A strategy for constructing new predicates in first order logic

Stephen Muggleton (steve@turing.ac.uk)
The Turing Institute
36 North Hanover St
Glasgow G1 2AD, UK *

Abstract

There is increasing interest within the Machine Learning community in systems which automatically reformulate their problem representation by defining and constructing new predicates. A previous paper discussed such a system, called CIGOL, and gave a derivation for the mechanism of inverting individual steps in first order resolution proofs. In this paper we describe an enhancement to CIGOL's learning strategy which strongly constrains the formation of new concepts and hypotheses. The new strategy is based on results from algorithmic information theory. Using these results it is possible to compute the probability that the simplifications produced by adopting new concepts or hypotheses are not based on chance regularities within the examples. This can be derived from the amount of information compression produced by replacing the examples with the hypothesised concepts. CIGOL's improved performance, based on an approximation of this strategy, is demonstrated by way of the automatic "discovery" of the concept of radiation. This example also demonstrates CIGOL's ability to ignore irrelevant background knowledge and deal with multiple interacting concepts.

*This work was supported by the British Government's Alvey Logic Database Demonstrator project and a grant from the US Army Research Institute for the Behavioural and Social Sciences through its European Research Office, London, England, Contract no. DAJA45-86-0047.

1 Introduction

A concept can only be learned if it can be represented. More than this, an appropriate representation language facilitates a simple and elegant description of a target concept. In [17] we describe a system called CIGOL which automatically develops its own representation language in order to efficiently represent target concepts. Initially CIGOL is provided with pertinent background knowledge in the form of Horn clauses in first order logic. CIGOL is then presented with a sequence of ground unit clauses, representing positive instances of the target concept. Following the presentation of each example CIGOL presents the user with a sequence of hypotheses which take one of two forms: either "Is X true?" or "What shall I call the following concept?". The first type of question involves a generalisation which could be used to derive previous examples. Note that each negative response from the user adds a neagtive instance to CIGOL's "training set" which otherwise would consist solely of positive instances. The second type allows the introduction of new relational predicates which enable the target concept to be represented more efficiently. Since these forms of generalisation are chained together, the introduction of a new predicate is typically followed by further generalisation and/or decomposition into related sub-concepts.

The generality of the approach used allows CIGOL to exhibit a number of facets of Machine Learning. Thus CIGOL can be classed with systems which carry out

1. **inductive concept formation** such as [11, 21]
2. **constructive induction** such as [22, 13]
3. **discovery** such as [10, 9, 6]
4. **generalisation of single examples using background knowledge** such as [5, 15, 24]

Unlike most learning systems described in the literature CIGOL uses an unrestricted form of first order Horn clause logic which allows predicate relations to take not only variables and constants as arguments but also complex terms. This allows CIGOL to learn not only simple structural concepts, but also more complex program fragments. The various hypothesis forming mechanisms employed by CIGOL are based on inverting individual steps of a resolution proof. This approach is a generalisation of the approaches used

by Sammut and Banerji [24], Muggleton [16] and Banerji [1]. Other strongly related work in progress can be found in Wirth [27] and Wrobel [28].

In [17] we provided a derivation for the inverse resolution operators employed by CIGOL but to a large degree left open the question of strategy of operator application. The result was that CIGOL as described in [17] proposed a number of irrelevant and uninteresting hypotheses based on the discovery of chance and unimportant regularities within the presented examples. In this paper we describe a formal framework for a strategy of operator application aimed at avoiding the generation and testing of uninteresting hypotheses. The strategy is based on results from algorithmic information theory[4]. The minimal bit size difference criterion unifies what are usually perceived as two different kinds of inductive gain. The gain which is produced by increasing the cover of a concept and the store-cost gain involved in simplifying the decription of the concept, possibly involving decomposition into simpler sub-problems. An approximation to the new strategy has been incorporated into a new version of CIGOL. Sample results of the improved behaviour of CIGOL are included for a discovery problem from Francis Bacon's *Novum Organum.*

