A Model for Generalization Based on Confirmatory Induction

Nicolas Lachiche¹ and Pierre Marquis²

¹ CRIN/CNRS - INRIA Lorraine, Bâtiment LORIA, Campus scientifique B.P. 239,
54506 Vandœuvre-lès-Nancy Cedex, France - e-mail: lachiche@loria.fr

² CRIL/Université d'Artois, Rue de l'Université, S.P. 16,
62307 Lens Cedex, France - e-mail: marquis@lens.lifl.fr

Abstract. Confirmatory induction is based on the assumption that unknown individuals are similar to known ones, i.e. they satisfy the properties shared by known individuals. This assumption can be represented inside a non-monotonic logical framework. Accordingly, existing approaches to confirmatory induction take advantage of the machinery developed so far for non-monotonic inference. However, they are based on completion policies that are unnecessary strong for the induction purpose. The contribution of this paper is twofold: some basic requirements that any model for generalization based on confirmatory induction should satisfy are proposed. Then, a model for generalization based on Hempel's notion of confirmation is introduced. This model is rational in the sense that it satisfies the rationality postulates we exhibit; moreover, the completion principle on which this model is based captures exactly the similarity assumption, hence the model can be considered minimal as well.

1 Introduction

Inductive generalization is inference of general laws from examples. It is a central concern of Machine Learning and the Philosophy of Sciences; accordingly, many researches have been devoted to it for several decades. Despite a huge amount of work on this topic, both from philosophers, logicians and AI researchers, *there is no consensus on what induction precisely is*. Indeed, different kinds of induction can easily be envisioned, and they cannot be reduced to some common ground. This diversity has been recently acknowledged by the AI community. Particularly, Flach [5] clearly points out two forms of induction, so-called explanatory induction and confirmatory induction.

Given some background knowledge Th , *explanatory induction* aims at generating generalizations G which deductively explain the observation report E , in the sense that in every world where Th and G are true, E is also true. Formally, we have: $G \wedge Th \models E$.

Contrastingly, *confirmatory induction* is based on the evidence that an inductive hypothesis does not necessarily need to explain some observations, but to be confirmed by them in some sense. For instance, if all what I know is that Tweety is a bird, Tweety flies, Superman flies, Garfield is a cat and every cat is

not a bird, then provided that every unknown individual behaves like the known ones (i.e. every individual possesses every property shared by both Tweety, Superman and Garfield), Woodstock can be assumed flying whenever it is known to be a bird. Similar conclusions can be drawn with every bird. More formally,

$$G =_{\text{def}} \forall X (\text{bird}(X) \Rightarrow \text{flies}(X))$$

can be considered as a generalization of

$$E =_{\text{def}} \{\text{bird}(\text{Tweety}), \text{flies}(\text{Tweety}), \text{flies}(\text{Superman}), \\ \text{cat}(\text{Garfield}), \forall X (\text{cat}(X) \Rightarrow \neg \text{bird}(X))\}.$$

Though G is not a logical consequence of evidence E , it is satisfied (we also say “confirmed”) by every known individual.

Within confirmatory induction, there is no need to keep separate some background knowledge with the observation report; every piece of knowledge is viewed as evidence and incorporated into E . Clearly enough, E is not a logical consequence of G , hence G cannot be viewed as an explanation of E (at least, with a deductive meaning): assuming that every bird flies does not explain at all why Tweety is a bird.

While explanatory induction is commonly viewed as reversed deduction, confirmatory induction can be considered as deduction from a completed theory [14,9]. The implicit completion of E which is performed must capture the assumption that *every unknown individual is similar to the known ones, i.e. it exhibits the same properties*. This “induction principle” is often referred to as *the similarity assumption*.

Viewing induction as deduction w.r.t. a completed theory allows one to put forward interesting connections with several non-monotonic inference formalisms, including the closed-world assumption [17] and its generalizations [13], Clark’s completion [2] [19] and some forms of circumscription [11] [12]. Indeed, non-monotonic inference can often be considered as deduction from a completed theory. However, inductive inference is non-monotonic inference of a special kind, and it does not come down to any standard form of non-monotonic inference. Such a confusion exists in the models for confirmatory induction by [6,10,15,16,14]. All these approaches take advantage of the machinery developed for non-monotonic inference but they do not make a precise separation between the knowledge level of induction (i.e. what is a generalization?) and the inference level of induction (i.e. how to compute generalizations?). In particular, the notions of confirmation on which such approaches are based are not clearly pointed out. In our opinion, a clear separation between confirmation and generalization is necessary for a good understanding of confirmatory generalization. Moreover, the rationality issue is not addressed. Thus, some features of existing approaches are motivated by the intended notion of generalization (at the knowledge level) whereas some other ones, e.g. restrictions of the representation language, are required by the machinery used to derive generalizations (at the inference level).

