Pre-publication draft of paper accepted for publication in the Proceedings of the IJCAI-93 Workshop on Machine Learning and Knowledge Acquisition, pp 236 -252

DLGref2: Techniques for Inductive Knowledge Refinement

Geoffrey I. Webb
School of Computing & Mathematics, Deakin University Geelong, Vic. 3217, Australia

Abstract

This paper describes and evaluates machine learning techniques for knowledge-base refinement. These techniques are central to Einstein, a knowledge acquisition system that enables a human expert to collaborate with a machine learning system at all stages of the knowledge-acquisition cycle. Experimental evaluation demonstrates that the knowledge-base refinement techniques are able to significantly increase the accuracy of nontrivial expert systems in a wide variety of domains.

Keywords

Knowledge-base refinement; Data-driven machine learning; Induction

1 Introduction

Einstein is a knowledge acquisition system that enables a human expert to collaborate with a machine learning system at all stages of the knowledge acquisition and refinement cycle. Both the human expert and the machine learning system can suggest modifications and/or critique the knowledge-base at any stage of development. The manner in which this collaboration is managed has been described in detail elsewhere (Webb, 1992*a*, 1993*a*). This paper describes the techniques used by the machine learning subsystem to refine a knowledge-base.

A number of factors distinguish the demands placed upon Einstein's inductive refinement sub-system from those placed upon previous inductive refinement programs.

One of the core design decisions that lie behind Einstein is that it should be easy for the human user to use. To this end, it employs a restricted form of production rule-based knowledge-base. The conclusion of a rule is restricted to a single categorical assertion (also known as a classification). There are no intermediate reasoning steps, so the conclusion of one rule may not appear in the condition of another. Each rule can be interpreted in isolation. That is, in order to determine whether a rule will be fired for a case it is not necessary to refer to any other rule.

Aside from being forced to work within the constraints placed upon it by the form of knowledge-representation that is employed, because the refinement sub-system is modifying rules created by the human expert, it is essential that it minimises the degree of change wrought when refining rules. In general, a user would find it extremely frustrating to have key aspects of his or her input to the knowledge-base expunged by the refinement sub-system without extremely good cause.

A further requirement is that the refinement sub-system be able to work with real world data which may be incomplete and/or inaccurate.

There are numerous previous techniques for inductive knowledge-base refinement. While all offer significant facilities, none solves all knowledge refinement problems. Some are able to refine or delete existing rules but unable to add new rules (Ginsberg, 1988; Ginsberg, Weiss & Politakis, 1988; Wilkins and Buchanan, 1986; Ma and Wilkins, 1991; Rada, 1985; Caruana, 1989; Quinlan, 1987). Others examine only a single example case at a time and thus are not able to take advantage of a machine learning system's capacity for detailed analysis of multiple cases (Davis & Lenat, 1982; Smith, Winston, Mitchell & Buchanan, 1985). Several systems utilise the initial knowledge-base when developing the new, but do not explicitly constrain the degree of change that may be wrought upon that knowledge-base when developing the refined version (Reinke & Michalski, 1988; Pazzani & Brunk, 1991; Lee & Ray, 1986). Ourston & Mooney's (1990) system is unable to accommodate inaccurate data. Reinke & Michalski's (1988) approach requires that any specialisation of a rule should cover all positive cases covered by the original rule. This can result in needless complexity in the final knowledge-base.

2 DLGref2

DLGref2, and its precursor, DLGref (Webb, l992*c*) are variants of the DLG (Webb, 1991, l992*b*; Webb & Agar, 1992) data-driven machine learning algorithm. DLG differs from most previous data-driven machine learning algorithms by the use of least generalisation to develop successive rules of a rule set. Its core operations are very similar to those of GOLEM (Muggleton & Feng, 1990) which was developed simultaneously and independently.

DLGref is unable to accommodate noisy data and cannot specialise the range of values covered by a clause in a rule's condition. DLGref2 extends DLGref to handle these two cases.

DLGref2 is designed to operate on the types of production rules that are required by Einstein. Each rule is restricted to a single categorical conclusion. That is, each conclusion must assign a single category (called a class) to a case. Further, all classes in a single knowledge base must be mutually exclusive. Finally, the rule base must be flat. That is, it is not possible to use the conclusion of one rule in the condition of another. However, while DLGref2 has been designed and evaluated within this restricted context, it should scale up to more complex knowledge representation schemes, and, indeed, current work is investigating exactly this issue.

DLGref2 is applied to a knowledge-base once for each class. Only rules for that class are considered during a single application. During such an application, all example cases that belong to that class are considered to be positive examples and all cases that do not belong to the class are considered to be negative examples. If the condition for a rule is satisfied with respect to a case then the rule is said to cover that case.

First, all cases covered by rules that do not misclassify cases are removed. This prevents other rules from being needlessly generalised to cover those cases.

Next, all rules that misclassify cases are examined in turn.

1. The DLG induction algorithm is applied using all negative cases but only the positive cases that are covered by the rule. This process develops a rule, *spec_rule,* that covers as many as possible of the positive cases. Depending

upon the value function used with DLG, either it will not be possible for *spec_rule* to cover any negative cases, or there will be a trade-off between the number of negative and positive cases covered. *Spec_rule* will always be a specialisation of the rule being refined *(rule).* When developing *spec_rule,* the first case considered is always the most central (or typical case) and subsequent cases are examined in order of extremity (or atypicallity).

