bigvee_{L \in H} \neg L$ are clauses because O and H are sets of literals. Similarly, the equation (2) is equivalent to $B \not\models \neg H$. Hence,

for any hypothesis H, its negated form $\neg H$ is deductively obtained as a "new" theorem of $B \cup \{\neg O\}$ which is not an "old" theorem of B alone. Moreover, by (3), every literal in $\neg H$ is an instance of a literal in $\overline{\Gamma} = \{\neg L \mid L \in \Gamma\}$.

SOLAR (SOL for Advanced Reasoning) [12] is a sophisticated deductive reasoning system based on SOL resolution [2], which is complete for finding subsumption-minimal consequences belonging to a given language bias (called a *production field*). Consequence-finding by SOLAR is performed by skipping literals belonging to a production field $\overline{\Gamma}$ instead of resolving them. Those skipped literals are then collected at the end of a proof, which constitute a clause as a logical consequence of the axiom set. SOLAR can thus be used to implement an abductive system that is *complete* for finding minimal explanations. A production field containing ground literals is converted to a non-ground production field by way of [14] to assure completeness of ground hypotheses in abduction.

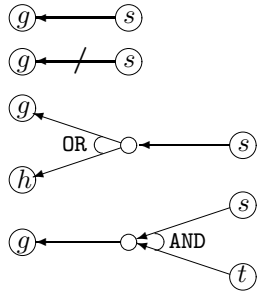
Unlike many other top-down resolution-based abductive procedures, which are designed for Horn clauses or normal logic programs, SOLAR is designed for *full clausal theories* containing non-Horn clauses. SOLAR avoids producing non-minimal/redundant consequences using various state-of-the-art pruning techniques [12], thereby enumeration of (negated) hypotheses is efficiently realized. This is a strong point of SOLAR because many abductive systems compute just single or a few hypotheses based on their own evaluation methods. Hypothesis enumeration by SOLAR, on the other hand, enables us to compare many different hypotheses and select the most probable ones from them in a statistical way based on an EM algorithm working on binary decision diagrams [5].

3 Rule Abduction

We now formalize *rule abduction*. Let B be our background knowledge. Suppose that we observe some phenomenon O as a result of some event I. We here assume that O is somehow *caused by* I, and our task of rule abduction is to explain why or how it is caused. Here, I and O are called an *input event* and a *goal event*, respectively. We represent causal relations by logical formulas. Using first-order predicate logic, we can represent algebraic properties of causal relations (such as transitivity and non-reflexivity) and constraints in a natural way. At the same time, abductive inference can be used to infer missing relations as well as missing events. This process corresponds to filling the gaps in causal graphs.

A *causal graph* is a directed graph representing causal relations, and consists of the sets of nodes and arcs. A *direct causal relation* corresponds to a directed arc, and a *causal chain* is represented by the reachability between two nodes. When there is a direct causal relation from the node s to the node g, we define that $connected(g, s)$ is true (5). Note that $connected(g, s)$ only shows that s is one of possible causes of g, and thus the existence of $connected(g, t)$ $(s \neq t)$ means that s and t are alternative causes for g. If we know that there is no direct causal link from s to g, we add an *integrity constraint* of the form (6), which is equivalent to the formula $\neg connected(g, s)$. If a direct causal relation from s has *nondeterministic effects* g and h, it is represented in the disjunction

of the form (7). On the other hand, if a direct causal relation to g has *conjunctive causes* s and t, it is represented in the disjunction of the form (8). Any other direct causal relation can be represented in a combination of these components.



$$connected(g, s) \tag{5}$$

$$\leftarrow connected(g, s) \tag{6}$$

$$connected(g, s) \vee connected(h, s) \tag{7}$$

$$connected(g, s) \vee connected(g, t) \tag{8}$$

The logic behind (8) is explained as follows. The relation that "g is caused by s and t" is intuitively written as $(g \leftarrow s \wedge t)$ in the object level, which is equivalent to $(g \leftarrow s) \vee (g \leftarrow t)$, hence the disjunction $connected(g, s) \vee connected(g, t)$.

When there is a *causal chain* from s to g, we defined that $caused(g, s)$ is true. Then, we have the following formulas as axioms:

$$caused(X, Y) \leftarrow connected(X, Y). \tag{9}$$

$$caused(X, Y) \leftarrow connected(X, Z) \wedge caused(Z, Y). \tag{10}$$

Here, the predicates *connected* and *caused* are both *meta-predicates* for object-level propositions g and s. That is, rules like causal relations in the object level are represented by atoms in the meta level. By this way, we can realize *rule abduction* in the meta level. We can also allow variables in those object-level expressions like $g(T)$ and $s(T)$, in which g and s are treated as function symbols in the meta level just as in the same way as Prolog can allow higher-order expressions, yet no recursive application of g and s is allowed.