The contribution of this paper is twofold. First, some basic requirements that any model for generalization based on confirmatory induction should satisfy are pointed out. Second, a model for generalization based on Hempel’s proposal for confirmation is presented. Both the language of evidence and the language of

generalizations used in this model are more expressive than the languages used in existing approaches. Consequently, this model allows one to derive generalizations that cannot be built up using previous models for generalization; it also prevents from generating unexpected generalizations.

2 Rationality Postulates for Generalization

The purpose of this section is to point out some requirements that every model for generalization should satisfy. Particularly, we focus on generalization at the knowledge level.

Let us define a *model for generalization based on confirmatory induction* as a 5-uple $\langle L_E, L_H, L_G, \vdash, \approx \rangle$. L_E (resp. L_H, L_G) is a representation language of observation reports including eventually some background knowledge (resp. of confirmed statements, of generalizations). $\vdash$ is a confirmation relation between L_E and L_H , and $\approx$ a generalization relation between L_E and L_G .

In such models, the notion of confirmed statement and the notion of generalization are kept separate. While confirmatory induction aims at deriving statements supported by some evidence, inductive generalization aims at pointing out general laws supported by some evidence. Thus confirmatory generalization must satisfy:

- *Confirmatory foundation*: if $E \approx G$, then $E \vdash G$.
- *Ampliative reasoning*: if $E \models G$, then $E \not\approx G$.
- *Generality condition*: if $E \approx G_1$, $E \approx G_2$ and $G_1 \models G_2$, then $G_1 \equiv G_2$.

The first postulate characterizes generalizations based on confirmatory induction. The ampliative reasoning postulate and the generality condition postulate allow one to discriminate generalizations among all confirmed statements: generalizations must convey new information, and as much information as possible.

Since inferring generalizations from evidence consists in selecting some statements among those confirmed by evidence, generalization benefits of some rationality postulates for confirmation given by Hempel [7,8] and Flach [5], namely:

- *Verification principle*: if $E \approx H$ and $E \wedge H \models F$, then $E \wedge F \approx H$.
- *Falsification principle*: if $E \approx H$ and $E \wedge H \models F$, then $E \wedge \neg F \not\approx H$.
- *Consistency condition*: if E is consistent, then E is consistent with the whole set of the hypotheses it confirms.
- *Right equivalence*: if H_1 and H_2 are equivalent and $E \approx H_1$, then $E \approx H_2$.
- *Left equivalence*: if E_1 and E_2 are equivalent and $E_1 \approx H$, then $E_2 \approx H$.
- *Cautious monotonicity*: if $E \approx H$ and $E \approx F$, then $E \wedge F \approx H$.

The other rationality postulates for confirmation proposed by Hempel and Flach cannot be kept for generalization. Ampliative reasoning requires the entailment condition and reflexivity to be given up. Generality condition is not compatible with right weakening, cumulative right weakening and right AND introduction. Left OR introduction must also be rejected because it is not compatible with the similarity assumption itself; for instance, $\forall X, flies(X)$ is confirmed

by $flies(Superman)$ and by $flies(Tweety)$, but we are not ready to consider it as confirmed by $flies(Superman) \vee flies(Tweety)$ (just because $\forall X, flies(x)$ could not be derived from $flies(Superman) \vee flies(Tweety)$ if the universe were reduced to $\{Superman, Tweety\}$).

All the requirements above do not define a unique model for generalization. They aim only at characterizing some properties that a rational model for generalization should satisfy. We call *minimal models for generalization* those relying on completion policies allowing one to capture the similarity assumption and nothing else. The completeness issue of these postulates is not addressed in this paper; while we consider that every reasonable model for generalization should satisfy these postulates, we do not claim that the converse is also the case.

3 Existing Approaches to Generalization are not Minimal ones

In this section, we show that existing approaches to generalization based on confirmation [6,10,15,16,14] use completion principles unnecessary strong for the induction purpose. Indeed, the similarity assumption cannot be accurately represented by the completion policies used in these approaches, even if it is embodied in them. Intuitively, the similarity assumption requires the universe to be circumscribed to known *individuals*, only, while completion principles used in non-monotonic inferences aim mainly at circumscribing *properties* of such individuals. As a consequence, these approaches can easily produce unexpected generalizations and miss interesting ones.