2. next, a new rule, *n,* is created such that –

   - *n* is a generalisation of *spec_rule;*
   - *n* is a specialisation of *rule,*
   - *n* covers no more negative cases than *spec_rule;* and
   - there is no generalisation of *n* that is a specialisation of *rule* and which covers no more negative cases than *spec_rule.*

   The technique used to select this rule is a variant of version space narrowing (Webb, 1993*b*).

3. As each rule is developed, the positive cases that it covers are removed. This prevents other rules from being needlessly generalised to cover those cases.

After all existing rules have been processed in this manner, they are each generalised in turn to cover further positive cases. After each such generalisation is performed, positive cases are once more removed.

Finally, if any positive cases remain that are not covered by the amended rules, new rules are developed, using the DLG algorithm.

A formal description of the above algorithm is provided in Appendix A.

An important feature of this algorithm is the use of a centrality or typicality measure in step 1 of the algorithm. The use of a most typical initial case leads to the development of a rule that covers as many as possible of the most typical cases covered by *rule* while the subsequent examination of most extreme cases increases the probability maximising the number of cases covered when allowing for the possibility of noisy data.

This process is illustrated in Figure 1. In this Figure, example cases are presented as points in a two dimensional space representing a case's age and height. Positive cases are represented by uppercase letters. Negative cases are represented by lowercase letters. Rules take the form `IF` $a \leq$ `AGE` $\leq b$ `AND` $c \leq$ `HEIGHT` $\leq d$ `THEN POSITIVE`. Thus, each rule can be viewed as defining a rectangle that includes those points that the rule assigns to the positive class. The thick outer line indicates the rule to be refined (IF $1 \leq$ AGE ≤ 4 AND $0.5 \leq$ HEIGHT ≤ 4.5 THEN POS1TIVE). It covers five positive and three negative cases. Assuming that the rule evaluation function favours rules that reach only correct conclusions over those that do not, and prefers rules that reach more correct conclusions over those that reach fewer correct conclusions, the best specialisation of the initial rule will cover A, B, C and E (IF $1.5 \leq$ AGE ≤ 4 AND $1 \leq$ HEIGHT ≤ 3.5 THEN POSITIVE). Examining successive least generalisations, if D is incorporated in the rule, it will not be possible to obtain the best rule. The best obtainable specialisation of the initial rule that covers D covers

just A, D and
E (IF $3 \leq AGE \leq 4$ AND $2 \leq HEIGHT \leq 4$ THEN POSITIVE).

DLGref2 selects the most central positive case covered by the initial rule, E, and forms the most specialised rule that covers that case (IF $3 \leq AGE \leq 3$ AND $2.5 \leq HEIGHT \leq 2.5$ THEN POSITIVE). It then generalises against the most extreme positive case, B. This results in (IF $1.5 \leq AGE \leq 3$ AND $l \leq HEIGNT \leq 2.5$ THEN POSITIVE). The next most extreme case is D, but an attempt to generalise against D is blocked because the resulting rule (IF $1.5 \leq AGE \leq 4$ AND $1 \leq HEIGHT \leq 4$ THEN POSITIVE) covers two negative cases. Continuing the process by generalising against C and A in turn, results in the desired rule. This process does not guarantee the selection of the best rule. However, starting from the most typical initial case does guarantee that the result will be, in some sense, typical of the initial rule, while generalising against successive most extreme cases has demonstrated the best results in extensive experimental evaluation.

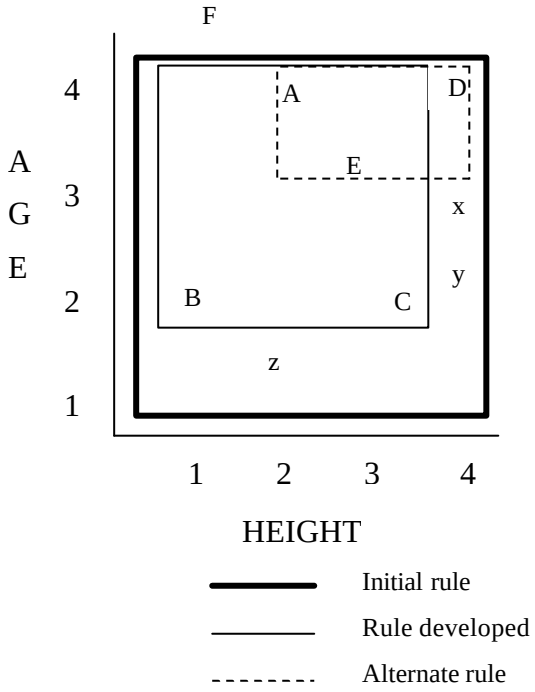

Figure 1: Illustration of use of centrality measure

After the replacement specialisation has been developed it will be further generalised to cover other positive cases, such as F, that are not covered by the initial rule. D will be covered by the refinement of some other initial rule, or by a new rule developed after all initial rules have been refined.

The use of a centrality measure to order cases during induction has been examined in detail by Webb (1992*e*).

