A Formal Approach to Learning from Examples

James P. Delgrande

School of Computing Science.
Simon Fraser University.
Burnaby. B.C.,
Canada V5A 1S6

Abstract

This paper presents a formal, foundational approach to learning from examples in machine learning. It is assumed that a learning system is presented with a stream of facts describing a domain of application. The task of the system is to form and modify hypotheses characterising the relations in the domain, based on this information. Presumably the set of hypotheses that may be so formed will require continual revision as further information is received.

The emphasis in this paper is to characterise those hypotheses that may potentially be formed, rather than to specify the subset of the hypotheses that, for whatever reason, should be held. To this end. formal systems are derived from which the set of potential hypotheses that may be formed is precisely specified. A procedure is also derived for restoring the consistency of a set of hypotheses after conflicting evidence is encountered. In addition, this work is extended to where a learning system may be "told" arbitrary sentences concerning a domain The approach is intended to provide a basic framework lor the development of systems that learn from examples, as well as a neutral point from which such systems may be viewed and compared.

1. Introduction

Learning from examples is an important yet basic subarea of machine learning. For this approach, a learning system receives information concerning a domain of application in the form of facts, or ground atomic formulae. On the basis of this information the learning system induces general statements characterising the domain, and hence hypothesises relations among the known relations in the domain. These hypotheses are phrased independently of any particular individuals. Further facts may enable other hypotheses to be formed while falsifying existing hypotheses. Thus the consistency of the set of hypothesised statements must continually be maintained as new information is discovered, and the question arises as to how a set of hypotheses may be modified as falsifying instances are encountered.

The early work of Patrick Winston [Winston 75] provides a good example of such an approach. In Winston's system, descriptions of concepts are formed from a set of carefully chosen examples of the concept and "near misses". A near miss is an example that is quite similar to an instance of the concept. but differs in a small number of significant details. Relevant features that the concept (presumably) must have are extracted from the positive examples, while negative information is extracted from the near misses. Thus from positive examples the program might infer that an arch must have two supports, while from a negative example it might infer that the supports must not be touching. As examples are received, the definition of a concept passes through successive refinements, presumably converging to some acceptable definition.

In this paper a formal, foundational approach to learning from examples is presented. The overall aim is to investigate the underlying formal aspects of such learning. In contrast to previous work, pragmatic concerns dealing with notions of evidence. confirmation and justification of hypotheses are ignored insofar as is possible. The goal of this paper then is to characterise the hypotheses that may *potentially* be formed. The question of what hypotheses may *Justifiably* be formed is not addressed. Thus for example if we have a set of black ravens and know of no non-black ravens, we could hypothesise that ravens are black. However the approach at hand gives no indication as to when such a hypothesis should be formed or what constitutes adequate evidence for such an assertion. So the goal is to determine formal criteria which prescribe the set of potential conjectures, rather than to determine pragmatic criteria whereby an acceptable set of conjectures may be formed. A similar distinction can be made in a deductive system, where an underlying logic specifies what *could* be derived, but not what *should* be derived.

The remainder of this section expands on these ideas and surveys related work. In the second section, a language for expressing conjectures is introduced, and formal systems are developed for guiding the formation of conjectures. These systems lead immediately to a procedure for restoring the consistency of a set of conjectures Extensions to the approach are briefly described in the third and fourth sections, and in the fifth section the approach is compared with representative AI systems for learning from examples. Further details, proofs of theorems, etc. may be found in [Delgrande 85].

1.1. The Approach

The domain of application is assumed to be describable as a collection of individuals and relations on these individuals. Further, it is assumed that some portion of the domain, described by a finite set of ground atomic formulae (or. informally, "facts"), is known by the learning system. As time progresses the learning system' will presumably encounter new information, and so the set of known ground atomic formulae will monotonically increase. Initially I assume that this (increasing) set of ground formulae is *all* that is known about the domain.

Hypotheses are proposed and modified on the basis of this finite, monotonically increasing set of ground instances. The hypotheses are expressed in a language. HL. that is a simple variant of the language of elementary algebra. The criteria for proposing a hypothesis are straightforward: there is a reason to do so (i.e. some notion of evidence is satisfied) and the hypothesis is not known to be false. These criteria though are far too simplistic, and in general the resultant set of hypotheses will be inconsistent.

These difficulties are circumvented in the following manner. With each term in a sentence of HL we can associate two subsets of the ground instances, consisting of those known to satisfy the term and those known to not. Thus to the term "black raven" we can associate the set of individuals known to be black and a raven, and the set known to be either non-black or non-raven. These (pairs of) sets interrelate in various ways: formal systems are developed to precisely characterise relations between terms in HL by means of these sets. From these systems, ground instances whose truth values are unknown can be iteratively located so that determining their truth value leads to a convergence of the hypothesis set to consistency. These "knowable but unknown ground instances" arc composed of constants and predicates symbols that occur in the set of known ground atomic formulae. Informally they correspond to unknown but potentially knowable "facts" in the domain. The capability of testing individuals for membership in a relation, where both individuals and relation have been encountered in the set of known ground atomic formulae, proves essential for restoring consistency in a set of hypotheses.

The approach is clearly a restriction of the general problem of induction. However, induction, as such, plays a relatively minor role; it is used to suggest an initial (and usually inconsistent) set of hypotheses, which then are modified using strictly deductive techniques. The set of hypotheses that may be formed is shown to be perhaps surprisingly general and in fact (with respect to expressiveness) subsumes a number of existent systems for learning from examples.

In summary, I initially assume that:

1. the domain is describable as a set of ground atomic formulae.
2. some finite subset of the ground atomic formulae is known;
3. the set of known ground atomic formulae is correct and error-free;
4. the set of known ground atomic formulae grows monotonically with time:
5. known individuals may be tested for membership in a known relation.

The first assumption is somewhat restrictive, and in section 4 is relaxed so that a learning system may be "told" arbitrary sentences. Also in the third section, the second assumption is relaxed to allow relations in the domain whose membership is completely known.

