1 Abduction

Antonis C. Kakas
University of Cyprus, Cyprus

2 Synonyms

explanation; diagnosis;

3 Definition

Abduction is a form of reasoning, sometimes described as "deduction in reverse", whereby given a rule that "$A\ follows\ from\ B$" and the observed result of "A" we generate as a fact the condition "B" of the rule so that the observation follows deductively from the rule and the fact. We think of "B" as a possible explanation for the observation according to the given theory that contains our rule. This new information and its consequences (or ramifications) according to the given theory can be considered as the result of a (or part of a) learning process based on the given theory and driven by the observations that are explained by abduction. Abduction can be combined with induction in different ways to enhance this learning process.

4 Motivation and Background

Abduction is, along with induction, a *synthetic* form of reasoning whereby the syllogism generates, in its explanations, new information not hitherto contained in the current theory with which the reasoning is performed. As such, it has a natural relation to learning, and in particular to *knowledge intensive learning*, where the new information generated aims to complete, at least partially, the current knowledge (or model) of the problem domain as described in the given theory.

Early uses of abduction in the context of machine learning concentrated on how abduction can be used as a theory revision operator for identifying where the current theory could be revised in order to accommodate the new learning data. This includes the work of Michalski in [12], Ourston and Mooney in [17] and Abe et al in [4]. Another early link of abduction to learning was given by the *explanation based learning* method [2], where the abductive explanations of the learning data (training examples) are generalized to all cases.

Following this it was realized that the role of abduction in learning could be strengthened by linking it to induction, culminating to a hybrid integrated approach to learning where abduction and induction are tightly integrated to provide powerful learning frameworks such as the ones of Progol 5.0 [15] and HAIL [20]. On the other hand, from the point of view of abduction as "inference to the best explanation" [7] the link with induction provides a way to distinguish between different explanations and to select those explanations that give a better inductive generalization result.

The application of abduction, on its own or in combination with induction, to problems of Systems or Computational Biology trying to model biological processes and pathways at different levels, (see e.g. [10, 22, 25, 18, 19]) provides an important source of challenges for these methods of knowledge intensive learning.

5 Structure of the Learning

Abduction contributes to the learning task by first explaining, and thus rationalizing, the training data according to a given and current model of the domain to be learned. These abductive explanations either form on their own the result of learning or they feed into a subsequent phase to generate the final result of learning.

5.1 Abduction in Artificial Intelligence

Abduction as studied in the area of Artificial Intelligence and the perspective of learning is mainly defined in a logic-based approach[1] as follows.

Given a set of sentences T (a theory presentation), and a sentence O (observation), the abductive task is the problem of finding a set of sentences H (abductive explanation for O) such that:

(1) $T \cup H \models O$, and

(2) $T \cup H$ is consistent.

where $\models$ denotes the deductive entailment relation of the formal logic used in the representation of our theory and consistency refers also to the corresponding notion in this logic. The particular choice of this underlying formal framework of logic is in general a matter that depends on the problem or phenomena that we are trying to model. In many cases this is based on first order predicate calculus, as for example in the approach of Theory Completion in [16]. But other logics can be used, e.g. the non-monotonic logics of Default Logic or Logic Programming with Negation as Failure when the modelling of our problem requires this level of expressivity.

This basic formalisation as it stands, does not capture fully the explanatory nature of the abductive explanation H in the sense that it necessarily conveys some reason why the observations hold. It would for example allow an observation O to be explained by itself or in terms of some other observations rather than in terms of some "deeper" reason for which the observation must hold according to the theory T. Also as the above specification stands the observation can be abductively explained by generating in H some new (general) theory completely unrelated to the given theory T. In this case H does not account for the observations O according to the given theory T and in this sense may not be considered as an explanation for O relative to T. For these reasons, in order to specify a "level" at which the explanations are required and to understand these relative to the given general theory about the domain of interest, the members of

[1] Other approaches to abduction include the set covering approach see e.g [21] or case-based explanation see e.g [11].

an explanation are restricted to belong to a special pre-assigned, domain-specific class of sentences called *abducible*.

Hence abduction, is typically applied on a model, T, in which we can separate two disjoint sets of predicates: the *observable* predicates and the *abducible or open* predicates. The basic assumption then is that our model T has reached a sufficient level of comprehension of the domain such that all the incompleteness of the model can be isolated (under some working hypotheses) in its abducible predicates. The observable predicates are assumed to be completely defined (in T) in terms of the abducible predicates and other background auxiliary predicates; any incompleteness in their representation comes from the incompleteness in the abducible predicates. In practise, the empirical observations that drive the learning task are described using the observable predicates. Observations are represented by formulae that refer only to observable predicates (and possibly some background auxiliary predicates) typically by ground atomic facts on these observable predicates. The abducible predicates describe underlying (theoretical) relations in our model that are not observable directly but can, through the model T, bring about observable information.