Note that, while the above example and the evaluation to follow, examine only rules with simple attribute value tests, the use of least generalisation within DLGref2 ensures that it can be applied to much more complex forms of rule, as is demonstrated by the use of least generalisation to infer rules in first order predicate logic (Muggleton & Feng, 1990).

Benefits offered by DLGret2 include:

- rules that are consistent with the training set are generalised the least possible amount consistent with ensuring that all cases are covered by at least one rule;
- rules that are inconsistent with the training set and which cover one or more positive cases are modified so as to
  - minimise the change to the original statement of the rule
  - maximise the number of positive cases covered by the original rule, and, with lower preference, the number of positive cases covered by no other rule, that are covered by the new rule; and
  - minimise the number of negative cases that are covered by the new rule;
- rules that are inconsistent with the training set and which cover no positive cases are not altered on the assumption that the training set simply does not address the issues addressed by the rule;
- new rules are developed that cover any cases not covered by revised versions of initial rules; and
- the manner in which the algorithm accommodates noise, by trading-off the levels of positive and negative cover, can be altered by changing a rule evaluation function.

DLGref (the precursor to DLGref2) was evaluated by

- refining a set of rules developed by C4 rules (Quinlan, 1987) for the Hypothyroid domain
- refining a set of rules developed by DLG from one set of data, using another set of data; and
- refining rules that performed correctly for a single class in order to accommodate all classes.

In almost all cases DLGref was able to create refined rules that performed significantly better both than the unrefined rule set and than the rules developed by DLG (DLGref without access to the initial knowledge-base) alone (Webb, 1992*c*).

However, most of these studies employed initial rule sets created by DLG. DLG has been developed with a view to creating rule sets that are easy to comprehend. In consequence, each rule is intended to be as modular as possible; to be interpretable in isolation without reference to the rest of the rule set. In consequence, each rule can be readily revised in relative isolation.

Unfortunately, rule sets are often not so modular. For example, if there is a suitable conflict resolution strategy, rules will often be over-generalised on the grounds that the rule will not fire in inappropriate contexts due to higher priority rules firing. DLGref (and DLGref2) do not take account of conflict resolution strategies, and thus specialise such rules so that they stand as justifiable in isolation from other rules in the rule set. This makes the task of the refinement algorithm especially difficult for two reasons.

1. key information from the initial knowledge base (the inter-relationships between rules) is not considered during refinement; and
2. the refined knowledge base must incorporate additional complexity in order to distinguish all rules from negative cases. This must increase the possibility of incorporating inappropriate conditions, solely from the point of view that there are more conditions each of which may be inappropriate;

As it is intended that Einstein should be applicable to the refinement of knowledge-bases developed in different environments, it is important that its inductive refinement subsystem should be able to cope with rules that are optimised with respect to a conflict resolution strategy. Further, the human partner in the knowledge-acquisition process is encouraged to specify partial or incomplete rules when he or she has some insight into how to solve a particular problem but is unable to articulate a fully operational solution. When examined without consideration of the conflict resolution strategy for which they were developed, rules optimised with respect to such a strategy are equivalent to partial or incomplete knowledge. It is essential that Einstein's knowledge-base refinement sub-system be capable of adequately handling such rules.

3 Experimental Evaluation

Experimental evaluation of DLGref2 has been designed to explore how well it performs when refining rule sets in which the rules have been optimised to take account of a conflict resolution strategy (and thus constitute partial or incomplete knowledge when considered outside that context).

To this end, DLGref2 has been evaluated on its performance when refining rules developed by C4.5rules (Quinlan, 1992) against a wide variety of data sets. C4.5rules highly optimises its rule sets with regard to the conflict resolution strategy employed. It also treats missing values in a different manner to the current implementation of DLGref2. Whereas the current implementation of DLGref2 treats a missing value as a distinct value, C4.5rules does not (it considers that rules fail if a value referred to in the condition is missing from a case). Thus, the rules produced by C4.5rules constitute partial knowledge when interpreted in the context of the conflict resolution strategies assumed by DLGref2.

This is also a particularly difficult refinement task because C4.5rules is a sophisticated induction system renown for the high accuracy of the rules that it induces. DLGref2 is being asked to use induction to improve the output of one of the leading induction systems.

The algorithms were evaluated using a simple production rule language. The condition for a rule was restricted to a conjunction of clauses. Each clause related to a single attribute. For categorical attributes, a clause consisted of a set membership, such as *breast_quad* $\in$ *{unknown, upper-left, lower-left, central}.* For ordinal attributes, a clause consisted of one of the forms, *value is unknown; value* $\geq c$*; value* $\leq c$; *value is unknown* $\vee$ *value* $\geq c$*;* or *value is unknown* $\vee$ *value* $\leq c$, where c is a constant. Examples of clauses for a nominal attribute include *age* ≤ 20*, age* ≥ 20*, age is unknown, age is unknown* $\vee$ *age* ≤ 20 and *age is missing* $\vee$ *age* ≥ 20*.* Although the techniques are not restricted to such a language, the current software is.

DLGref2 can be customised to a task by altering the rule evaluation function that is employed. The rule evaluation function is used to compare rules during the system's

inductive search. Two rule evaluation functions were employed, *binomial* with upper error limit of 0.4 and *complete_and_consistent*.