The third assumption is clearly unrealistic for any practical learning system. However, arguably the issue of how to deal with erroneous data is a pragmatic one. and is not relevant to our concerns here. Consider, as illustration, where we have some conjecture (say. "ravens are black") and encounter an albino. If we don't want to totally abandon our original hypothesis, then there seems to be two ways we can discharge the exception. First, we could amend the conjecture to something like "normally ravens are black", and perhaps also introduce "normally albino ravens are white". That is. one way or another, the exception is "excused". Second we could determine, or simply declare, that the observation is erroneous. However, *this* procedure of determining that an observation is incorrect, or otherwise excusing it. is a pragmatic concern, and is distinct from our concern of what hypotheses "follow" potentially from a given set of ground atomic formulae.

1.2. Related Work

There has been much work in AI addressing the problem of learning from examples, including [Brown 73]. [Buntine 86], [Hayes-Roth 78]. [Mitchell 77]. [Shapiro 81]. [Solway and Riseman 77]. [Vere 78] and [Winston 75]. [Michalski 83] is a particularly detailed approach to learning from examples. An extensive survey of AI learning systems is given in [Dietterich et el 82]. while [Smith et al 77] describes a proposed model for learning from examples and [Dietterich and Michalski 83] compares four particular generalisation programs. Typically such work is concerned with proposing and refining a description of a concept, and many of the above approaches detail particular rules or strategies for forming a general concept from a set of instances. Most of this work also assumes that the only information regarding a domain is in the form of examples, or instances. In contrast, the work at hand deals with characterising the hypotheses formable under a set of (arguably) minimal assumptions and hence is concerned with exploring intrinsic properties and limitations of such approaches. In addition, an extension of learning from examples to include learning by being told is addressed.

The work presented in [Morik 86] employs assumptions similar to those used here, in that very little is assumed about the domain of application. However, in this case the author is interested in the problem of acquiring an initial model of a domain, but presupposes an agent and learning algorithms to help in this initial investigation. Thus this work addresses the first step of a number of steps in a technique-specific approach to learning.

Most formal approaches to learning from examples have been concerned with inducing instances of a given type of formal language. The area of *learning theory* [Gold 67] studies systems that implement functions from evidential states to languages. A survey of such approaches is presented in [Angluin and Smith 82], while [Osherson et el 83] gives recent results in this area. The key difference between such approaches and the present work is that no underlying formal grammar is assumed here, beyond that for elementary set theory.

2. Introducing Conjectural Information

2.1. Initial Considerations

The domain of application is assumed to be describable as a presumably infinite set of ground atomic formulae, formed from presumably infinite sets of individuals and predicates. At any point in time, the truth values of some subset of the ground atomic formulae are assumed to be known. Given a particular predicate then, all that can be known of it is a subset of those individuals (or tuples) which satisfy it and a subset of those individuals which do not. A predicate will be referred to as *known* if its truth value on a given individual (tuple) is known or can be determined. An individual will be referred to as *known,* if it is known to be or not be part of the extension of a known predicate. Informally, a known individual or predicate is one "encountered" by a learning system. The sets of tuples known to belong to the extension of a predicate and known to not belong to the extension are referred to as the *known extension* and the *known antiextension* respectively. So for a known n-place predicate *P* and known individuals a,, . . . ,a_n there are three possibilities:

1. $Pia_x, \ldots, a_{n})$ is known to be true.
2. -P(a_1 . . . , a_n) is known to be true.
3. Neither $P(a_1 \ldots . a_{n})$ nor -P(a_1 ,a_n) are known to be true.

Definition: For each known predicate symbol P define sets P_+ and P_ by:

P+ = {<a_1 . . . ,a_n> | P($a_1 \ldots . a_{n}$) is known to be true).

P_ = {<$a_1 \ldots . a_{n}$> | -P($a_1 \ldots . a_n$) is known to be true}.

Thus, to this point, the pair $<P_+, P_->$ expresses all that *can* be known about predicate P.

Conjectures are expressed in a language HL that is analogous to that of elementary set theory except that, for contrast, operators and relations are subscripted with the character "h". Most of the subscripted symbols are familiar; however "$|_h$" is used for the (hypothesised) disjointness relation, and "$\propto_h$", "$\circ_h$", and "$\uparrow_h$" for the converse, composition, and image operations. Note also that we will later distinguish the hypothesised equality of terms of HL, "$=_h$", from strict equality of terms, "$=$".

Definition: If **P** is the set of known predicate names, then the *terms* of HL are exactly those given by:

1. If $P \in \mathbf{P}$ then P is a term of HL.
2. If α, β are 2-place terms and γ a 1-place term of HL, then $\propto_h\alpha$, $\alpha\circ_h\beta$, and $\alpha\uparrow_h\gamma$ are terms of HL.
3. If α and β are terms of HL, then so are $\alpha\cap_h\beta$, $\alpha\cup_h\beta$, and $\neg_h\alpha$.

Definition: The *sentences* of HL are exactly given by:

If α, β are terms of HL, then $\alpha=_h\beta$, $\alpha\subset_h\beta$, $\alpha\subseteq_h\beta$, $\alpha|_h\beta$, $\alpha\neq\beta$, $\alpha\not\subset\beta$, $\alpha\not\subseteq\beta$, $\alpha\not|\beta \in$ HL.

So, for example,

$$River\cup_h Lake\cup_h Sea \subseteq_h Geo\_Feature$$

has the reading "the set of rivers, lakes, and seas is hypothesised to be contained in the set of geographical features" while

$$Uncle =_h (Brother\circ_h Parent)\cup_h (Husband\circ_h Sister\circ_h Parent)$$

is the familiar definition of uncle.