The assumptions on the abducible predicates used for building up the explanations may be subject to restrictions that are expressed through *integrity constraints*. These represent additional knowledge that we have on our domain expressing general properties of the domain that remain valid no matter how the theory is to be extended in the process of abduction and associated learning. Therefore, in general an *abductive theory* is a triple, denoted by $< T, A, IC >$, where T is the background theory, A is a set of abducible predicates and IC, is a set of integrity constraints. Then, in the definition of an abductive explanation given above, one more requirement is added:

(3) $T \cup H$ satisfies IC.

where the satisfaction of integrity constraints can be understood in several ways (see [**?**] and references therein). Note that the integrity constraints reduce the number of explanations for a set of observations filtering out those explanations that do not satisfy them. Based on this notion of abductive explanation a *credulous* form of abductive entailment is defined. Given an abductive theory, $T = \langle T, A, IC \rangle$, and an observation O then, O is *abductively entailed* by T, denoted by $T \models_A O$, iff there exists an abductive explanation of O in T.

This notion of abductive entailment can then form the basis of a coverage relation for learning in the face of incomplete information.

5.2 Abductive Concept Learning

Abduction allows us to reason in the face of incomplete information. As such when we have learning problems where the background data on the training examples is incomplete the use of abduction can enhance the learning capabilities.

Abductive Concept Learning (ACL) [9] is a learning framework that allows us to learn from incomplete information and to later be able to classify new cases that again could be incompletely specified. Under ACL we learn abductive theories, $\langle T, A, IC \rangle$

with abduction playing a central role in the covering relation of the learning problem. The abductive theories learned in ACL contain both rules, in T, for the concept(s) to be learned as well as general clauses acting as integrity constraints in IC.

Practical problems that can be addressed with ACL: (i) concept learning from incomplete background data where some of the background predicates are incompletely specified and (ii) concept learning from incomplete background data together with given integrity constraints that provide some information on the incompleteness of the data. The treatment of incompleteness through abduction is integrated within the learning process. This allows the possibility of learning more compact theories that can alleviate the problem of overfitting due to the incompleteness in the data. A specific subcase of these two problems and important third application problem of ACL is that of (iii) multiple predicate learning, where each predicate is required to be learned from the incomplete data for the other predicates. Here the abductive reasoning can be used to suitably connect and integrate the learning of the different predicates. This can help to overcome some of the non-locality difficulties of multiple predicate learning, such as order-dependence and global consistency of the learned theory.

ACL is defined as an extension of Inductive Logic Programming where both the background knowledge and the learned theory being abductive theories. The central formal definition of ACL is given as follows where examples are atomic ground facts on the target predicate(s) to be learned.

Definition 1 (Abductive Concept Learning)
Given

- *a set of positive examples* E^+,
- *a set of negative examples* E^-,
- *an abductive theory* $T = \langle P, A, I \rangle$ *as background theory,*
- *an hypothesis space* $\mathcal{T} = \langle \mathcal{P}, \mathcal{I} \rangle$ *consisting of a space of possible programs* $\mathcal{P}$ *and a space of possible constraints* $\mathcal{I}$

Find

A set of rules $P' \in \mathcal{P}$ *and a set of constraints* $I' \in \mathcal{I}$ *such that the new abductive theory* $T' = \langle P \cup P', A, I \cup I' \rangle$ *satisfies the following conditions*

- $T' \models_A E^+$,
- $\forall e^- \in E^-, T' \not\models_A e^-$.

where E^+ *stands for the conjunction of all positive examples.*

An individual example e is said to be *covered* by a theory T' iff $T' \models_A e$. In effect, this definition replaces the deductive entailment as the example coverage relation in the ILP problem with abductive entailment to define the ACL learning problem.

The fact that the conjunction of positive examples must be covered means that, for every positive example, there must exist an abductive explanation and the explanations for all the positive examples must be consistent with each other. For negative examples, it is required that no abductive explanation exists for any of them. Abductive concept learning can be illustrated as follows.

Example 2 *Suppose we want to learn the concept $father$. Let the background theory be $T = \langle P, A, \emptyset \rangle$ where:*

$P = \{parent(john, mary), male(john),$
$parent(david, steve),$
$parent(kathy, ellen), female(kathy)\}$
$A = \{male, female\}$.