2 Generality and inverse resolution

Using standard notation from logic one can define the generality relation between well-formed-formulae F_1 and F_2 as follows

$$F_1 \text{ is more general than } F_2 \text{ iff } F_1 \vdash F_2$$

where $F_1 \vdash F_2$ should be read as "F_1 *entails* F_2" or alternatively "F_2 *is provable from* F_1". Note that this simple definition allows us not only to compare the relative generality of atomic formulae and clauses but also the same relationship for arbitrary pairs of theories (sets of clauses). Buntine [3] describes an algorithm aimed at computing this generality relationship which he terms "generalised subsumption". For a fuller discussion of the subject of generality the reader is referred to Niblett [18]. Following Plotkin [19] we may more precisely define the setting of inductive learning using the following relationship

$$I \wedge B \wedge H \vdash E^+ \tag{1}$$

where I is background knowledge which is not pertinent to the present learning problem, B is background knowledge which is pertinent to the problem,

H is an hypothesis consisting of one or more clauses and E^+ is a set of positive examples. In addition, if E^- is a set of negated formulae representing counter-examples then we can guard against over-generalisation by ensuring that $I \wedge B \wedge E^- \wedge H$ is not *unsatisfiable*, i.e. self-inconsistent. The explicit definition of the inductive setting described by (1) allows us to both pose and answer the following questions concerning the approach to learning embodied in CIGOL. Since this setting is analogous to both scientific theory formation and automatic program construction the author will indulge in a certain amount of mixed metaphor in the following discussion.

Question 1 *How can we construct H given I, B, and E^+?*

Answer: All methods outside enumeration and testing of H rely on applying efficient "generalisation operations" which incrementally construct H from B and E^+. Michalski [12] notes that these generalisation operations can be based on reversing the deductive rules of inference which allow us to derive E^+ from B and H. Michalski's INDUCE system uses the inversions of a wide variety of deductive rules of inference. However, we note that this is somewhat analogous to the position in theorem proving before the introduction of the universal rule of deductive inference known as resolution [23]. The thesis behind CIGOL is that appropriate inversions of resolution provide an efficient, sufficient and complete mechanism for the inductive setting described by (1).

Question 2 *Given that the predicate vocabulary used in E^+ and E^- might reasonably be limited to the "observation language" of experimentation and measurement, how do we develop a "theoretic language" of predicates which cannot be directly observed?*

Answer: Clearly the theory described by B and H can in principle contain predicates which, although relevant to the entailment of the observations E^+ and E^-, are not expressed in the vocabulary of E^+ and E^-. In CIGOL this "theoretic vocabulary" is introduced via the "W" operator (Intra-construction) (see [17]) and generalised and integrated into B using the "V" operator (Absorption).

Implementation details relating to answers 1 and 2 are given in [17]. However, familiarity with practical implementation details and the methods of scientific investigation would suggest that we must at least address the following additional questions.

Question 3 *How do we effectively constrain the generation of possible hypotheses H?*

Question 4 *How can we judge our confidence in any particular H?*

Question 5 *What is the criterion for distinguishing between the relevant background knowledge B and irrelevant background knowledge I?*

In the following sections we discuss possible approaches to answering questions 3-5.

3 Search strategies and algorithmic information

3.1 Version spaces

In [14] Mitchell describes a general search strategy for inductive inference, known as the "Version space" approach. This method involves the maintenance of two sets, S and G. These sets represent respectively the least and most general hypotheses which are consistent with the examples so far. The idea is that as increasing numbers of examples are presented the space of plausible hypotheses defined by S and G converges in the limit to a singleton. At this point the system can be said to have recognised the concept.