The known extension and antiextension of a known predicates are, by definition, known. From these, the known extensions and antiextensions of general terms in HL can easily be determined. Thus, for example, the *hypothetical intersection* of P and Q is known to contain just those elements that both P and Q are true of, and is known to not contain just those elements that either P or Q is known to not be true of. For the hypothesised operations we obtain the extension/antiextension pairs:

Proposition:

Complement: $\neg_h P$ is $<P_-, P_+>$.

Intersection: $P\cap_h Q$ is $<P_+\cap Q_+, P_-\cup Q_->$.

Union: $P\cup_h Q$ is $<P_+\cup Q_+, P_-\cap Q_->$.

Converse: $\propto_h P$ is
$<\{<y,x> \mid <x,y> \in P_+\}, \{<y,x> \mid <x,y> \in P_-\}>$.

Image: $P\uparrow_h Q$ is
$<\{y \mid (\exists x)(<x,y> \in P_+ \wedge x \in Q_+)\}, \varnothing>$
for binary relation P and one-place predicate Q.

Composition: $P\circ_h Q$ is
$<\{<x,z> \mid (\exists y)(<x,y> \in P_+ \wedge <y,z> \in Q_+)\}, \varnothing>$.

These operations generalise easily to ternary and higher-order predicates. Other operations, including domain and range, may clearly be defined in terms of these.

All that can be known about a term of HL is the known extension and antiextension. On the basis of these sets we want to form conjectured relations between terms. Now, naively, two terms of HL may be conjectured to be equal when there is some reason to do so (i.e. some evidence criteria is satisfied) and there are no known counter-examples. We obtain:

Definition: $\alpha=_h\beta$ when $\alpha_+ \cap \beta_+ \not\supseteq f$ and $\alpha_+ \cap \beta_- = \varnothing$ and $\alpha_- \cap \beta_+ = \varnothing$.
for some "evidence function" f.

The only requirement that we place on f is that the intersection of α_+ and β_+ be non-empty, and that f be uniformly applied across hypothesised statements of equality. Conditions for containment and disjointness can also be easily specified:

Definition: $\alpha\subseteq_h\beta$ iff $\alpha\cap_h\beta =_h \alpha$.
$\alpha\subset_h\beta$ iff $\alpha \subseteq_h \beta$ and $\alpha \neq \beta$.
$\alpha|_h\beta$ iff $\alpha \subseteq_h \neg_h\beta$.

This approach to forming conjectures, of course, is hopelessly simplistic. For example, assume that *SM* means that one can supervise M.Sc. students, while *SP* and *HP* means that one can supervise Ph.D. students or has a Ph.D. (respectively). If all that is known is that

$$SM(John),\ SP(John),\ HP(John),\ \text{and}\ SM(Mary),\ \neg SP(Mary)$$

then if the evidence function f were simply equal to unity, we would have, according to our naive criteria:

$$SP \subset_h SM,\ SP =_h HP,\ \text{along with}\ HP =_h SM.$$

This clearly is inconsistent. A potential solution to this difficulty is to determine the truth values of select ground instances, where both the predicate and the individual are known, but where the truth value of the ground instance is not known. Thus, if $HP(Mary)$ was determined to be true, then $SP \subset_h HP$ could be formed; if $HP(Mary)$ was determined to be false, then $HP =_h SM$ could be weakened to $HP \subset_h SM$.

These considerations seem to imply that on occasion we need to be able to determine the value of some $P(a_1, \ldots, a_n)$, provided that P and $a_1, \ldots, a_n$ are known. However, in general we would not want to determine the truth values of all known predicates applied to all combinations of known individuals. The reason for this is combinatorial: given p known n-place predicates and m individuals, there are pm^n knowable ground instances. In the approach to be described, at most $p(p-1)$ of these combinations need to be known for hypothesising relations.

While the previous example produced an inconsistency, we also may obtain hypothesised statements that are weaker than would be expected. Thus, according to the naive criteria, is is quite possible that there is evidence for $\alpha =_h \beta$ but not for $\neg_h\alpha =_h \neg_h\beta$. As will be seen, this difficulty cannot be rectified in any system based on our initial assumptions. So three questions arise:

1. How can "select" ground instances be determined for the restoration of consistency?
2. How can we characterise the conjectures to which such a procedure for restoring consistency can be applied?
3. Given that the relations among hypotheses are weaker than the relations among corresponding statements of elementary algebra, in what ways are these relations weaker?

These questions are answered in the next two subsections by examining the algebra of the known extensions and antiextensions of terms of HL.

2.2. An Algebra for Forming Hypotheses

Two terms of HL are defined to be (strictly) equal when their known extensions and antiextensions coincide. Containment ($\leqslant$) is introduced by the usual definition. Hence:

Definition: For α, β terms of HL,

$\alpha = \beta$ iff $\alpha_+ = \beta_+$ and $\alpha_- = \beta_-$.
$\alpha \leqslant \beta$ iff $\alpha\cap_h\beta = \alpha$.

The resultant algebra HLA is given by HLA $= [\mathbf{H}; \neg_h, \cap_h, \cup_h]$, where the carrier **H** is:

$$\mathbf{H} = \{<a,b> \mid a,b \subseteq \mathbf{I} \text{ and } a\cap b = \varnothing\}$$

for a set of known individuals I. The pair of elements in an element of H corresponds to a possible known extension/antiextension pair. Upper and lower bounds of H are given by:

Definition: $1 = \langle I, \emptyset \rangle \quad 0 = \langle \emptyset, I \rangle$

We obtain the following postulates:

Postulates:

P1 $\alpha \cap_h \beta = \beta \cap_h \alpha \qquad \alpha \cup_h \beta = \beta \cup_h \alpha$

P2 $\alpha \cap_h (\beta \cap_h \gamma) = (\alpha \cap_h \beta) \cap_h \gamma$
$\alpha \cup_h (\beta \cup_h \gamma) = (\alpha \cup_h \beta) \cup_h \gamma$

P3 $\alpha \cap_h (\alpha \cup_h \beta) = \alpha \qquad \alpha \cup_h (\alpha \cap_h \beta) = \alpha$