The binomial value function can be described by:

$$\text{value}(r) = \frac{\text{poscover}(r) - 0.5 - \text{cover}(r)(1 - u)}{\sqrt{\text{cover}(r)u(1 - u)}}$$

where poscover(*r*) is the number of positive cases covered by *r*, cover(*r*) is the total number of cases covered by *r* and *u* is the upper error limit (a number between 0 and 1).

The binomial function approximates an evaluation of the level of evidence that, were the rule applied to the population from which the training set was drawn, it would misclassify less than the upper error limit of cases. The higher the value of the function, the greater the level of evidence.

The *complete_and_consistent* value function assigns -1 to any description that covers any number of negative cases. If a description does not cover any negative cases its value is set to the number of positive cases that it covers.

Use of the complete and consistent value function results in the development of rule sets that are complete and consistent with regard to the training set, where this is possible. Use of the binomial value function enables the development of rules that are incomplete or inconsistent with the training set, by seeking to develop rules that maximise the evidence that there is less than the specified level of noise in the training set. The use of each of these value functions with the DLG algorithm has been evaluated in detail elsewhere (Webb, 1992*d*).

In order to apply a set of production rules it is necessary to define a rule interpreter. When applying a rule set to a case, all rules were examined to determine whether their conditions were satisfied. Where the conditions of multiple rules were satisfied, the conclusion of the rule that covered the most cases from the training set was fired. Where no rule's condition was satisfied, the most common class from the training set was assigned to the class. This interpreter is equivalent in effect to the interpreter for which C4.5rules optimises its rules.

The conflict resolution strategy employed by this interpreter was not considered during DLGref2's rule induction. Thus, although no conflict resolution strategy was considered during induction, C4.5rule's strategy was employed during rule application, further increasing the difficulty of DLGref2 producing rules that outperform C4.5rule's rules.

DLGref2 was evaluated by application to ten machine learning data sets from the UCI repository of machine learning data sets (Murphy & Aha, 1992). For all of these data sets, the cases are divided into a number of mutually exclusive classes. The induction task is to develop an expert system that can classify a case by reference to the values of its attributes. These data sets are described in Table 1. The first column of Table 1 presents the number of attributes by which each case is described. The second column presents the percentage of these attributes for which the values are ordinal. The third column presents the percentage of attribute values which are missing from the data. The fourth column presents the number of cases in the data set. The fifth column presents the percentage of these cases that belong to the class which is represented by the most cases in the data set. The sixth column presents the percentage of cases for

which there is another case that is identical in all respects except that it belongs to another class. Where this value is not zero it is not possible to develop complete and consistent classifiers with respect to the training set due either to noise or the lack of necessary attributes. The last column presents the number of classes in the data set.

For each test, the data set was randomly divided into three subsets, training set 1 *(45%)*, training set 2 (45%) and the evaluation set (10%). C4.5rules was applied to training set 1 to create an expert system. DLGref2 was then applied to refine this expert system against training set 2 twice, once with each value function. DLG (DLGref2 with no initial expert system) was also applied to training set 2 once, with each value function, in order to determine the ability of the machine learning system to develop rules from the data unaided. All five expert systems (those developed by C4.5rules, DLG with each value function and DLGref2 with each value function) were then evaluated by application to the evaluation set. One hundred such tests, each time using a different random division of the data into training and evaluation sets, were conducted for each set of data.

Domain	No of Attributes	Ordinal %	Missing %	No of cases	Most common class %	Indist-inguish-able %	No of classes
breast cancer	9	0	3	286	70	5	2
echocardiogram	6	83	3	74	68	0	2
glass type	9	100	0	214	40	0	3
hepatitis	19	32	6	155	79	0	2
house votes 84	16	0	0	435	61	0	2
hypothyroid	29	24	6	3772	92	0	4
iris	4	100	0	150	33	0	3
lymphography	18	0	0	148	55	0	4
F11 multiplexer	11	0	0	500	50	0	2
primary tumor	17	0	4	339	25	18	22

Table 1: UCI data sets

Table 2 presents the results of these experiments. For each data set, the mean accuracies obtained by DLGref 2 (complete and consistent), C4.5rules, DLG (complete and consistent), DLGref2 (binomial), C4.5rules and DLG (binomial) are provided. Following each of the mean accuracies for C4.5rules and DLG, is the *p* obtained by a one tailed matched pairs t-test comparing the respective accuracy with that obtained by DLGref2. Where DLGref2 outperformed the other algorithm, this represents the probability that the mean performance of the algorithms over an infinite number of tests would be identical. Where *p* is less than or equal to 0.05, the difference in performance is statistically significant at the 0.05 level. For ease of identification, *p* values indicating statistically significant improvements in performance resulting from the use of DLGref2 are presented in bold type and those indicating statistically significant decreases in performance are underlined.