Might this simple and attractive technique be adapted to the purpose of guiding the search in CIGOL? The answer is "no" for the following reasons. In Mitchell's description, hypotheses are single clauses constructed from a fixed language and containing no terms as arguments other than variables and constants. The generality relationship for Version spaces can be defined by saying that clause C is more general than clause D whenever $C \vdash D$ (see section 2). This generality relationship induces a *finite* lattice over the space of hypotheses, an essential pre-condition for the Version space approach to be effective. CIGOL has a less restricted form of hypothesis language consisting of arbitrary *sets* of Horn clauses. In this case the generality relationship for hypotheses becomes: theory T_1 is more general than theory T_2 whenever $T_1 \vdash T_2$ (section 2). This relationship induces an *infinite* lattice over first order Horn clause theories. Moreover, although top and bottom of the lattice can be defined by theories which are equivalent to the empty clause (the logical constant *false*) and the empty theory (*true*) respectively, according to Plotkin [19] there exist infinite length ascending and descending chains of generality within this lattice. Clearly this indicates that irrespective of considerations of computational efficiency, a Version space search could not be expected to converge within such a lattice. We must therefore look to some alternative model to guide and constrain the search through this more complex lattice.

3.2 Algorithmic information theory

Following the lead of Kolmogorov [8] various information theorists [4, 26, 2] have investigated the relationship between computation, randomness and message complexity. The basic intuition rests on the observation that although the strings

0101001101110011000101101011 00 and
0101010101010101010101010101

have approximately the same Shannon information content [25], the second contains a higher degree of regularity than the first. As an alternative to standard information measures Kolmogorov defined the algorithmic information of a finite string s as being equal to the bit length of the minimal Universal Turing machine program s^* which generates precisely s as output. Thus long regular strings have lower Kolmogorov information than strings of the same length which have no regularity. In addition, Kolmogorov defines a *random* string to be one which cannot be compressed by being encoded as a program for a reference Universal Turing machine.

By definition, inductive construction of first order theories involves a form of information compression. This follows from the fact that most finitely expressible first order theories entail an unbounded set of instances. Moreover, inductive inference from a finite set of examples can never be carried out with absolute confidence. However, we feel increased confidence in hypotheses that cover increasing numbers of examples. We will now attempt to formalise the notion of confidence in hypotheses directly with respect to compression of information.

3.3 CIGOL hypotheses and chance regularity

For the reader's convenience we provide a proof sketch for the following theorem and corollary from algorithmic information theory [4].

Theorem 1 *Let Σ_n be the set of all binary strings of length n, T_r be an arbitrarily chosen reference Turing machine and the k-bit-compressible strings of length n, $K_{n,k}$, be defined as $\{y : y \in \Sigma_n, x \in \Sigma_{n-k}, T_r(x) = y\}$. The set $K_{n,k}$ has at most 2^{n-k}elements.*

Proof Since Turing machines are deterministic T_r either induces a partial one-to-one or many-to-one mapping from the elements of Σ_{n-k} to the elements of $K_{n,k}$. Thus $|K_{n,k}| \leq |\Sigma_{n-k}| = 2^{n-k}$. $\square$

Corollary 2 *The probability of a binary string generated by tossing an unbiased coin being compressible by k bits using any Turing machine T_r as a decoding mechanism is at most 2^{-k}.*

Proof Applying theorem 1, the proportion of randomly generated strings which are compressible by k bits is at most $2^{n-k}/2^n = 2^{-k}$. □

Note that T_r is merely used here for decoding compressed strings. In addition these results hold irrespective of the choice of T_r. As mentioned in the previous section information theory defines the absolute information of a string by making use of the special case in which T_r is a reference Universal Turing Machine. However, this is immaterial to the present discussion since the discovery of such a minimal-length encoding is an undecidable problem [4]. On the other hand, clearly the mere compressibility of a string relative to a particular decoding machine which is known to halt on all inputs *is* decidable.

Now consider again the inductive setting described by

$$I \wedge B \wedge H \vdash E^+$$

Within CIGOL, background knowledge is built up incrementally. Imagine that the theory P is built entirely on the basis of examples. Though some of P will be irrelevant to some examples, we can view P as being a single hypothesis which entails the examples. Thus

$$P \vdash E^+$$

Of course an inductive agent cannot know the origin of the examples E^+. When evaluating the results of experimentation one generally makes use of a null hypothesis, $\overline{\mathcal{H}}$, which is the negative of the hypothesis, $\mathcal{H}$, being tested. By refuting $\overline{\mathcal{H}}$ one demonstrates the plausibility of $\mathcal{H}$. In our setting we might take the null hypothesis to be that every bit in the encoding of the examples E^+ was produced by tossing a coin. Note that $\mathcal{H}$ is an hypothesis about an hypothesis. We can find an upper bound on the probability of the null hypothesis using corollary 2 by defining a reference Turing Machine T_r which, given an encoded version of P as input generates an encoded version of E^+ as output. Thus