P4 $\alpha \cap_h (\beta \cup_h \gamma) = (\alpha \cap_h \beta) \cup_h (\alpha \cap_h \gamma)$
$\alpha \cup_h (\beta \cap_h \gamma) = (\alpha \cup_h \beta) \cap_h (\alpha \cup_h \gamma)$

P5 $\alpha \cap_h \alpha = \alpha \qquad \alpha \cup_h \alpha = \alpha$

P6 $\alpha \cap_h (\beta \cup_h (\alpha \cap_h \gamma)) = (\alpha \cap_h \beta) \cup_h (\alpha \cap_h \gamma)$
$\alpha \cup_h (\beta \cap_h (\alpha \cup_h \gamma)) = (\alpha \cup_h \beta) \cap_h (\alpha \cup_h \gamma)$

P7 $\alpha \cap_h 0 = 0 \qquad \alpha \cup_h 0 = \alpha \qquad \alpha \cap_h 1 = \alpha \qquad \alpha \cup_h 1 = 1$

P8 $\alpha = \neg_h \neg_h \alpha$

P9 $\neg_h(\alpha \cup_h \beta) = \neg_h \alpha \cap_h \neg_h \beta \qquad \neg_h(\alpha \cap_h \beta) = \neg_h \alpha \cup_h \neg_h \beta$

P10 $\alpha \cap_h \neg_h \alpha \leqslant \beta \cup_h \neg_h \beta$

These postulates very nearly, but don't quite, characterise Boolean algebras. Instead of a postulate for a universal complement,

$$\alpha \cap_h \neg_h \alpha = 0, \qquad \alpha \cup_h \neg_h \alpha = 1$$

we obtain the weaker "Kleene" postulate P10. However we retain postulates governing universal bounds (P7) and involution (P8) as well as De Morgan's laws (P9). The weakened complement arises from the fact that the known extension and antiextension of a predicate typically do not together constitute the set of known individuals 1. This algebra has been investigated under the names of *normal involution lattices* [Kalman 58] and *Kleene algebras* [Kleene 52].

To further characterise the conjectures, the propositional logic corresponding to HLA is derived in [Delgrande 85], and soundness and completeness results are obtained. An alternative, three-valued semantics for the logic is also provided This development is not repeated here. However, it is worth noting that, unsurprisingly, negation in the logic. HLL. is weaker than in classical propositional logic: we lose *reductio ad absurdum* as a method of proof: also we lose the law of the excluded middle. Finally, it may be noted that HLL appears in [Rescher 69] as the system S3. However this system is mentioned only in passing, and no results concerning it are provided.

23. Restoring the Consistency of Hypotheses

This section examines how the consistency of a set of conjectures can be enforced and maintained in the face of conflicting ground instances. The maintenance of consistency relies of the relationship between the naive criteria for forming conjectures in HL and conditions for equality of expressions of HLA. Consider for example the criteria for hypothesised equality:

$\alpha =_h \beta$ when $\alpha_+ \cap \beta_+ \geqslant f$ and $\alpha_+ \cap \beta_- = \emptyset$ and $\alpha_- \cap \beta_+ = \emptyset$

and contrast it with strict equality:

$\alpha = \beta$ iff $\alpha_+ = \beta_+$ and $\alpha_- = \beta_-$.

For hypothesised equality we are not guaranteed, for example, transitivity of equality. So if we have $P =_h Q$ and $Q =_h R$ we are not guaranteed $P =_h R$. For strict equality we of course have transitivity.

Now. in order to have asserted $P =_h Q$. there must have been some set of individuals e which provided evidence for this hypothesis. Clearly, these instances will either also provide evidence for $P =_h R$ (i.e. will satisfy the naive criteria), or if not. will refute both $P =_h R$ and $Q =_h R$, and so. in any case, consistency can be restored. We can use a similar procedure for restoring the consistency of any set of hypotheses which should imply a certain conclusion in HLA but. for a particular case, do not. We obtain:

Theorem: Let $a_1, \ldots, a_n, a \in$ HL and assume $a_1, \ldots, a_n$ have been hypothesised according to our naive criteria, and a is derivable from $a_1 \ldots a_n$ in HLA. Then ground instances can be determined from the set of confirming instances for $a_1, \ldots, a_n$ that will either

1. refute one of $a_1 \ldots, a_n$ or
2. allow a to be hypothesised.
3. allow some a to be hypothesised where o' implies a in HLA.

Outline of Proof: Let the sequence of steps in a proof of α from $\alpha_1, \ldots, \alpha_n$. be $\zeta_1, \ldots, \zeta_m$ where $\zeta_m = \alpha$. I show that if there is evidence for $\zeta_1, \ldots, \zeta_i$ for $1 \leqslant i < m$ then there is evidence for ζ_{i+1}. or one of the original premisses. $\alpha_1, \ldots, \alpha_n$. can be falsified. "Evidence" is defined to consist of any metric that obeys the naive criteria of section 2.1. Since a proof is a sequence of steps from original premisses to desired conclusion, the theorem follows immediately. Note that it does not matter *which* proof of α from $\alpha_1, \ldots, \alpha_n$ is selected.

Alternative 3 in the theorem takes care of a "glitch" in the relation between strict equality and hypothesised equality of terms of HLA: hypothesised equality requires that the extensions of the terms in question have a non-empty intersection, whereas strict equality simply requires that the extensions and antiextensions coincide. Thus it is possible to have $P =_h Q$. based on a set of individuals e. However such a set of individuals will not satisfy the naive criteria for $\neg_h P =_h \neg_h Q$ and so in this case we cannot use e to provide evidence for $\neg_h P =_h \neg_h Q$. This is despite the fact that we can prove $\neg_h P = \neg_h Q$ from $P = Q$ in HLA.

The theorem guarantees for example that if we can hypothesise that

$A \subseteq_h B, B \subseteq_h C, C \subseteq_h D$. but not $A \subseteq_h D$.

then we can locate individuals and predicates that will, via our criteria for evidence, allow the conclusion AQ_hD to be hypothesised, or else will refute one of the premisses.