Treating the experiment as consisting of twenty different treatments (ten data sets by two evaluation functions) the following results are apparent. In no case does DLGref2 lead to a decrease in accuracy to those of both C4.5rules and DLG. In eleven cases, the use of DLGref2 results in an increase in performance over both those obtained through C4.5rules and DLG alone. In six cases, the use of DLGref2 leads to a statistically significant improvement in accuracy over both C4.5rules and DLG alone (in two further cases, those relating to the lymphography data, the results very closely approach significance at the 0.05 level). In twelve cases the use of DLGref2 leads to a statistically significant improvement in accuracy to that obtained by at least one of C4.5rules or DLG alone. In only two cases, those for the hypothyroid data, does the use of DLGref2 lead to a significant decrease in accuracy over that obtained by C4.5rules. In no case does the use of DLGref2 lead to a significant decrease in accuracy to that obtained by DLG alone.

These results are an outstanding endorsement of the DLGrref2 algorithm, especially when one considers the difficulties posed for a DLG style algorithm when refining the style of rules developed by C4.5rules, as discussed above.

	Complete & Consistent					**Noise**				
Domain	**DLG ref2**	**C4.5 rules**	***p***	**DLG**	***p***	**DLG ref2**	**C4.5 rules**	***p***	**DLG**	***p***
breast cancer	68.5	68.3	0.431	68.1	0.300	70.0	68.3	**0.018**	69.6	0.236
echocardiogram	71.2	70.4	0.368	70.0	0.179	70.1	70.4	0.434	71.3	0.216
glass type	74.8	71.6	**0.009**	71.0	**0.000**	74.3	71.6	**0.026**	71.4	**0.002**
hepatitis	81.8	79.7	**0.021**	82.2	0.305	81.0	79.7	0.136	81.9	0.154
house votes 84	94.2	93.9	0.240	94.1	0.365	94.0	93.9	0.396	91.1	**0.000**
hypothyroid	98.9	99.3	0.000	98.1	**0.000**	98.9	99.3	0.000	97.3	**0.000**
iris	94.9	93.2	**0.009**	93.5	**0.004**	94.7	93.2	**0.016**	93.5	**0.013**
lymphography	77.7	78.8	0.171	75.9	0.056	77.7	78.8	0.179	75.6	0.051
F11 multiplexer	96.5	84.9	**0.000**	91.7	**0.000**	94.5	84.9	**0.000**	81.2	**0.000**
primary tumor	36.3	36.5	0.430	35.6	0.056	38.5	36.5	**0.013**	38.6	0.387

Table 2: Summary of experimental results (mean accuracies over 100 tests)

It is interesting to attempt to characterise the types of domains for which DLGref2 is successful and those for which it is not. There are two data sets for which the use of the binomial value function, but not the complete and consistent value function, leads to a significant improvement in performance for the refined expert system over that of the original expert system - breast cancer and primary tumor. A one-tailed matched pairs t-test reveals that DLGref2 with the binomial value function significantly outperformed DLGref2 with the complete and consistent value function for both of these data sets (breast cancer: p=0.11; primary tumor: p=0.000*).* Table 1 reveals that these are the only data sets which contain indistinguishable cases (cases with identical descriptions belonging to different classes). As a result, these are the only data sets for which it is not possible to create an expert system that is complete and consistent with regard to the data. As DLGref2 with the complete and consistent value function

attempts to create an expert system that is complete and consistent with regard to the data it is to be expected that it should not perform well with this data. Indeed, a one-tailed matched pairs t-test reveals that DLG with the binomial value function significantly outperforms (breast cancer: p=0.024; primary tumor: p=0.000) DLG with the complete and consistent value function for these domains alone; whereas DLG with the complete and consistent value function significantly outperforms DLG with the binomial value function for the house votes 84 (p=0.000), hypothyroid (p=0.000) and F11 multiplexer (p=0.000) data sets.

It is not apparent why DLGref2 with the complete and consistent value function should significantly improve upon the expert system created by C4.5rules for the hepatitis domain whereas the use of the binomial value function does not. As there is not a significant difference between the accuracies obtained by DLG alone (p=0.288) or DLGref2 (p=0.138) with each of these value functions for this data set, it is perhaps inadvisable to draw strong conclusions from the difference in this case.

DLGref2 with the complete and consistent value function significantly outperformed DLGref2 with the binomial value function for the F11 multiplexer data (p=0.000) only. Other than the significant differences already noted for the breast cancer, primary tumor and F11 multiplexer data sets, the difference in accuracies between DLGref2 with the complete and consistent and with the binomial value functions were not significant (echocardiogram: p=0.212; glass: p=0.292; hepatitis: p=0.138; house votes 84: p = 0.293; hypothyroid: p=0.l69; iris: p=0.239; lymphography: p=0.479).

The significant improvement in accuracy obtained by DLGref2 over that obtained by DLG alone when employing the binomial value function with the house votes 84 data can be attributed to the poor performance of DLG in this context.

There are a number of possible reasons why the use of DLGref2 should lead to a significant decrease in performance for the hypothyroid data. This data set is distinguished by having the most cases, the most attributes, the highest proportion of cases belonging to a single class and the equal highest number of missing values. Of these, it is plausible that having the highest proportion of cases belonging to a single class is most relevant, as this enables C4.5rules to create a very simple expert system, essentially noting only exceptions to the policy of assigning all cases to the most common class. Due to the policy of requiring every rule to stand in its own right, DLG and DLGref2 create substantially more complex expert systems in these contexts.