When a causal graph is incomplete, there is no path between a goal event g and an input event s. Then, an abductive task infers missing links (and sometimes missing nodes) to complete a path between the two nodes. This is realized by setting the abducibles Γ as the atoms containing *connected* only: $\Gamma = \{connected(\_, \_)\}$. The observation is given in the form of the causal chain $caused(g, s)$, but we usually assume that there is no direct causal relation between them, i.e., $\leftarrow connected(g, s)$, otherwise we do not need abduction.

Suppose the *multiple* observations $\texttt{caused}(\texttt{g}, \texttt{s}) \wedge \texttt{caused}(\texttt{h}, \texttt{s})$. Then, a possible explanation is: $\exists \texttt{X}(\texttt{connected}(\texttt{g}, \texttt{X}) \wedge \texttt{connected}(\texttt{h}, \texttt{X}) \wedge \texttt{connected}(\texttt{X}, \texttt{s}))$. This $\texttt{X}$ can be unified with some known node in the causal graph, but if it is assumed as a new node, then this corresponds to introduction of a *new predicate*. Note here that, to introduce this kind of explanations, we need to allow *existentially quantified formulas* as hypotheses. Abduction by SOLAR enables us to infer this form of hypotheses. For examples of hypotheses containing multiple intermediate nodes, we have $\exists \texttt{X} \exists \texttt{Y}(\texttt{connected}(\texttt{g}, \texttt{X}) \wedge \texttt{connected}(\texttt{h}, \texttt{Y}) \wedge \texttt{connected}(\texttt{X}, \texttt{s}) \wedge \texttt{connected}(\texttt{Y}, \texttt{s}))$ and $\exists \texttt{X} \exists \texttt{Y}(\texttt{connected}(\texttt{g}, \texttt{X}) \wedge \texttt{connected}(\texttt{X}, \texttt{Y}) \wedge \texttt{connected}(\texttt{h}, \texttt{Y}) \wedge \texttt{connected}(\texttt{Y}, \texttt{s}))$. By this way of reasoning, we can enumerate different types of network structures which are missing in the original causal graph.

4 Application to Knack Discovery

The details of this section is written in the full version [4]. With the problem setting in [1], SOLAR computes 52 minimal hypotheses when the maximum search depth and the maximum length of produced clauses are set to 5 and 15, respectively. One promising hypothesis is: $\exists \mathtt{X}\,(\mathtt{connected}(\mathtt{stable\_bow\_movement}, \mathtt{X}) \wedge \mathtt{connected}(\mathtt{flexible\_wrist}, \mathtt{X}) \wedge \mathtt{connected}(\mathtt{X}, \mathtt{keep\_arm\_close}))$. Through an experiment, we have discovered a new finding consisting of three rules to increase the sound in cello playing by replacing `X` with `increase_upper_arm_impedance`.

5 Discussion

This paper proposes a new method to abduce rules. Abducible rules were firstly considered in Theorist [13], where each rule is given a *name* and those names are treated as abducible atoms. This is convenient when we know exact patterns of rules as strong biases, but it is not practical to prepare all patterns in order to abduce unknown rules. On the other hand, we have realized rule abduction in the meta level and avoid searching in the space of object-level rules.

Abductive reasoning in Robot Scientist [8, 9] has been implemented with a version of SOL resolution restricted to Horn clauses [15]. In our work, SOLAR, SOL for full-clausal theories, is necessary since our formalization allows non-Horn clauses in background theories. In [15], a *reaction* is given as a pair of sets of compounds representing the substrates and products, and a *metabolic graph* is defined in such a way that each node is given as a set of compounds available by sequences of reactions. In the logical form, a metabolic graph is defined by:

$$
\begin{aligned}
&edge(X, Y) \leftarrow reaction(A, B) \wedge (A \subseteq X) \wedge (X \cup B = Y),\\
&path(X, Y) \leftarrow edge(X, Y),\\
&path(X, Z) \leftarrow edge(X, Y) \wedge path(Y, Z).
\end{aligned}
$$

Here, the last two rules are exactly the same as our definition of *caused* by means of *connected*, but *edge* defined in the first rule is more complicated than *connected*. Then, abduction of reactions produces a combinatorially large number of compound pairs by bidirectional uses of operations `member` (for $\subseteq$) and `append` (for $\cup$) in the first rule. In this paper, we adopt classical graphs instead of metabolic graphs, and the use of a disjunction of *connected* literals has an effect to represent a direct multi-causal relationship by (8), which solves such combinatorial problems in representing network structures. Note that a hypothesis of the form (8) can also be computed using SOLAR either by taking a disjunction of explanations of the form $connected(g, \_)$ or by obtaining a *disjunctive answer* [14] for an observation containing free variables like $caused(g, X)$.