$$T_r(I(P)) = O(E^+) \tag{2}$$

where $I(P)$ is an input tape encoding of P, $O(E^+)$ is an output tape encoding of E^+ and $I(P)$ is k bits shorter than $O(E^+)$. The machine T_r will be described in the next section. We will use X_k to denote the statement

that there exists such a k-bit compressed explanation $I(P)$ of $O(E^+)$. Now according to corollary 2

$$Pr(X_k|\overline{\mathcal{H}}) \leq 2^{-k}$$

We will use the probability of $\overline{X_k}$ given that the null hypothesis was true as a measure of our confidence that the compression produced by accepting P is not based on the discovery of chance regularities within E^+. This is

$$\begin{aligned} Pr(\overline{X_k}|\overline{\mathcal{H}}) &= 1 - p(X_k|\overline{\mathcal{H}}) \\ &\geq 1 - 2^{-k} \end{aligned} \tag{3}$$

Example 1 *Let $O(E^+)$ be 110 bits long and $I(P)$ be 100 bits long. Then*

$$\begin{aligned} Pr(\overline{X_k}|\overline{\mathcal{H}}) &\geq 1 - 2^{100-110} \\ &\geq 1 - 1/1024 \\ &\geq 0.999 \end{aligned}$$

Note the following intuitively appealing features of (3) as a measure of hypothesis confidence. Firstly, $Pr(\overline{X_k}|\overline{\mathcal{H}})$ is only well defined as a probability when k is positive, i.e. we can only have confidence in a theory which is less bulky than the facts on which it is based. Secondly, for all finite values of k $Pr(\overline{X_k}|\overline{\mathcal{H}})$ is less than 1, i.e. no matter how many facts are covered by a theory, we can never have total confidence in it. Thirdly, an increase in k when it is already large provides only a small increase in $Pr(\overline{X_k}|\overline{\mathcal{H}})$, i.e. there are diminishing returns in the confidence inspired in a theory involved in showing that it covers an increasingly large number of facts. All of these are standard assumptions within the philosophy of science [20].

It is now necessary to describe in more detail the reference Turing machine T_r and its input and output tape encodings I and O.

3.4 Encodings and the compression model

First an efficient Turing tape encoding M for any logical expression, S is described. The encoding M should be efficient in the sense that almost any tape encoding should correspond to a particular logical expression and vice versa. This is necessary for testing $\overline{\mathcal{H}}$ since we do not want to introduce the possibility of spurious compressibility due to inefficient encoding of the examples.

A set of Prolog clauses can be coded as a single logical expression, using list concatenation symbols as necessary to separate clauses. One such coding

might be to use a standard prefix coding [1], such as Huffman codes, for coding each function symbol and variable, and write the expression on the tape using reverse polish notation. Reverse polish allows us to ignore the requirement for bracketting and separators.

Let $sym(S)$ represent the combined set of variables and function symbols of given arity within S and let N be the sum of the frequencies of occurrence of elements of $sym(S)$ within S. Now if we write the relative frequency of occurrence of symbol s in S as p_s then, ignoring the length of the prefix table, $M(S)$ has a length of

$$|M(S)| \approx N \sum_{s \in sym(S)} -p_s \log_2 p_s \text{ bits}$$

according to Shannon information theory. There is obviously a similarity here to the entropy function used in ID3 [21], which should not be surprising given the common basis in information theory.