Table 3 presents the mean number of rules in the expert systems created by each system, along with the result of a one-tailed matched pairs t-test comparing differences between DLGref2 and the other two algorithms. As can be seen, DLGref2 creates significantly more rules than C4.5rules alone in all cases. DLGref2 creates significantly more rules then DLG alone for all tests other than hypothyroid when using the complete and consistent value function and F11 multiplexer using the complete and consistent value function. For the former there is no significant difference whereas for the latter DLGref2 produces significantly less rules. These results are not surprising when one considers the manner in which C4.5rules optimises the rule sets that it produces and that the rules produced by DLGref2 reflect more information (C4.5rules' rules and training set 2) than that used by either C4.5rules (training set 1) or DLG (training set 2) alone.

	Complete & Consistent					Noise				
Domain	**DLG ref2**	**C4.5 rules**	***p***	**DLG**	***p***	**DLG ref2**	**C4.5 rules**	***p***	**DLG**	***p***
breast cancer	22.6	5.6	0.000	20.7	0.000	17.3	5.6	0.000	15.5	0.000
echocardiogram	6.6	3.5	0.000	5.5	0.000	6.4	3.5	0.000	4.8	0.000
glass type	13.7	7.2	0.000	12.1	0.000	13.8	7.2	0.000	10.9	0.000
hepatitis	7.8	4.0	0.000	5.5	0.000	7.2	4.0	0.000	4.6	0.000
house votes 84	11.0	4.4	0.000	10.2	0.000	8.5	4.4	0.000	6.1	0.000
hypothyroid	12.7	7.6	0.000	12.7	0.407	12.7	7.6	0.000	10.1	0.000
iris	5.3	3.4	0.000	4.9	0.000	5.7	3.4	0.000	4.5	0.000
lymphography	12.1	6.4	0.000	9.2	0.000	11.7	6.4	0.000	8.0	0.000
F11 multiplexer	27.7	18.6	0.000	32.6	0.000	27.6	18.6	0.000	25.4	0.000
primary tumor	65.4	10.6	0.000	60.8	0.000	68.0	10.6	0.000	62.0	0.000

Table 3: Summary of rule set complexity (mean number of rules over 100 tests)

4 Future Research

The experimental results that have been obtained suggest that the refinement of the form of flat expert systems explored has been substantially mastered. It is of interest to examine whether the techniques can be extended to cover more complex forms of expert system. Of particular interest is the capacity of the approach to handle expert systems with multiple reasoning steps (the consequent of one rule can be used in the condition of another), probabilistic rules and rules optimised to take account of conflict resolution strategies.

Extension to handle probabilistic rules should not be problematic. Using the binomial value function, the system is already developing rules that are not consistent with the training set. To be regarded as probabilistic rules, these need only be further evaluated to add a probability assessment. Such evaluation would, presumably, take account of both the probability assigned the unrefined version of the rule and the number of positive and negative cases in the training set covered by the refined rule.

Extension to optimise the rule set with respect to the conflict resolution strategy employed should also be straight forward. To achieve this it should only be necessary, when refining a rule, to remove from the training set all cases covered by rules with higher precedence than the rule under examination. Depending upon the conflict resolution strategy employed, it may also be desirable to examine the relative priority of each rule. For example, with weighted rules it may be desirable to adjust rule weights before and/or after DLGref2 refinement, using techniques such as those developed by Rada (1985) and Caruana (1989).

It is less apparent to what degree DLGref2 requires modification in order to apply to the refinement of rule sets with multiple reasoning steps. It could certainly be used without modification to refine the end point rules, those whose consequents do not appear in the conditions of further rules. In order to tackle multiple reasoning steps, it may be necessary to integrate DLGref2 with techniques, such as those of Ourston &

Mooney (1990) that take account of the manner in which an alteration to one intermediate rule may generalise and/or specialise several other rules.

5 Conclusions

DLGref2 is a machine learning algorithm that supports inductive refinement of existing rules. DLGref2 creates a rule set consisting of rules that can each be interpreted in isolation of the rule set in which they are embedded. A precursor to this algorithm, DLGref, demonstrated the capacity in a wide variety of contexts to refine rule sets for which this condition held (Webb, 1992*c*). This study has demonstrated that the new version of the algorithm is able to improve the accuracy of initial rule sets for which this condition does not hold. This provides a strong indication of the capacity for the algorithm to operate with partial and incomplete rules. DLGref2 is able to operate both with data for which it is possible to create complete and consistent classifiers, and data for which it is not.

DLGref2 is restricted with respect to the types of rules on which it is designed to operate. Rule-bases with multiple reasoning steps, probabilistic rules and rules optimised with respect to an evaluation function will be desirable for many applications. However, the forms of rules inferred by DLGref2 also have advantages in some contexts. In the context of the integration of machine learning with knowledge acquisition, the use of these style of rules increases the modularity and clarity of the rule sets, especially to those who are not highly trained knowledge engineers. It is precisely for use in such a context that DLGref2 is designed.

Further, considered in the wider context of knowledge-base refinement algorithms, there are advantages to developing and thoroughly evaluating inductive knowledge-base refinement algorithms in simple, readily controlled and manipulated contexts, before applying them in more complex contexts. Ongoing research is extending the techniques developed to date to more complex contexts.