Let the training examples be:

$E^+ = \{father(john, mary), father(david, steve)\}$
$E^- = \{father(kathy, ellen), father(john, steve)\}$

In this case, a possible hypotheses $T' = \langle P \cup P', A, I' \rangle$ learned by ACL would consist of

$P' = \{father(X, Y) \leftarrow parent(X, Y), male(X).\}$
$I' = \{\leftarrow male(X), female(X).\}$

This hypothesis satisfies the definition of ACL because:

1. $T' \models_A father(john, mary), father(david, steve)$ *with* $\Delta = \{male(david)\}$*,*
2. $T' \not\models_A father(kathy, ellen)$*, as the only possible explanation for this goal, namely* $\{male(kathy)\}$ *is made inconsistent by the learned integrity constraint in* I'*.*
3. $T' \not\models_A father(john, steve)$*, as this has no possible abductive explanations.*

Hence, despite the fact that the background theory is incomplete (in its abducible predicates), ACL can find an appropriate solution to the learning problem by suitably extending the background theory with abducible assumptions. Note that the learned theory without the integrity constraint would not satisfy the definition of ACL, because there would exist an abductive explanation for the negative example $father(kathy, ellen)$*, namely* $\Delta^- = \{male(kathy)\}$*. This explanation is prohibited in the complete theory by the learned constraint together with the fact* $female(kathy)$*.*

The algorithm and learning system for ACL is based on a decomposition of this problem into two subproblems: (1) learning the rules in P' together with appropriate explanations for the training examples and (2) learning integrity constraints driven by the explanations generated in the first part. This decomposition allows ACL to be developed by combining the two Inductive Logic Programming settings of explanatory (predictive) learning and confirmatory (descriptive) learning. In fact, the first subproblem can be seen as a problem of learning from entailment, while the second subproblem as a problem of learning from interpretations.

5.3 Abduction and Induction

The utility of abduction in learning can be enhanced significantly when this is integrated with *induction*. Several approaches for synthesizing abduction and induction in learning have been developed, e.g. [1, 16, 24, 3]. These approaches aim to develop techniques for knowledge intensive learning with complex background theories. One problem to be faced by purely inductive techniques is that the training data on which the inductive process operates often contain gaps and inconsistencies. The general idea is that abductive reasoning can feed information into the inductive process by using the background theory for inserting new hypotheses and removing inconsistent data. Stated differently, abductive inference is used to complete the training data with hypotheses about missing or inconsistent data that explain the example or training data using the background theory. This process gives alternative possibilities for assimilating and generalizing this data.

Induction is a form of synthetic reasoning that typically generates knowledge in the form of new general rules that can provide, either directly, or indirectly through the current theory T that they extend, new interrelationships between the predicates of our theory that can include, unlike abduction, the observable predicates and even in some cases new predicates. The inductive hypothesis thus introduces new, hitherto unknown, links between the relations that we are studying thus allowing new predictions on the observable predicates that would not have been possible before from the original theory under any abductive explanation.

An inductive hypothesis, H, extends, like in abduction, the existing theory T to a new theory $T' = T \cup H$, but now H provides new links between observables and non-observables that was missing or incomplete in the original theory T. This is particularly evident by the fact that induction can be performed even with an empty given theory T, using just the set of observations. The observations specify incomplete (usually extensional) knowledge about the observable predicates, which we try to *generalise* into new knowledge. In contrast, the generalising effect of abduction, if at all present, is much more limited. With the given current theory T, that abduction always needs to refer to, we implicitly restrict the generalising power of abduction as we require that the basic model of our domain remains that of T. Induction has a stronger and genuinely new generalising effect on the observable predicates than abduction. While the purpose of abduction is to extend the theory with an explanation and then reason with it, thus enabling the generalising potential of the given theory T, in induction the purpose is to extend the given theory to a new theory, which can provide new possible observable consequences.

This complementarity of abduction and induction – abduction providing explanations from the theory while induction generalises to form new parts of the theory – suggests a basis for their integration within the context of theory formation and theory development. A *cycle of integration* of abduction and induction [3] emerges that is suitable for the task of incremental modelling (Figure 1). Abduction is used to transform (and in some sense normalize) the observations to information on the abducible predicates. Then induction takes this as input and tries to generalize this information to general rules for the abducible predicates now treating these as observable predicates for its own purposes. The cycle can then be repeated by adding the learned informa-