Example 2 *Let S be [crow(harry), (black(X) :- crow(X))]. Then $sym(S)$ is {':-'/2, '.'/2, crow/1, black/1, X/0, harry/0, []/0}, $N = 10$ and the corresponding relative frequencies are <0.1, 0.2, 0.2, 0.1, 0.2, 0.1, 0.1>. Thus $|M(S)| \approx 10(4 \times 0.1 \times 3.3 + 3 \times 0.2 \times 2.3) \approx 27$ bits.*

Clearly we can use M as the output tape encoding function O described in the previous section, rewriting (2) as

$$T_r(I(P)) = M(E^+)$$

Could we also use M as the input tape encoding function I? In fact we cannot, since the logic program P does not contain sufficient information to describe which particular instances it was derived from. Thus in general there is no Turing machine which could take an encoding of an arbitrary logic program P and print out the set of instances E^+ from which P was derived. This leaves us in a predicament as to how to represent this additional information for the purposes of our model of hypothesis confidence.

The following is a possible solution. Devise a numbering scheme for all instances entailed by P. Now append the numbers corresponding to the particular examples in E^+ onto the encoded description of P to make $I(P)$. Such a numbering scheme is called a Gödel numbering after [7]. A natural

[1] Prefix codes are variable length bit patterns used to encode the symbols in a message. Efficient coding schemes allow one to encode each symbol in close to the optimal of $-\log_2 p$ bits per symbol, where p is the relative frequency of the symbol within the message.

numbering scheme that suggests itself is to consecutively number the unit clauses in order of their first appearance within the levelwise expansion of the resolution universe, $R^*(P)$ [23]. The levels of $R^*(P)$ are defined as follows

$$\begin{aligned} R^0(P) &= P \\ R^n(P) &= R^{n-1}(P) \cup \{C : C_1, C_2 \in R^{n-1}(P), \\ &\qquad C \text{ is the resolvent of } C_1 \text{ and } C_2\} \end{aligned}$$

$R^*(P)$ is simply the closure $R^0(P) \cup R^1(P) \ldots$ We will use $unit(i, P)$ to denote the ith unit clause i in this enumeration. However, this numbering scheme is still not adequate for the purpose since $R^*(P)$ does not contain all the *ground* unit clauses entailed by P. However, it is straightforward to show that for every ground unit clause L entailed by P there is a unit clause L' in $R^*(P)$ and a substitution θ such that $L = L'\theta$. Thus our input tape encoding I can simply be as follows

$$I(P) = M(< P, u_1, \theta_1, \ldots, u_n, \theta_n >)$$

where $unit(u_i, P)\theta_i$ is the ith example from E^+, and M is used to encode the entire expression $< P, u_1, \ldots >$.

We are now in a position to calculate the lower bound on hypothesis confidence expressed in inequality (3) of the previous section.

Example 3 *Let E^+ = {crow(tom), crow(dick), crow(harry), crow(janis)} and P = {crow(X)}. Now $I(P) = M(<$[crow(X)], 0, (X/tom), 0, (X/dick), 0, (X/harry), 0, (X/janis)$>)$. Computing encoding lengths in the fashion demonstrated in example 2 we find that $|M(E^+)| \approx 32$ bits while $|I(P)| \approx 65$ bits. Thus since the encoding for the hypothesis is longer than the encoding of all of the examples we can attribute no confidence to the hypothesis.*

Example 4 *Let E^+ = a(b(c(d(e(f(g(h)))))))), a(b(c(d(e(f(g(i)))))))), a(b(c(d(e(f(g(j))))))) and P = a(b(c(d(e(f(g(X)))))))). In this case $|M(E^+)|$ = 96 bits and $|I(P)|$ = 73 bits. Substituting these values into (3) produces a confidence of at least $1 - 2^{-23} = 0.9999999$, i.e. virtual certainty that the compression produced by accepting the hypothesis is not accidental.*

Examples 3 and 4 demonstrate an important point regarding the truncation operator within CIGOL. In the version of CIGOL described in [17] hypotheses were preferred entirely on the basis of textual simplicity. As a result, the learning sessions demonstrated that CIGOL had a strong tendency to overgeneralise when applying the truncation operator. Since this

operator merely replaces terms by variables, the most preferred description is bound to be the most general, such as $member(X, Y)$. However, since simplicity seemed to be a powerful heuristic for all other cases, it seemed arbitrary to use a different heuristic for this special case. As examples 3 and 4 demonstrate, the confidence bound described by (3) produces a much more satisfactory result, and clearly distinguishes between generalisations based on weak and strong similarities respectively.