The proof of the theorem is constructive, and leads immediately to a procedure which will either locate evidence for a conjecture a that follows from supported premisses $O_1, \ldots, a_n$. or else will refute one of $a_1, \ldots, a_n$. In the third alternative for the theorem, a specific sentence a of HLA which implies a is identified. The procedure is linear in the length of the proof of a. Since the evidence required for the naive criteria for forming conjectures is drawn from finite sets of individuals and relations, consistency can thus be restored in a set of conjectures by repeatedly applying this procedure. Moreover, if we begin solely with a set of ground instances we will, by repeated application of our naive criteria for forming conjectures together with this procedure, arrive at a set of consistent conjectures.

This resolves the three questions posed at the end of section 2.1 First, the procedure derived from the above theorem shows how instances can be located for the restoration of consistency. Second, the conjectures to which the procedure can be applied are those that are governed by the postulates of HLA. Thirdly, then, the conjectures to which the procedure may be applied correspond precisely to the sentences of elementary algebra

(including composition, the converse, and the image), except that we do not have a universal complement. Moreover, this limitation is unavoidable in this approach (or any approach based on the five assumptions listed in the introduction). However, we do retain involution. De Morgans laws, laws concerning universal bounds, and the "Kleene" postulate.

3. HLA, Boolean Algebra, and Naive Set Theory

This section describes an extension to the approach wherein entities in the domain whose extensions are completely known are also considered. So far 1 have assumed that for a particular predicate in the domain, such as *Red.* only part of the extension and part of the complement of the extension can be known. The set of *all red* things (i.e. the extension of *Red)* cannot wholly be known by a learning system. While this seems reasonable for things such as *Red, Bird,* and *Left_of.* in other cases it is unreasonable. For example. I may know that a particular suck of blocks is composed of the blocks $\{S_1.....s_K\}$ In this case we could perhaps name this set slack $Stack_1$ and so $Stackl = \{S_1.....s_k\}$. The crucial point here, of course, is that all the members of *Stackl* are known, in contrast to the membership of *Red.* which cannot be completely known. Sets such as *Stackl,* whose extension is known. I will refer to as *reducible.* Sets such as *Red.* whose extension cannot be wholly known. 1 will refer to as *irreducible.* Reducible sets, clearly, correspond exactly with the familiar notion of "set". In this paper though I will be concerned only with sets with a finite extension.

So there are two questions of interest:

1. How can we formally characterise the irreducible sets?
2. How do the reducible and irreducible sets interrelate?

These questions are addressed by giving a set of axioms to characterise the set of allowable reducible and irreducible sets. For both reducible and irreducible sets, the axioms developed parallel those in the system of Zermelo-Fraenkel [Fraenkel et al 73]. We obtain:

Notation:

The letters $a, b, c, \cdots$ will stand for reducible sets.
The letters $A, B, C, \cdots$ will stand for irreducible sets.
The letters $\cdots, x, y, z$ will stand for either reducible or irreducible sets.

Set Axioms:

Existence:

i) If $a \in \mathbf{UP}$, $\{a\}$ is a set.

ii) If $a, b \subseteq \mathbf{UP}$ and $a \cap b \neq \emptyset$ then (a, b) is an irreducible set.

Extensionality:

i) $(a)(b)(x)(x \in a \equiv x \in b) \supset a = b$

ii) $(A)(B)(x)((x \in A_+ \equiv x \in B_+) \wedge (x \in A_- \equiv x \in B_-)) \supset A = B$

Pairing: $(x)(y)(\exists a)(z)(z \in a \equiv (z = x \vee z = y))$

Sum:

i) $(a)(\exists b)(x)(x \in b \equiv (\exists c)(x \in c \wedge c \in a))$

ii) $(a)((\exists D)(D \in a) \supset$
$(\exists B)(x)[x \in B_+ \equiv$
$((\exists c)(x \in c \wedge c \in a) \vee (\exists C)(x \in C_+ \wedge C \in a))] \cdot$
$\wedge [x \in B_- \equiv \neg(\exists c)$
$((x \in c \wedge c \in a) \wedge (C)(x \in C_- \wedge C \in a))])$

iii) $(A)(\exists C)(x)([x \in C_+ \equiv (\exists B(x \in B_+ \wedge B \in A_+)$
$\vee (\exists b)(x \in b \wedge b \in A_+))] \wedge C_- = \emptyset)$

Power Set:

i) $(a)(\exists b)(c)(c \in b \equiv c \subseteq a)$

ii) $(A)(\exists B)(c)((c \in B_+ \equiv c \subseteq A_+) \wedge$
$(c \in B_- \equiv (c \subseteq (A_+ \cup A_-) \wedge c \not\subseteq A_+)))$

Separation:

i) $(a)(\exists b)(x)(x \in b \equiv x \in a \wedge \sigma(x))$ for b not free in σ, and σ reducible or irreducible.

ii) $(A)(\exists B)(x)((x \in B_+ \equiv (x \in A_+ \wedge \sigma(x))) \wedge$
$(x \in B_- \equiv (x \in A_- \vee \neg\sigma(x))))$
for B not free in σ, and σ reducible or irreducible.

Regularity: $(a)(a \neq \emptyset \supset (\exists x)(x \in a \wedge (y)(y \in x \supset y \notin a)))$
where for x irreducible, $y \in x$ means $y \in x_+$.

The axioms for irreducible sets are derived from intuitions similar to those motivating the axioms for the reducible sets. For extensionality, for example, we would want to say that two irreducible sets are equal just when their known extension and antiextension coincide. For the power set axiom, for every irreducible set A we know that the power set of A must contain the power set of A_+; moreover, this is all that can be known to be in the power set. On the other hand, any set of elements not contained in A_+ cannot be in the power set. and hence are in the antiextension. [Delgrande 85] contains an informal discussion and development of the other set axioms.