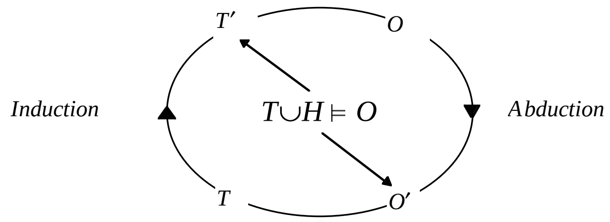

Figure 1: The cycle of abductive and inductive knowledge development. The cycle is governed by the 'equation' $T \cup H \models O$, where T is the current theory, O the observations triggering theory development, and H the new knowledge generated. On the left-hand side we have induction, its output feeding into the theory T for later use by abduction on the right; the abductive output in turn feeds into the observational data O' for later use by induction, and so on.

tion on the abducibles back in the model as new partial information on the incomplete abducible predicates. This will affect the abductive explanations of new observations to be used again in a subsequent phase of induction. Hence through this cycle of integration the abductive explanations of the observations are added to the theory, not in the (simple) form in which they have been generated, but in a generalized form given by a process of induction on these.

A simple example, adapted from [20], that illustrates this cycle of integration of abduction and induction is as follows. Suppose that our current model, T, contains the following rule and background facts:

sad(X) $\leftarrow$ *tired(X), poor(X).*
tired(oli), tired(ale), tired(kr),
academic(oli), academic(ale), academic(kr),
student(oli), lecturer(ale), lecturer(kr).

where the only observable predicate is $sad/1$.

Given the observations $O = \{sad(ale), sad(kr), not\ sad(oli)\}$ can we improve our model? The incompleteness of our model resides in the predicate $poor$. This is the only abducible predicate in our model. Using abduction we can explain the observations O via the explanation:

E = { *poor(ale), poor(kr), not poor(oli)* }.

Subsequently, treating this explanation as training data for inductive generalization we can generalize this to get the rule:

poor(X) $\leftarrow$ *lecturer(X)*

thus (partially) defining the abducible predicate $poor$ when we extend our theory with this rule.

This combination of abduction and induction has recently been studied and deployed in several ways within the context of Inductive Logic programming (ILP). In particular, *Inverse Entailement* [16] can be seen as a particular case of integration of abductive inference for constructing a “bottom” clause and inductive inference to generalize it. This is realized in Progol 5.0 and applied to several problems including the discovery of the function of genes in a network of metabolic pathways [10], and more recently to the study of inhibition in metabolic networks [22, 23]. In [14] an ILP system called ALECTO, integrates a phase of *extraction-case abduction* to transform each case of a training example to an abductive hypothesis with a phase of induction that generalizes these abductive hypotheses. It has been used to learn robot navigation control programs by completing the specific domain knowledge required, within a general theory of planning that the robot uses for its navigation [13].

The development of these initial frameworks that realize the cycle of integration of abduction and induction prompted the study of the problem of *completeness* for finding any hypotheses H that satisfies the basic formal of finding a consistent hypothesis H such that $T \cup H \models O$ for a given theory T, and observations O. Progol was found to be incomplete [24] and several new frameworks of integration of abduction and induction have been proposed. such as SOLDR [6], CF-Induction [5] and HAIL [20]. In particular, HAIL has demonstrated that one of the main reasons for the incompleteness of Progol is that in its cycle of integration of abduction and induction it uses a very restricted form of abduction. Lifting some of these restrictions, through the employment of methods from Abductive Logic Programming [8], has allowed HAIL to solve a wider class of problems. HAIL has also recently been used to learn Event Calculus theories for action description.

5.4 Abduction in Bioinformatics

Abduction has found a rich field of application in the domain of bioinformatics and the declarative modelling of computational biology. In a project called, Robot Scientist [10], Progol5.0 was used to generate abductive hypotheses about the function of genes. Similarly, learning the function of genes using abduction has been studied in GenePath [25] where experimental genetic data is explained in order to facilitate the analysis of genetic networks. Also in [18] abduction is used to learn gene interactions and genetic pathways from microarray experimental data. Abduction and its integration with induction has been used in the study of inhibitory effect of toxins in metabolic networks [22, 23] taking into account also the temporal variation that the inhibitory effect can have. Another bioinformatics application of abduction [19] concerns the modelling of Human Immunodeficiency Virus (HIV) drug resistance and using this in order to assist medical practitioners in the selection of anti-retroviral drugs for patients infected with HIV.

References

[1] Hilde Ade and Marc Denecker. AILP: Abductive inductive logic programming. In *IJCAI*, pages 1201–1209, 1995.

[2] G. DeJong and R.J. Mooney. Explanation-based learning: An alternate view. *Machine Learning*, 1:145–176, 1986.

[3] P. Flach and A.C. Kakas. Abductive and inductive reasoning: Background and issues. In P. A. Flach and A. C. Kakas, editors, *Abductive and Inductive Reasoning*, Pure and Applied Logic. Kluwer, 2000.