In summary, the reader should note that the new strategy is based on preferring inverse resolution operators on the basis of their abilities to shorten the minimal acehivable encoding of the examples.

4 Example sessions

Francis Bacon's *Novum Organum* (1620) is an early but thorough exposition of what is now known as scientific method. As an example of the method of hypothesis formation Bacon demonstrates that many of the properties of light can be inferred from a small set of known facts. In this section we demonstrate the performance of the revised CIGOL which employs the "confidence" statistic developed in previous sections of this paper. The example below is in the spirit of Bacon's exposition of the properties of light. User input is underlined, and excessive computer output with no corresponding input from the user is replaced by "...".

```
!- [-inverse].
!- show_clauses.
 inverse(huge,tiny).
 inverse(large,small).
 inverse(small,large).
 inverse(tiny,huge).
 proportional(huge,huge).
 proportional(large,large).
 proportional(small,small).
 proportional(tiny,tiny).
!- situation(dist(light,board,tiny),
  illum(board,huge)).
 Confidence = NIL for (situation(dist(light,
  board,tiny),illum(board,A)):-inverse(A,tiny))
 Confidence = NIL for (situation(dist(light,
  board,A),illum(board,B)):-inverse(B,A))
```

```
   ...
!-
```

In the session so far the user started by loading background knowledge concerning the qualitative relations *inverse* and *proportional*. The user now describes an observed situation, *situation(dist(light,board,tiny), illum(board,huge))*, involving a board and a light source. In the situation described, the light source is a tiny distance from the board, and the illumination on the board is huge. CIGOL tries various hypotheses, including a possible inverse relationship between distance and illumination. However, with only a single example on which to base the hypothesis CIGOL has insufficient confidence to suggest the hypothesis to the user. In the version of CIGOL described in [17] CIGOL would at this point have chosen the hypothesis *(situation(dist(light,board,A),illum(board,B)):-inverse(B,A))* to present to the user since this is the simplest within the hypothesis space. Using the new confidence statistic, CIGOL acts more cautiously since this hypothesis, although simple, is no simpler than the presented example itself. We now continue the session.

```
!- situation(dist(light,board,large),
   illum(board,small)).
 TRUNCATION: (0.999)
 Is situation(dist(light,board,A),
  illum(board,B)) always true?  n.
  ...
 ABSORPTION: (0.99999)
  New clauses:[(situation(dist(light,board,A),
   illum(board,B)):-inverse(A,B))]
  cover new facts:  [situation(dist(light,board,
   huge),illum(board,tiny)),situation(dist(light,
   board,small),illum(board,large)),...]
 Are new clauses always true?  y.
!-
```

Given the additional example, CIGOL finds a high confidence level (0.999) for applying "truncation" since the two examples share a lot of structure (see examples 3 and 4). However, the user rejects this and is instead presented, by absorption, with the previously rejected hypothesis stating the inverse distance relation. Note that this hypothesis, though textually more complex than the truncation leads to an even higher confidence level using the new

model. The reason for this is that no additional substitutions need be stored to describe the Gödel numbers of an absorption or intra-construction. Note also that the background knowledge concerning the qualitative *proportional* relationship was never proposed in any hypothesis. The reason again for this is that the new confidence statistic always rejects hypotheses unless they simplify the description. This provides something of an answer to question 5 in section 2 of this paper. It should be noted that the limitations of efficient search using a best first algorithm can lead to the normal problems to do with local minima.

The inverse distance hypothesis is accepted by the user. Interestingly, Bacon proposed this qualitative relation for light from simple observations more than 60 years before Newton's quantitative inverse square distance law for gravitation.

Next the user provides examples concerning the illumination on an opaque globe, resulting in the additional inverse distance law *situation(dist(light,globe,A),heat(globe,B)) :- inverse(A,B)*. CIGOL then combines these two laws by constructing a new predicate as follows.

```
INTRA-CONSTRUCTION (0.99997)
 situation(dist(light,A,B),heat(A,C)):-
  p559(A),inverse(B,C).
 p559(board).
 p559(globe).
What shall I call p559?  opaque.
!-
```

Thus CIGOL discovers a set of objects which reflect, and the user names the new concept "opaque".