The axiomatisation then provides a set of constraints that bound the irreducible sets and. moreover, specifies how they may interrelate. As well it provides a primitive basis for defining and justifying the hypothetical operations of the previous section. For example, it is easily shown using the axioms of separation and extensionality that:

Theorem: $(A)(B)(\exists C)(x)((x \in C_+ \equiv x \in A_+ \wedge x \in B_+)$
$\wedge (x \in C_- \equiv x \in A_- \vee x \in B_-))$ and C is unique.

So for any two irreducible sets there is a third set whose known extension consists of elements common to the known extensions of the first two sets and whose known antiextension consists of elements in either of the known antiextensions of the two sets. In addition, this third set is unique. This in turn justifies the definition:

Definition: $A \cap_h B = C$ iff $(x)((x \in C_+ \equiv x \in A_+ \wedge x \in B_+) \wedge$
$(x \in C_- \equiv x \in A_- \vee x \in B_-))$.

Thus, $A \cap_h B$ is $< A_+ \cap B_+, A_- \cup B_- >$. The other hypothetical operations can be similarly defined. However, we can go beyond this, and consider what we would obtain from applying an operator to a reducible and an irreducible set. For example. for intersection we obtain:

Definition: $A \cap_h b = c$ iff $(x)(x \in c \equiv (x \in b \wedge x \in A_+))$.

Thus $A \cap_h b$ is the reducible set $A_+ \cap b$. Thus the result of intersecting a known set of blocks with the (irreducible) predicate *Red* is the subset of the blocks known to be red.

Lastly, we can consider an extension of the language HL where predicates that apply to sets of objects can be related to those that apply to only single (pairs of) objects. For example, the notion of hypothesised transitive closure is introduced into HLA in [Delgrande 85]. The system augmented by the transitive closure is general enough to allow for the expression of (the oft-cited example of) an arch. Thus we can express hypotheses concerning arches, for example, that a stack of objects is a set of objects that satisfies the transitive closure of the *On* relation and. conversely, any set of objects so bounded is hypothesised to be a stack. From this an arch could be defined as an object consisting of two pillars which are stacks, a lintel which is on top of the pillars where the pillars are not touching, and so on. The limitations of the previous section of course still apply, and so any such hypotheses are bound by the relations of HLA.

The set axioms then provide a more primitive basis for forming conjectures and, additionally, expand the expressiveness of the system. However, while the set of allowable individuals has been vastly expanded, the form of the conjectures is

unchanged. The formal results of the last section still apply and. since in the intended model the property of being an individual is decidable, the overall system remains decidable.

4. Learning from Examples and Learning by Being Told

In this section 1 consider where we have a knowledge base (KB) that consists of an arbitrary, consistent set of sentences that the system has been "told" are true. We now want to form hypotheses from the ground instances, but also taking these other sentences into account. Since we require that the KB be able to reason about knowledge and hypothesis, this part of the problem either requires or presupposes a theory of incomplete knowledge. 1 have taken the latter course, and adopted the theory given in [Levesque 81]. This work presents a logical language KL that can refer both to application domains and to what a knowledge base might know about such domains. KL extends first-order logic (FOL) by adding a sentential operator K, where Ka can be read as "a is known to be true". KL is extended here to a language called HKL that is able to deal also with conjectural sentences. HKL extends KL by the addition of a sentential operator H. where Ha can be read as "a is conjectured to be true". Using HKL we can express sentences such as "John or Bill is hypothesised to be a teacher" or "it is known that Mary is hypothesised to be a teacher". HKL is specified as follows:

Axiom Schemata

1. The axioms of FOL
2. $K\alpha$. where α is an axiom of FOL
3. $K(\alpha \supset \beta) \supset (K\alpha \supset K\beta)$
4. $\alpha \equiv K\alpha$ where terms of α are within the scope of a K or H
5. $(x)K\alpha \supset K(x)\alpha$
6. $K\alpha \supset (\neg H\alpha \wedge \neg H\neg\alpha)$
7. $K(\alpha \supset \beta) \supset (\neg K\beta \supset (H\alpha \supset H\beta))$
8. $H(\alpha \supset \beta) \supset (\neg K\beta \supset (K\alpha \supset H\beta))$
9. $H(\alpha \supset \beta) \supset (\neg K\beta \supset (H\alpha \supset H\beta))$
10. $((x)(K\alpha \vee H\alpha) \wedge (\exists x)\neg K\alpha) \supset H(x)\alpha$

Rules of Inference - Modus ponens and universal generalisation

The first five axiom schemata and the rules of inference are those of KL. We obtain that knowledge and conjecture are closed under modus ponens (3. 7, 8. 9). and meta-knowledge is complete and accurate (4). Also if something is known, neither it nor its negation is conjecture (6). and generalisation applies analogously to conjecture and knowledge (5. 10). What this extension from KL to HKL buys us is a means of distinguishing and reasoning with sentences known to be true, from those that are only hypothesised to be true. In [Delgrande 85] an extension of the soundness and completeness results of [Levesque 81] (with respect to a possible worlds semantics) is provided.

HKL seems to have reasonable properties with respect to reasoning deductively with knowledge and hypothesis. There is a problem however with updating a KB (i.e. with "telling" a KB a new sentence). Consider where we have a known portion. KB_K, of the KB and we want to form a hypothetical component. KB_h,, based on the known portion. Basically we want to "apply" HL to this KB to produce a hypothesised component. Thus for example if KB_k is

$$P(a),\ (x)(P(x) \supset Q(x)),\ R(a),$$

then applying HL to what is known about the ground instances could yield:

$$P =_h Q,\ Q =_h R,\ P =_h R$$

or the equivalent hypothetical KB in HKL

$$(x)(P(x) \equiv Q(x)),\ (x)(Q(x) \equiv R(x)),\ (x)(P(x) \equiv R(x)),$$

The basic idea is that KB_K, which is expressed in HKL. determines a set of ground instances and a set of sentences that are representable in HL. By applying the procedure of section 2.3 for restoring consistency to these sets we obtain a set of hypotheses expressed in HL. If KB = [KB_K. KB_h.] where initially KB/, = 0 then we have the following procedure for forming a hypothetical component.