In [17], inference about metabolic pathways has been realized using CF-induction [3], in which both abduction and induction are used to generate hypotheses. Although CF-induction has been implemented by calling SOLAR, it requests the user to interact with the system to construct an appropriate hypothesis. On the other hand, SOLAR itself can easily enumerate abductive hypotheses without any user interaction. Moreover, predicate invention by CF-induction

should be realized in the framework of *inverse resolution* [11], which constructs parent clauses from their resolvents using the lgg operator and invents the literal resolved upon. This predicated invention can be used in a limited situation, while SOLAR realizes generation of new predicates in meta-level abduction. In the current implementation, predicate invention occurs in the ground level, and extension to allow function symbols and recursive expressions is a future work.

References

1. Furukawa, K., Kobayashi, I., Inoue, K. and Suwa, M.: Discovering knack by abductive reasoning, an English version is in preparation, 2009. A Japanese version has appeared as a report in JSAI SIG-SKL (Skill Science), January 2009: `http://www.jaist.ac.jp/ks/skl/papers/sig-skl-20090109-3.pdf`
2. Inoue, K.: Linear resolution for consequence finding, *Artificial Intelligence*, 56:301–353, 1992.
3. Inoue, K.: Induction as consequence finding, *Machine Learning*, 55:109–135, 2004.
4. Inoue, K., Furukawa, K. and Kobayashi, I.: Abducing rules with predicate invention, in: *Post-Proceedings of ILP 2009*, LNAI, Springer, to appear, 2009.
5. Inoue, K., Sato, T., Ishihata, M., Kameya, Y. and Nabeshima, H.: Evaluating abductive hypotheses using an EM algorithm on BDDs, in: *Proceedings of IJCAI-09*, to appear, 2009.
6. Kanehisa, M. and Goto, S.: KEGG: Kyoto encyclopedia of genes and genomes, *Nucleic Acids Research*, 28:27–30, 2000.
7. Kobayashi, I. and Furukawa, K.: Modeling physical skill discovery and diagnosis by abduction, *Information and Media Technologies*, 3(2):385–398, 2008.
8. King, R.D., *et al.*: Functional genomic hypothesis generation and experimentation by a robot scientist, *Nature*, 427:247–252, 2004.
9. King, R.D., *et al.*: The automation of science, *Science*, 324:85–89, 2009.
10. Muggleton, S. and Bryant, C.: Theory completion and inverse entailment, in: *Proceedings of ILP 2000*, LNAI 1866, pp. 130–146, Springer, 2000.
11. Muggleton, S. and Buntine, W.: Machine invention of first-order predicate by inverting resolution, in: *Proceedings of the 5th International Workshop on Machine Learning*, pp. 339–351, Morgan Kaufmann, 1988.
12. Nabeshima, H., Iwanuma, K. and Inoue, K.: SOLAR: a consequence finding system for advanced reasoning, in: *Proceedings of TABLEAUX '03*, LNAI, 2796, pp.257–263, Springer, 2003.
13. Poole, D.: A logical framework for default reasoning. *Artificial Intelligence*, 36:27–47, 1988.
14. Ray, O. and Inoue, K.: A consequence finding approach for full clausal abduction, in: *Proceedings of DS '07*, LNAI, 4755, pp.173–184, Springer, 2007.
15. Reiser, P.G.K., King, R.D., Kell, D.B., Muggleton, S.H., Bryant, C.H. and Oliver, S.G.: Developing a logical model of yeast metabolism, *Electronic Transactions in Artificial Intelligence*, 5-B2(024):223–244, 2001.
16. Tamaddoni-Nezhad, A., Chaleil, R., Kakas, A. and Muggleton, S.: Application of abductive ILP to learning metabolic network inhibition from temporal data, *Machine Learning*, 65:209–230, 2006.
17. Yamamoto, Y., Inoue, K. and Doncescu, A.: Integrating abduction and induction in biological inference using CF-Induction, in: Lodhi, H. and Muggleton, S. (eds.), *Elements of Computational Systems Biology*, to appear, John Wiley & Sons, 2009.