The user next goes through the same process with respect to the heat properties of the light source for the given objects, eventually producing the similar rule *situation(dist(light,A,B),heat(A,C)):- opaque(A), inverse(B,C)*, i.e. opaque objects get hot when close to the light source, and are cooler when the light source is moved away. CIGOL now combines the two analogous rules together to construct a new predicate again as follows.

```
INTRA-CONSTRUCTION (0.99999)
 situation(dist(light,A,B),C):-
  p773(A,C,D),opaque(A),inverse(B,D).
 p773(A,heat(A,B),B).
 p773(A,illum(A,B),B).
```

```
 What shall I call p773?  radiation.
!- show_clauses
 inverse(huge,tiny).
 inverse(large,small).
 inverse(small,large).
 inverse(tiny,huge).
 proportional(huge,huge).
 proportional(large,large).
 proportional(small,small).
 proportional(tiny,tiny).
 opaque(board).
 opaque(globe).
 situation(dist(light,A,B),C):-
  radiation(A,C,D),opaque(A),inverse(B,D).
 radiation(A,heat(A,B),B).
 radiation(A,illum(A,B),B).
```

CIGOL has managed to combine the concepts of light and heat to produce a new 3-place relation which the user calls *radiation*. The three arguments of radiation correspond respectively to the *transmitter*, *radiation-type* and *receiver*. Note that this predicate construction is rather like a second order *analogy*, even though CIGOL works only in first order logic. The reason CIGOL managed to carry out a pseudo-second-order analogy is because the properties *heat* and *illumination* were described within the examples using function symbols rather than predicate symbols.

At the send of the session the user types *show_clauses* to reveal the entire set of clauses.

5 Discussion

Machine invention of concepts within unrestricted first order Horn clause logic is at least as ambitious as most other problems within Artificial Intelligence. For this reason progress is likely to be slow. However, the method of inverting resolution described in [17] provides a logical basis for a very general form of inductive inference. This paper describes an information theoretic approach to constraining the hypothesis space for inverse resolution, and an attempt is made to integrate the logical, computational and information-based aspects of hypothesis formation.

Many obstacles lie ahead in the further development of the inverse resolution approach to machine learning. These include

1. **Noise.** CIGOL works on the unacceptable assumption of completely noise-free data.

2. **Time.** CIGOL does not take into account the time complexity of executing the hypotheses which it forms.

3. **Unrestricted operators.** The derivation of the inverse resolution operators in [17] used a number of assumptions to simplify the derivation. These assumptions restrict the CIGOL operators unduly.

The work described in sections 3.3 and 3.4 of this paper may give us some lead on the problem of noise. This comes from the definition within algorithmic information theory of random (incompressible) strings. Indeed incompressibility is one of the most ubiquitous features that distinguises noise from signal. Thus the incompressibility of sections of a set of examples would be a strong indication that the corresponding examples are noisy. It remains to be seen whether the relationship between noise and compressibility might be put to use within a system such as CIGOL.

References

[1] R.B. Banerji. Learning in the limit in a growing language. In *IJCAI-87*, pages 280–282, Los Angeles, CA, 1987. Kaufmann.

[2] C. Bennett. Logical depth and physical complexity. In R. Herken, editor, *The Universal Turing Machine A Half Century Survey*, pages 227–257. Kammerer and Unverzagt, Hamburg, 1988.

[3] W. Buntine. Generalised subsumption and its applications to induction and redundancy. *Artificial Intelligence*, 36(2):149–176, 1988.

[4] G. Chaitin. *Information, Randomness and Incompleteness - Papers on Algorithmic Information Theory.* World Scientific Press, Singapore, 1987.

[5] G. DeJong. Generalisations based on explanations. In *IJCAI-81*, pages 67–69. Kaufmann, 1981.

[6] S.L. Epstein. On the discovery of mathematical theorems. In *IJCAI-87*, pages 194–197, Los Angeles, CA, 1987. Kaufmann.