[4] B. Malfait H. Ade and L. De Raedt. Ruth: an ilp theory revision system. In *ISMIS94*. Springer-Verlag, 1994.

[5] K. Inoue. Inverse entailment for full clausal theories. In *LICS-2001 Workshop on Logic and Learning*, 2001.

[6] K. Ito and A. Yamamoto. Finding hypotheses from examples by computing the least generlisation of bottom clauses. In *Proceedings of Discovery Science '98*, pages 303–314. Springer, 1998.

[7] J.R. Josephson and S.G. Josephson, editors. *Abductive Inference: Computation, Philosophy, Technology*. Cambridge University Press, 1994.

[8] A.C. Kakas, R.A. Kowalski, and F. Toni. Abductive Logic Programming. *Journal of Logic and Computation*, 2(6):719–770, 1992.

[9] A.C. Kakas and F. Riguzzi. Abductive concept learning. *New Generation Computing*, 18:243–294, 2000.

[10] R.D. King, K.E. Whelan, F.M. Jones, P.K.G. Reiser, C.H. Bryant, S.H. Muggleton, D.B. Kell, and S.G. Oliver. Functional genomic hypothesis generation and experimentation by a robot scientist. *Nature*, 427:247–252, 2004.

[11] D.B. Leake. Abduction, experience and goals: A model for everyday abductive explanation. *The Journal of Experimental and Theoretical Artificial Intelligence*, 7:407–428, 1995.

[12] R. S. Michalski. Inferential theory of learning as a conceptual basis for multistrategy learning. *Machine Learning*, 11:111–151, 1993.

[13] S. Moyle. Using theory completion to learn a robot navigation control program. In *Proceedings of the 12th International Conference on Inductive Logic Programming*, pages 182–197. Springer-Verlag, 2002.

[14] S. A. Moyle. *An investigation into Theory Completion techniques in Inductive Logic Programming*. PhD thesis, Oxford University Computing Laboratory, University of Oxford, 2000.

[15] S. Muggleton. Inverse entailment and Progol. *New Generation Computing*, 13:245–286, 1995.

[16] S.H. Muggleton and C.H. Bryant. Theory completion using inverse entailment. In *Proc. of the 10th International Workshop on Inductive Logic Programming (ILP-00)*, pages 130–146, Berlin, 2000. Springer-Verlag.

[17] D. Ourston and R. J. Mooney. Theory refinement combining analytical and empirical methods. *Artificial Intelligence*, 66:311–344, 1994.

[18] I. Papatheodorou, A. Kakas, and M. Sergot. Inference of gene relations from microarray data by abduction. In *Proceedings of the Eighth International Conference on Logic Programming and Non-Monotonic Reasoning (LPNMR'05)*, volume 3662, pages 389–393. Springer, 2005.

[19] O. Ray, A. Antoniades, A. Kakas, and I. Demetriades. Abductive Logic Programming in the Clinical Management of HIV/AIDS. In G. Brewka, S. Coradeschi, A. Perini, and P. Traverso, editors, *Proceedings of the 17th European Conference on Artificial Intelligence*.

[20] O. Ray, K. Broda, and A. Russo. Hybrid Abductive Inductive Learning: a Generalisation of Progol. In *13th International Conference on Inductive Logic Programming*, volume 2835 of *LNAI*, pages 311–328. Springer Verlag, 2003.

[21] J. Reggia. Diagnostic experts systems based on a set-covering model. *International Journal of Man-Machine Studies*, 19(5):437–460, 1983.

[22] A. Tamaddoni-Nezhad, A. Kakas, S.H. Muggleton, and F. Pazos. Modelling inhibition in metabolic pathways through abduction and induction. In *Proceedings of the 14th International Conference on Inductive Logic Programming*, pages 305–322. Springer-Verlag, 2004.

[23] Alireza Tamaddoni-Nezhad, Raphael Chaleil, Antonis Kakas, and Stephen Muggleton. Application of abductive ilp to learning metabolic network inhibition from temporal data. *Mach. Learn.*, 64(1-3):209–230, 2006.

[24] A. Yamamoto. Which hypotheses can be found with inverse entailment? In *Proceedings of the Seventh International Workshop on Inductive Logic Programming*, pages 296–308. Berlin, 1997. LNAI 1297.

[25] B. Zupan, I. Bratko, J. Demsar, P. Juvan, J.A Halter, A. Kuspa, and G. Shaulsky. Genepath: a system for automated construction of genetic networks from mutant data. *Bioinformatics*, 19(3):383–389, 2003.