DLGref2 has been successfully incorporated in the Einstein knowledge acquisition system. This system enables a human expert to collaborate with a machine learning subsystem at all stages or the knowledge acquisition process. The success of DLGref2 and the environment in which it is embedded demonstrates that the integration of machine learning and knowledge elicitation is feasible within the current state-of-the-art.

Acknowledgments

This research has been supported by the Australian Research Council and the Apple University Development Fund. I am grateful to Ross Gollan for statistical advice and comments.

References

Caruana, R. A., **"The automatic training of rule bases that use numerical uncertainty representations,"** in *Uncertainty in Artificial Intelligence 3*. Kanal, L. N., Levitt, T. S. & Lemmner, J. F. (Eds.) Elsevier Science, 1989.

Davis, R. & Lenat, D. B., *Knowledge-Based Systems in Artificial Intelligence*. McGraw Hill, 1982.

Ginsberg, A., *Automatic Refinement of Expert System Knowledge Bases*. Pitman, 1988.

Ginsberg, A., Weiss, S. M. & Politakis, P., **"Automatic knowledge base refinement for classification systems."** *Artificial Intelligence*, Vol. 35, pp. 197-226, 1988.

Lee, W. D. & Ray, S. R., **"Rule refinement using the probabilistic rule generator,"** in *Proceedings of the Fifth National Conference on Artificial Intelligence*, Morgan Kaufmann, 1986.

Ma, Y. and Wilkins, D. C., **"Improving the performance of inconsistent knowledge bases via combined optimization method,"** in *Proceedings of the Eighth International Machine Learning Workshop*, pp. 23-27, 1991.

Muggleton, S. & Feng, S., **"Efficient induction of logic programs"** *in Proceedings of the First Conference on Algorithmic Learning Theory*, Tokyo, 1990.

Murphy, P. & Aha, D. **UCI Repository of machine learning databases**. [Machine- readable data repository]. University of California, Department of information and Computer Science, Irvine, CA, 1992.

Ourston, D. & Mooney, R. J, **"Changing the rules: A comprehensive approach to theory refinement."** in *AAAJ-90*, pp. 8 15-820, 1990.

Pazzani, M. J. & Brunk, C. A., **"Detecting and correcting errors in rule-based expert systems: An integration of empirical and explanation-based learning."** *Knowledge Acquisition*, Vol. 3, pp. 157-173, 1991.

Quinlan, J. R., **"Generating production rules from decision trees."** *Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, Morgan Kaufmann, 1987.

Quinlan, J. R. *C4.5: Programs for Machine Learning*, Morgan Kaufmann, 1992.

Rada, R., **"Gradualness facilitates knowledge refinement,"** *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-7, pp. 523-53 1, 1985.

Reinke, R. B. & Michalski, R. S., **"Incremental learning of concept descriptions: A method and experimental results,"** in *Machine Intelligence* 11, Hayes, J.E., Michie, D. & Richards, J. (Eds.), Clarendon Press, 1988.

Smith, R. G.. Winston, H. A., Mitchell, T. M. & Buchanan, B. G., **"Representation and use of explicit justifications for knowledge base refinement,"** in *Proceedings of the Ninth International Joint Conference on Artificial Intelligence.*, Morgan Kaufmann, 1985.

Webb, G.I., **Accommodating noise during induction by generalization**. *Technical Report* TR C92/13, Deakin University School of Computing and Mathematics, 1992*d*.

Webb, G.I., **"Control. Capabilities and Communication: Three Issues for Machine-expert Collaborative Knowledge-acquisition**. To be published in the *Proceedings of the 1993 European Knowledge Acquisition Workshop*, Toulouse, 1993*a*.

Webb, G.I., **Data-driven inductive knowledge-base refinement**. *Technical Report* TR *C92/10*, Deakin University School of Computing and Mathematics, 1992*c*.

Webb, G.I., **Inductive generalisation of complete and consistent production rules**. *Technical Report* TR C93/l5, Deakin University School of Computing and Mathematics, 1993*b*.

Webb, G.I., **Learning Disjunctive Class Descriptions by Least Generalisation**. *Technical Report* TR *C92/9*, Deakin University School of Computing and Mathematics, 1992*b*.

Webb, G.I., "**Man-machine collaboration for knowledge acquisition,"** in *Al '92*. World Scientific, 1992*a*.

Webb, G.I., **"Rule optimisation and theory optimisation: Heuristic search strategies for data-driven machine learning**. In *Knowledge Acquisition for Knowledge-Based Systems*, Motada, H., Mizoguchi, R., Boose, J. & Gaines, B. (Eds) 10S Press, 1991.

Webb, G.I., **Search techniques for induction by generalization**. *Technical Report* TR C92/12 Deakin University School of Computing and Mathematics, 1992*e*.

Webb, G.I. & Agar, J., **"Inducing diagnostic rules for glomerular disease with the DLG machine learning algorithm**." *Artificial Intelligence in Medicine*, 4: 3-14, 1992.

Wilkins, D.C. & Buchanan, B. G., **"On debugging rule sets when reasoning under uncertainty,"** in *AAAI-86: Proceedings of the Fifth National Conference on Artificial Intelligence*, Philadelphia, pp. 448-454, 1986.