Let us recall that Helft's model for generalization is based on subimplication [1], Marquis', Muggleton and De Raedt's, De Raedt and Dzeroski's models on inference w.r.t. minimal Herbrand model and De Raedt and Bruynooghe's on Clark's completion. These three forms of non-monotonic inference circumscribe some properties that individuals satisfy; particularly, they give to positive information a special status w.r.t. negative information: every positive ground fact which cannot be deduced from the evidence database is assumed false. Because such a completion is not always desired, unexpected generalizations can be generated. For instance, $E =_{def} \{orange(Garfield), bird(Tweety)\}$ does not confirm $G =_{def} \forall X, \neg orange(X) \vee \neg bird(X)$ unless E is explicitly completed with $\{\neg orange(Tweety), \neg bird(Garfield)\}$, thanks to the closed-world assumption. Clearly enough, such a completion formula has nothing to do with induction.

A second problem with existing approaches is that the completion of a consistent database may easily result in an inconsistent database, whatever minimal models [6,10,16,14] or Clark's completion [15] are used. In order to deal with this problem, the representation languages of evidence and generalizations are restricted in the approaches mentioned above. Once again, such restrictions are not required by the similarity assumption. They may lead one to miss some interesting generalizations. For instance, none of the approaches above allows one to consider $\forall X, orange(X) \vee bird(X)$ as a generalization of $orange(Garfield) \vee bird(Tweety)$.

4 A Model for Generalization based on Hempel's Notion of Confirmation

In this section, a model for generalization $\langle L_E, L_H, L_G, \vdash, \approx \rangle$ is pointed out. It satisfies the requirements stated in Section 2 and is minimal as well. Interestingly, it does not suffer from the impediments described above.

4.1 Confirmation

The notion of confirmation on which our model is based is close to those proposed by Hempel [7]. Intuitively, E confirms H if and only if E would entail H if the universe were reduced to the set of individuals appearing in E . In other words, E confirms H whenever H is a logical consequence of E under the similarity assumption. Because no additional assumptions are taken into account, the models for generalization based on such a notion of confirmation can be considered as minimal ones.

Definitions. In order to figure out what the universe would be when restricted to known individuals, Hempel introduced the concept of *development of a formula for a finite class of individual constants*:

Definition 1. The development of a first-order formula H for a finite class of individual constants C , noted $D_C(H)$, is inductively defined as follows:

$$\begin{array}{l|l} D_C(\neg H) & =_{def} \neg D_C(H) \\ D_C(H \wedge G) & =_{def} D_C(H) \wedge D_C(G) \\ D_C(\forall x H) & =_{def} \bigwedge_{c \in C} D_C(H[x \leftarrow c]) \\ D_C(H) & =_{def} H, \text{ if } H \text{ is ground} \end{array} \quad \left| \quad \begin{array}{l} D_C(H \Rightarrow G) =_{def} D_C(H) \Rightarrow D_C(G) \\ D_C(H \vee G) =_{def} D_C(H) \vee D_C(G) \\ D_C(\exists x H) =_{def} \bigvee_{c \in C} D_C(H[x \leftarrow c]) \end{array} \right.$$

where $H[x \leftarrow c]$ is the formula which results from replacing in H every free occurrence of x by c .

In our approach, we consider the *Herbrand domain* of E , i.e. the set C_E of all constant symbols occurring in E as the set of known individuals. For instance, the development of the formula $H =_{def} \forall X (bird(X) \Rightarrow flies(X))$ for the Herbrand domain of

$$E =_{def} \{bird(Tweety), flies(Tweety), flies(Superman), cat(Garfield), \forall X (cat(X) \Rightarrow \neg bird(X))\}$$

is the ground formula

$$\begin{aligned} & (bird(Tweety) \Rightarrow flies(Tweety)) \\ & \wedge (bird(Superman) \Rightarrow flies(Superman)) \\ & \wedge (bird(Garfield) \Rightarrow flies(Garfield)). \end{aligned}$$

This ground formula represents the same information than H provided that the universe is reduced to $\{Tweety, Superman, Garfield\}$.

Definition 2. — L_E is the language of universal formulas of function-free first-order logic without equality such that at least one constant symbol occurs in the representation of any observation report, i.e. the Herbrand domain of any E is finite and non-empty.

– We also impose that $L_H = L_E$.

Definition 3. Let $E \in L_E$ and $H \in L_H$ s.t. $C_H \subseteq C_E$. $E \vdash H$ if and only if $E \models_{D_{C_E}}(H)$.