1. Let G = |g|g is a ground atomic formula and KB*h
2. Apply the procedure of section 2.3 to G to obtain a consistent set of conjectures C expressed in HL.
3. Let KB_h, be the translation of the sentences of C into sentences of HKL; exclude any of those provable in KB_k

There is generally a straightforward translation of sentences from HL to HKL and. given this, the above procedure can be constructed in a straightforward manner. However it proves to be the case that unless KB_k is equivalent to a set of ground instances, this procedure may result in inconsistency. The difficulty is that the procedure of section 2.3 relies on the existence of knowable ground instances whose truth value is unknown. However in first-order logic, and so in HKL. it is possible to attribute a property to an unknown individual, and this attribution may lead to inconsistency here. For example, consider where all that is known is

$$P(a),\quad Q(a),\quad R(b),\quad S(b),$$
$$(\exists x)(P(x) \wedge \neg Q(x)) \vee (\exists x)(P(x) \wedge \neg S(x))$$

Hence, for the corresponding relations in the domain, either *P*=*Q* or *R*=*S*. If we apply the procedure for restoring consistency to the known ground instances and sentences that can be expressed in HL. we obtain

$$P =_h Q \text{ and } R =_h S$$

which, of course, when translated into HKL. is inconsistent with the original sentences.

While this last result appears somewhat limiting, things in practice may not be too bad. Several considerations are relevant. First, the assumptions underlying HL are those that presumably underlie any system that learns from examples. Hence the problems addressed here arguably are the problems that must be addressed by any system that learns from examples, or else must be discharged by means of *a priori* decisions by the system designers. Second, while applying the procedure to a general KB may lead to inconsistency, it need not necessarily do so. If it does, it may be possible that pragmatic considerations can be used to resolve or skirt a particular inconsistency.

5. Comparison with Learning Systems

This section compares the present approach with related work on learning from examples. Three systems are particularly relevant and serve to place the present work within the field. The early work of John Seely Brown [Brown 73] on automatic theory formation is a direct precursor to mine. Patrick Winstons dissertation [Winston 75] on learning structural descriptions from examples is a well-known early AI learning system and serves as a good representative of approaches to learning from examples. Ehud Shapiro's work [Shapiro 81] is similar to mine in broad outline, except that the author makes substantial assumptions, concerning how the domain *of* application is described.

The task of Browns system is to propose definitions for a set of binary relations based on knowledge of the extensions of the relations. The system begins with a set of binary relations

$R = \{R_1 \ldots R_n\}$ and a database containing the complete extension for each $R_i \in R$. Hence there is no notion of modifying a definition in light of later knowledge. A procedure is given for proposing definitions of the relations. However the body of a definition is restricted to be the disjunction of compositions of relations. This format though is adequate for a variety of domains, including that of kinship relations. The system is heuristic and was intended for direct implementation. Thus it dealt with matters such as efficiently searching for possible definitions, proposing definitions in a "simplest first" manner, etc. No analysis is carried out with regard to what may be conjectured, nor is an algorithmic analysis of the system given. In contrast 1 have not addressed implementation issues, but rather have attempted to address general problems of hypothesis formation, and thus issues dealing with characterising a set of conjectures and maintaining the consistency of a set of conjectures.

Winston's work was briefly described in the introduction. Basically Winston is concerned with the pragmatic aspects of a learning system, and concentrates on techniques to speed the learning process. In some sense then his approach is complementary to the one taken here. The system of section 3 subsumes the conjectures that may be formed in Winston's system: thus anything that can be formed in his system can also be conjectured in HL. (Winston actually gives a semantic net representation for his concepts. This representation however is clearly equivalent to a set of binary relations, and is useful mainly as a notational or implementational device.) However, in HL the set of formable conjectures and the means of restoring consistency are precisely laid out whereas Winston does not address these issues.

Shapiro's work is. superficially, the most similar to that presented here. Shapiro assumes that a domain is described by a stream of ground instances; based on the ground instances, a set of conjectured axioms for the domain is proposed and refined. A general, incremental algorithm for proposing a set of rules which imply the known ground instances is developed. The algorithm has tuneable parameters that determine the complexity of the structure of a hypothesis. The key difference between Shapiro's work and the work at hand is that Shapiro makes substantial assumptions about the way the domain is described. In particular, the domain is assumed to be describable by a set of rules in the form of restricted Horn clauses In addition the user is given some control over the form of the hypotheses. These assumptions allow an elegant algorithm for inducing rules for a wide class of problems to be derived. In the approach at hand, in contrast, the emphasis is on what may potentially be formed, rather than what can efficiently be induced.

6. Conclusion

This work develops a formal, unified, and general (but basic) framework for investigating learning from examples. A primary goal was to keep the approach as general as possible and independent of any particular domain, representation scheme, or set of learning techniques. Hence, for example, there is no restriction placed on the ordering of the ground instances nor is there any assumption that the input examples have been already aggregated into complex entities such as arches. Neither is there any restriction with regard to introducing new ("known) predicate names during the learning process. Also, no agent is assumed to exist, to help direct or focus the acquisition process.

Presumably the issues addressed here are common to, and are relevant to. any system for learning from examples (at least any system that satisfies the five assumptions given in the introduction). Hence the framework may be appropriate as both a basis for the development of systems that learn from examples, and perhaps as a neutral point from which such systems may be viewed and compared. However only a set of formal issues have been addressed, and the concern has been with what conjectures may potentially be formed, rather than with which of those conjectures should in fact be held. Pragmatic issues concerned with the justification of conjectures, strength of evidence, and degrees of confirmation, to name just a few. are outside the scope of this work.