[7] K. Gödel. *On Formally Undecidable Propositions of Principia Mathematica and Related Systems.* Oliver and Boyd, London, 1962.

[8] A.N. Kolmogorov. Three approaches to the quantitative definition of information. *Prob. Inf. Trans.*, 1:1–7, 1965.

[9] P. Langley, G.L Bradshaw, and H. Simon. Rediscovering chemistry with the Bacon system. In R. Michalski, J. Carbonnel, and T. Mitchell, editors, *Machine Learning: An Artificial Intelligence Approach*, pages 307–330. Tioga, Palo Alto, CA, 1983.

[10] D.B. Lenat. On automated scientific theory formation: a case study using the AM program. In J.E. Hayes and D. Michie, editors, *Machine Intelligence 9.* Horwood, New York, 1981.

[11] R. Michalski and J. Larson. Selection of most representative training examples and incremental generation of vl1 hypotheses: the underlying methodology and the description of programs ESEL and AQ11. UIUCDCS-R 78-867, Computer Science Department, Univ. of Illinois at Urbana-Champaign, 1978.

[12] R.S. Michalski. A theory and methodology of inductive learning. In R. Michalski, J. Carbonnel, and T. Mitchell, editors, *Machine Learning: An Artificial Intelligence Approach*, pages 83–134. Tioga, Palo Alto, CA, 1983.

[13] R.S. Michalski and R. Stepp. Learning from observation: conceptual clustering. In R. Michalski, J. Carbonnel, and T. Mitchell, editors, *Machine Learning: An Artificial Intelligence Approach*, pages 331–364. Tioga, Palo Alto, CA, 1983.

[14] T.M. Mitchell. Generalisation as search. *Artificial Intelligence*, 18:203–226, 1982.

[15] T.M. Mitchell, R.M. Keller, and S.T. Kedar-Cabelli. Explanation-based generalization: A unifying view. *Machine Learning*, 1(1):47–80, 1986.

[16] S.H. Muggleton. Duce, an oracle based approach to constructive induction. In *IJCAI-87*, pages 287–292. Kaufmann, 1987.

[17] S.H. Muggleton and W. Buntine. Machine invention of first-order predicates by inverting resolution. In S.H. Muggleton, editor, *Inductive Logic Programming*. Academic Press, London, 1992.

[18] T. Niblett. A study of generalisation in logic programs. In *EWSL-88*, London, 1988. Pitman.

[19] G.D. Plotkin. *Automatic Methods of Inductive Inference*. PhD thesis, Edinburgh University, August 1971.

[20] K. Popper. *Conjectures and Refutations: The Growth of Scientific Knowledge*. Routledge and Kegan Paul, London, 1972.

[21] J.R. Quinlan. Discovering rules from large collections of examples: a case study. In D. Michie, editor, *Expert Systems in the Micro-electronic Age*, pages 168–201. Edinburgh University Press, Edinburgh, 1979.

[22] L. Rendell. Substantial constructive induction using layered information compression: tractable feature formation in search. In *IJCAI-85*, pages 650–658. Kaufmann, 1985.

[23] J.A. Robinson. A machine-oriented logic based on the resolution principle. *JACM*, 12(1):23–41, January 1965.

[24] C. Sammut and R.B Banerji. Learning concepts by asking questions. In R. Michalski, J. Carbonnel, and T. Mitchell, editors, *Machine Learning: An Artificial Intelligence Approach. Vol. 2*, pages 167–192. Kaufmann, Los Altos, CA, 1986.

[25] C.E. Shannon and W. Weaver. *The Mathematical Theory of Communication*. University of Illinois Press, Urbana, 1963.

[26] R.J. Solomonoff. A formal theory of inductive inference. *J. Comput. Sys.*, 7:376–388, 1964.

[27] R. Wirth. Learning by failure to prove. In *EWSL-88*, pages 237–251, London, 1988. Pitman.

[28] S. Wrobel. Automatic representation adjustment in an observational discovery system. In *EWSL-88*, pages 253–262, London, 1988. Pitman.