In our approach, hypothesis H is confirmed by evidence E whenever H can be deduced from E under the similarity assumption. Accordingly, this notion of confirmation can be considered as a refinement of Flach’s consistency-based confirmation [3,4], where E and H are only required to be consistent altogether.

The various restrictions put on L_E and L_H are motivated as follows. First, if empty Herbrand domains were allowed (e.g. $\forall X, flies(X)$ is considered as an admissible evidence), then confirmed statements which are inconsistent with the corresponding evidence could be easily derived (e.g. $\forall X, \neg flies(X)$ is confirmed by $\forall X, flies(X)$ since the development of $\forall X, \neg flies(X)$ for the empty Herbrand domain is true). This would violate the consistency condition. The absence of equality in the language allows one to reject evidence like $E =_{def} \forall X, (X \neq Superman)$. Indeed, since the development of $H =_{def} \forall X, (X = Superman)$ for $\{Superman\}$ is $Superman = Superman$, which must be considered true in every first-order language with equality, H would be confirmed by E . Once again, the consistency condition would be violated. A similar problem would occur with $E =_{def} \exists X, \neg flies(X) \wedge flies(Superman)$ and $H =_{def} \forall X, flies(X)$ if existential formulas were accepted in the language of evidence. This is why this is not the case. Finally, the Herbrand domain of confirmed statements H is required to be included in the Herbrand domain of the corresponding evidence E , so as to avoid the happening of “miraculously generated” individuals in confirmed statements.

Hempel’s Confirmation. Though the confirmation relation above is close to Hempel’s proposal [8], it differs from it in several ways. First, our notion of confirmation corresponds to Hempel’s *direct confirmation*. We do not consider confirmation defined by extending direct confirmation with logical consequence, as Hempel did to satisfy the entailment condition, because this postulate is not compatible with the ampliative reasoning postulate (cf. Section 2).

Additionally, we do not impose H to be valid whenever $D_{C_E}(H)$ is valid; for instance, given $E =_{def} flies(Tweety)$ and $H =_{def} \forall X, flies(X) \vee \forall X, \neg flies(X)$, we consider that the fact that $D_{C_E}(H) = flies(Tweety) \vee \neg flies(Tweety)$ is valid, while H is not, does not prevent E from confirming H .

Finally, assimilating known individuals with the Herbrand domain of evidence imposes to consider $E_1 =_{def} flies(Superman) \wedge (flies(Superman) \vee flies(Tweety))$ and $E_2 =_{def} flies(Superman)$ as two different evidences. Indeed, while these two formulas are logically equivalent, they do not convey the same information w.r.t. individuals of the domain: from the first one, we want to conclude that two individuals belong to the domain. Such a distinction has a significant influence on the confirmatory relation: in our model, $H = \forall X, flies(X)$ is considered as confirmed by E_2 but not by E_1 ; as a consequence, left logical equivalence is satisfied by $\vdash$ if we consider that two formulas are equivalent if

and only if they are logically equivalent and share the same Herbrand domain, i.e. they are logically equivalent under the domain-closure axiom and the unique name axiom. Hempel addressed this problem differently by considering only the *essential constants* of E , i.e. constants of E that must occur in every formula logically equivalent to E [8].

4.2 Generalization

Definition 4. As a representation language L_G for generalizations, we consider the language of confirmed statements L_H , restricted to universal clauses.

This last restriction could be easily given up but we take it because clausal theories are easy to understand. Moreover, it is helpful at the inference level. Excluding the equality symbol from the language of generalizations is necessary to avoid considering as a generalization the domain-closure axiom which is always confirmed since it is implicit in the logic.

Definition 5. Let $E \in L_E$ and $G \in L_G$. $E \approx G$ if and only if

- $E \vdash G$, and
- $E \not\models G$, and
- If G' is a clause such that $E \vdash G'$, $E \not\models G'$ and $G' \models G$, then $G \models G'$.

The rationality postulates for confirmatory generalization given in Section 2 are satisfied by $\approx$. Obviously, $\approx$ satisfies the first three postulates (confirmatory foundation, ampliative reasoning and generality condition). Verification and falsification principles and cautious monotonicity also hold. The consistency condition is satisfied as well (the proof is similar to the one proposed by Hempel in [7]). Right equivalence is a consequence of properties of the development of a universal formula for a finite class of individual constants also proved by Hempel [7]. Finally, left equivalence holds provided that logical equivalence over the same Herbrand domain is considered.