Formal systems are developed for introducing and maintaining the consistency of conjectures. An exact specification of what conjectures may potentially be formed is provided, and it is shown how the consistency of a set of conjectures can be restored in the face of conflicting instances. The system illustrates that a reasonably rich and expressive set of conjectures can be derived using only a minimal set of assumptions. Two extensions to the system are described. First the system is augmented to allow relations in the domain whose extension is completely known; to this end an axiomatisation of the (so-called) reducible and irreducible sets is provided. Also addressed is learning from examples, but where the system may be told arbitrary sentences in addition to the ground instances. The first extension does not affect the formal results previously obtained; the second is limiting, in that it may give rise to problems with inconsistency that have to be resolved by pragmatic means.

The expressiveness of the system is indicated by the fact that it is as at least as general as a number of existing systems, including [Brown 73]. [Hayes-Roth 78). [Vere 78]. and [Winston 75]. Results concerning decidability lend credence to the possibility that learning systems based directly on this approach and. in particular, incorporating the procedure for restoring consistency, may be efficiently implementable. In addition, given the generality of the approach, it is possible that the framework could also provide an appropriate starting point for an investigation of other types of learning systems. That is. the approach could conceivably be extended by incorporating further assumptions concerning the domain, underlying representation scheme, or an agent to assist in the learning.

The approach as it stands may have immediate practical applications. As a specific example, database systems often use integrity constraints to partly maintain consistency and reliability. However, given a large number of relations, it is an arduous task to specify all integrity constraints and to ensure that the set is consistent. The approach then seems suited to the task of automatically proposing and verifying such constraints. This possibility is explored in [Delgrande 87].

Acknowledgement

This paper is based on my doctoral dissertation in the Department of Computer Science at the University of Toronto. 1 would like to thank my supervisor. John Mylopoulos. for his guidance, as well as Graeme Hirst. David Israel, and Hector Levesque. Financial assistance from the Province of Ontario and the Department of Computer Science. University of Toronto, is gratefully acknowledged.

Bibliography

[I] D. Angluin and C.H. Smith, "A Survey of Inductive Inference: Theory and Methods". Computing Surveys, 15,3, September 1983 University. 1982

[2] J.S. Brown, "Steps Toward Automatic Theory Formation". Proceedings of the Third International Conference on Artificial Intelligence, Stanford. Ca.. 1973. pp 121-29

[3] J.P. Delgrande. "A Foundational Approach to Conjecture and Knowledge". Ph.D. thesis. Technical Report CSRI-173. Department of Computer Science. University of Toronto. September 1985

[4] J.P. Delgrande. "Formal Bounds on the Automatic Generation and Maintenance of Integrity Constraints", 6th ACM Symposium on Principles of Database Systems. March 1987

[5] T.G. Dietterich. B. London. K. Clarkson and G. Dromey. "Learning and Inductive Inference". The Handbook of Artificial Intelligence, P.R. Cohen and E.A. Feigenbaum (eds.). William Kaufmann Inc.. 1982

[6] T.G. Dietterich and R.S. Michalski. "A Comparative Review of Selected Methods for Learning from Examples", in Machine learning: An Artificial Intelligence Approach. R.S. Michalski, J.G. Carbonell. and T.M. Mitchell (eds.), Tioga. 1983

[7] A.A. Fraenkel. Y. Bar-Hillel and A. Levy, Foundations of Set Theory 2nd revised ed.. North-Holland Pub. Co.. 1973

[8] E.M. Gold. "Language Identification in the Limit". Information and Control 10. 1967. pp 447-474

[9] F. Hayes-Roth, "The Role of Partial and Best Matches in Knowledge Systems", in Pattern-Directed Inference Systems, D.A. Waterman and F. Hayes-Roth (eds.). Academic Press. 1978

[10] J.A. Kalman. "Lattices with Involution". Transactions of the American Mathematical Society, vol. 87. 1958. pp 485-491

[11] S.C. Kleene. Introduction to Metamathematics. North Holland Pub. Co.. 1952

[12] H.J. Levesque. "A Formal Treatment of Incomplete Knowledge Bases", Ph.D. thesis. Department of Computer Science. University of Toronto, 1981

[13] R.S. Michalski, "A Theory and Methodology of Inductive Learning", in Machine Learning: An Artificial Intelligence Approach. R.S. Michalski. J.G. Carbonell. and T.M Mitchell (eds.). Tioga. 1983

[14] T.M. Mitchell. "Version Spaces: A Candidate Elimination Approach to Rule Learning". Proceedings of the Fifth International Conference on Artificial Intelligence, Cambridge. Mass.. 1977. pp 305-310

[15] K. Morik. "Acquiring Domain Models", in Proceedings of the Knowledge Acquisition for Knowledge-Based Systems Workshop. Banff. Canada. 1986

[16] D.N. Osherson, M. Stob and S. Weinstein, "Formal Theories of Language Acquisition: Practical and Theoretical Perspectives". Proceedings of the Eighth International Conference on Artificial Intelligence, Karlsruhe, West Germany, 1983

[17] N. Rescher, Many-valued Logic, McGraw-Hill. 1969

[18] E.Y. Shapiro, "Inductive Inference of Theories from Facts". Research Report 192. Department of Computer Science, Yale University. 1981

[19] R.G. Smith. T.M. Mitchell. R.A. Chestek and B.G. Buchanan. "A Model for Learning Systems". Proceedings of the Fifth International Conference on Artificial Intelligence. Cambridge. Mass.. 1977. pp 338-343

[20] E.M. Solway and E.M. Riseman, "Levels of Pattern Description in Learning". Proceedings of the Fifth International Conference on Artificial Intelligence, Cambridge. Mass.. 1977. pp 801-11

[21] S.A. Vere. "Inductive Learning of Relational Productions". in Pattern-Directed Inference Systems. Waterman and Hayes-Roth (eds.). Academic Press. 1978

[22] P.H. Winston. "Learning Structural Descriptions from Examples" in The Psychology of Computer Vision, P. Winston (ed). McGraw-Hill. 1975