Appendix A

algorithm DLGref2

Inputs: *rules:* an initial set of rules for a single class
POS: a set of examples belonging to that class
NEG: a set of examples that do not belong to the class
value: a function from rules to numeric values such that the higher the value the greater the preference for the rule. This function will usually take account of the number of positive and negative cases covered by the rule.

Output: *rules:* a revised set of rules for the conclusion

```
for r is set to each rule in rules in succession
    if r covers no negative cases
      remove from POS all cases that r covers
    end if
end for
for r is set to each rule in rules in succession
    if r covers negative cases
      spec_rule <- induce_rule (covered_cases(r,POS), NEG, value)
      if spec rule is not if FALSE then positive
          r <- select_rule_from_region_of_maximal_cover(r, spec_rule)
      end if
      remove from POS all cases that r covers
    end if
end for
while POS is not empty
    new_rule <- induce_rule(P0S, NEG, value)
    if  new_rule is if FALSE then positive
      remove all remaining cases form POS
    else
      new_rule <- select_rule _from_region_of_maximal_cover(if TRUE then positive,
      new_rule)
      remove from POS all cases that new_rule covers
      add new_rule to rules
    end if
end while
```

algorithm induce_rule

Inputs: *POS:* a set of examples belonging to that class
NEG: a set of examples that do not belong to the class
value: a function from rules to numeric values such that the higher the value the greater the preference for the rule This function will usually take account of the number of positive and negative cases covered by the rule.

Output: *rule:* a revised set of rules for the conclusion

```
rule <- if false then positive
```

for *c* is set to each case in *POS* ordered from most to least central
  r <- *least_generalisation(rule, c)*
  if *r* covers no cases in *NEG*
    rule <- *r*
    exit from the for loop without examining any more cases
  end if
end for
if *rule* ≠ `if false then positive`
  for *c* is set to each case in *POS* ordered from least to most central
    r <- *least_generalisation(rule, c)*
    if *r* covers no cases in *NEG*
      rule <- *r*
    end if
  end for
end if

algorithm select_rule_from_region_of_maximal_cover

Inputs: *gen_rule:* a rule representing the most general bound of the version space to be explored
spec_rule: a rule representing the most specialised bound of the version space to be explored. This rule must be a specialisation of *gen_rule.*

Output: *rule:* a rule from within the version space that is as general as possible while covering no more negative cases than *spec_rule.*

Re-express *gen_rule* and *spec_rule* in conjunctive normal form.
while *spec_rule* ≠ *gen_rule*
  for each conjunct *c* in the condition of *spec_rule*
    if deleting *c* from *spec_rule* increases *spec_rule's* cover of negative cases
      add *c* to the condition of *gen_rule*
    end if
end for
for each conjunct *c* in the condition of *spec_rule*
    if *c* is not in the condition of *gen_rule* and adding *c* to the condition of *gen_rule* does not decrease the negative cover of *gen_rule*
      remove *c* from *spec rule*
    end if
end for
  find the conjunct *c* from the condition of *spec_rule* that is not in the condition of *gen_rule* such that when *c* is added to the condition of *gen_rule* negative cover is maximised
  remove *c* from *spec_rule*
end while
rule <- *gen_rule*

least_generalisation(rule, case) returns a least generalisation of the rule against the case. A rule *l* is a least generalisation of rule *r* against case *c* iff

- *l* is a generalisation of *r;*
- *l* covers *c;* and

- there is no generalisation of *r* that is also a specialisation of *l* and which also covers *c*.

The centrality of a case *c* from a set of cases *S* is measured by

$$\sum_{i=1}^{\#S} dist(c, S^i)$$

where

- #*S* is the number of cases in *S;*
- S^i is the *i*th case in *S* according to any arbitrary ordering of cases; and
- $dist(a,b) = \sqrt{\sum_{i=1}^{n} dist(a^i, b^i)^2}$

where

- *n* is the number of attributes in the domain; and
- $dist(a^i, b^i)$ represents the distance between the values of the *i*th attribute for *a* and *b.*

To prevent bias arising from the use of different scales for each attribute, the values of each attribute are re-scaled to a value between 0 and 1 inclusive. This can be achieved for ordinal attributes by the formula

$$scaled = \frac{val - \min}{\max - \min'}$$

where *val* is the unscaled value and *min* and *max* are the minimum and maximum values for the attribute, respectively.

For categorical attributes, it is not appropriate to consider the space defined by the dimension as Euclidean. Rather, all points in the dimension should be considered equidistant to all other points. The distance between any two different categorical values for the same attribute equals $\frac{1}{n}$, where *n* is the number of values for the attribute.

It is also necessary to consider attributes that can assume both categorical and ordinal values. This arises in when an ordinal attribute may also assume the value *unknown.* In this circumstance, the value *unknown* is assumed to be equidistant from all other values for the attribute, the distance being $\frac{\#unknown}{\#cases}$, where *#unknown* is the number of cases for which the value of the attribute is unknown and *#cases* is the total number of cases in the training set. Ordinal values are scaled by the formula

$$scaled = \frac{val - \min}{\max - \min} . \frac{\#known}{\#cases}$$

where *val* is the unscaled value, *#known* is the number of cases for which the value of the attribute is not unknown, *#cases* is the total number of cases and *min* and *max* are the minimum and maximum values for the attribute, respectively.