Generalizations G are derived thanks to a generate-and-test search strategy. Especially, a candidate formula for generalization is first generated, then checked for generalization. Such a generate-and-test approach is also used in all existing techniques. So as to promote the production of most general clauses first, the language of generalizations is searched w.r.t. subsumption ordering. The search can be realised through iterative deepening or a variant of SE-tree search [18] for a first-order logic language without functional symbols. The search can also be constrained by focusing on clauses built up exclusively from a user-defined set of predicates, or by limiting the number of literals in generalizations.

Given the database

$\{bird(Tweety), cartoon(Tweety), flies(Superman), cartoon(Superman)\}$,

Claudien, the system described in [15], produces the set of generalizations:

$\{\forall X, \neg bird(X) \vee \neg flies(X); \forall X, \neg bird(X) \vee cartoon(X);$

$\forall X, \neg flies(X) \vee cartoon(X); \forall X, \neg cartoon(X) \vee bird(X) \vee flies(X)\}$.

Contrastingly, our model gives the set: $\{\forall X, cartoon(X); \forall X, bird(X) \vee flies(X)\}$.

Clearly enough, the clauses generated thanks to our model are more general ones; moreover, the unexpected generalization $\forall X, \neg bird(X) \vee \neg flies(X)$ is avoided in our approach.

References

1. Geneviève Bossu and Pierre Siegel. Saturation, nonmonotonic reasoning and the closed-world assumption. *Artificial Intelligence*, 25:13–63, 1985.
2. Keith L. Clark. Negation as failure. In Hervé Gallaire and Jack Minker, editors, *Proceedings of the Symposium on Logic and Databases*, pages 293–322, New York, 1978. Plenum Press.
3. Peter Flach. *Inductive Logic Programming*, chapter A framework for Inductive Logic Programming, pages 193–210. S. Muggleton, academic press edition, 1992.
4. Peter Flach. *Conjectures: an inquiry concerning the logic of induction*. PhD thesis, Tilburg University, 1995.
5. Peter Flach. Rationality postulates for induction. In *Proc. TARK'1996*, pages 267–281, De Zeeuwse Stromen, The Netherlands, 1996.
6. Nicolas Helft. Induction as nonmonotonic inference. In *Proc. KR'89*, pages 149–156, Toronto, Canada, 1989. Morgan Kaufmann.
7. C. G. Hempel. A purely syntactical definition of confirmation. *Journal of Symbolic Logic*, 8(4):122–143, 1943.
8. C. G. Hempel. Studies in the logic of confirmation. *Mind*, 54(213 & 214):1–26 & 97–121, 1945.
9. Nicolas Lachiche and Pierre Marquis. Abduction, induction and completion policies. In Peter Flach, editor, *ECAI'96 Workshop on Abductive and Inductive Reasoning*, pages 43–46, Budapest, Hungary, 1996.
10. Pierre Marquis. Building up inductive generalizations from facts. In *Proc. ECAI'92*, pages 446–450, Vienna, Austria, 1992. Wiley & Sons.
11. John McCarthy. Circumscription: a form of non-monotonic reasoning. *Artificial Intelligence*, 13:27–39, 1980.
12. John McCarthy. Applications of circumscription to formalizing common-sense knowledge. *Artificial Intelligence*, 28:89–116, 1986.
13. Jack Minker. On indefinite databases and the closed world assumption. In *Proc. CADE'82*, pages 292–308, Berlin, Germany, 1982. Lecture Notes in Computer Science 138, Springer-Verlag.
14. Stephen Muggleton and Luc De Raedt. Inductive logic programming: Theory and methods. *Journal of Logic Programming*, 19:629–679, 1994.
15. Luc De Raedt and Maurice Bruynooghe. A theory of clausal discovery. In *Proc. IJCAI'93*, pages 1058–1063, Chambéry, France, 1993.
16. Luc De Raedt and Saso Dzeroski. First-order jk -clausal theories are pac -learnable. *Artificial Intelligence*, 70:375–392, 1994.
17. Raymond Reiter. On closed-world data base. In Hervé Gallaire and Jack Minker, editors, *Proceedings of the symposium on logic and data bases*, pages 55–76, New York, 1978. Plenum Press.
18. Ron Rymon. Search through systematic set enumeration. In *Proc. KR'92*, pages 539–550, Cambridge, Massachusetts, USA, 1992. Morgan Kaufmann.
19. John C. Shepherdson. Negation in logic programming. In Jack Minker, editor, *Foundations of deductive databases and logic programming*, pages 19–89, 1988.