Pre-publication draft of a paper which appeared in the Proceedings of PKAW'96: The Pacific Knowledge Acquisition Workshop, pp170-188, UNSW

Experimental evaluation of integrating machine learning with knowledge acquisition through direct interaction with domain experts

Geoffrey I. Webb and Jason Wells

School of Computing and Mathematics, Deakin University, Australia.

Abstract

Machine learning and knowledge acquisition from experts have distinct and apparently complementary knowledge acquisition capabilities. This study demonstrates that the integration of these approaches can both improve the accuracy of the knowledge base that is developed and reduce the time taken to develop it. The system studied, called The Knowledge Factory is distinguished by the manner in which it supports direct interaction with domain experts with little or no knowledge engineering expertise. The benefits reported relate to use by such users. In addition to the improved quality of the knowledge base, in questionnaire responses the users provided favourable evaluations of the integration of machine learning with knowledge acquisition within the system.

1 Introduction

On the face of it, machine learning and knowledge acquisition from experts provide differing and complementary means of developing knowledge-based systems. The apparent manner in which the strengths of one match the weaknesses of the other led to the development of a number of systems that integrate the two approaches (Attar Software, 1989: Davis & Lenat, 1982; De Raedt. 1992; Le Grand & Sallantin 1994; Monk et al., 1993; Nedellec & Causse, 1992: O'Neil & Pearson, 1987; Schmalhofer & Tschaitschian, 1995; Smith *et al* 1985; Tecuci & Kodratoff, 1990; Tecuci, 1995; Webb, 1996; Wilkins, 1988). This integration is expected to have a synergistic effect with the power of the resulting combined approach being greater than the power of either of its components. However, while there have been numerous reports of successful applications of these tools (Monk *et al.*, 1993; Tecuci & Kodratoff, Webb, 1996), previous research has not demonstrated that the results in any way exceeded those that could have been obtained by one of the component approaches alone. This paper presents a formal evaluation of the benefits obtained through integrating machine learning into a knowledge acquisition environment in a system called *The Knowledge Factory* (Webb, 1996).

2 The Knowledge Factory

The Knowledge Factory (Webb. 1996) is an interactive knowledge acquisition environment that was developed with the intention of enabling a domain expert to collaborate with a machine learning system throughout the knowledge acquisition and maintenance process. Like the approach of Tecuci, 1995, it is distinguished from learning apprentices (Attar Software. 1989: Davis & Lenat, 1982; De Raedt, 1992: Monk *et al*, 1993; Nedellec & Causse, 1992; O'Neil & Pearson, 1987; Schmalhofer & Tschaitschian, 1995; Smith *et al*, 1985; Tecuci & Kodratoff, 1990; Wilkins, 1988) by the manner in which it is designed to be used directly by experts with minimal knowledge engineering training or experience. By contrast, learning apprentices are designed to provide machine learning facilities for use by knowledge engineers. It is distinguished from a number of knowledge elicitation systems designed for direct use by experts (Boose, Compton *et al*, 1992) not only by its provision of machine learning facilities, but also by not relying upon the expert to always be able to provide suitable task solutions.

It is distinguished from the approach of Tecuci. 1995 by its use of less complex forms of interaction with the user. Restriction to simple user interactions is believed to be appropriate for the target user population: domain experts with little or no knowledge engineering experience or training. In particular, the interface and knowledge representation scheme has been kept simple. The knowledge representation scheme is restricted to flat attribute-value classification rules. That is, the knowledge base consists of a set of production rules. The antecedent of a rule is a set of tests on attribute values. The consequent is a simple classification statement. All rules directly relate input attributes to an output class.

Experience with the use of The Knowledge Factory in complex financial and medical knowledge acquisition tasks indicated that both experienced knowledge engineers and users with minimal knowledge engineering or computing skills believed that the software was a valuable knowledge acquisition aid (Webb, 1996). A formal study in which university students in a third year artificial intelligence and expert systems unit were given an artificial knowledge acquisition task, found that the subjects believed that the integration of machine learning into the system was valuable and found the software easy to use (Webb, 1996). None of this evaluation, however, establishes any comparative advantage for the integration of machine learning with knowledge elicitation, as done by The Knowledge Factory, over any alternative. The current study was performed in order to seek support for the proposition that there exist knowledge acquisition tasks for which knowledge acquisition is assisted by the integration of machine learning as supported within The Knowledge Factory.

To this end, two versions of the software were developed, one containing the machine learning facilities and one from which these facilities were removed. This enabled a direct evaluation of the effect of those facilities upon the knowledge acquisition process.

It should be noted that, even with the machine learning facilities removed, The Knowledge Factory is still a fully functional knowledge acquisition environment. It contains both extensive facilities for specifying and editing rules and for evaluating the performance of those rules on example data.

The two versions of the system were compared through use in an assignment for a third year undergraduate university computer science unit. The use of undergraduate computer science students with minimum knowledge acquisition training and no knowledge acquisition experience was believed to be appropriate as the tool is intended for users with little or no training in knowledge engineering.

As the system is intended for users with relevant domain knowledge such knowledge was simulated in the experiments by providing the subjects with tuition in the subject matter before knowledge acquisition began.

3 Experimental Method

All twenty-nine students in the third year unit *Artificial Intelligence and Expert Systems* at Deakin University were given an assignment that involved two knowledge acquisition tasks. All students involved were asked whether they would consent to have their performance utilised in a research study and were told that they could withdraw their consent at any stage during the experiment.

Only one subject had any knowledge engineering experience prior to commencing the unit. This student was repeating the unit having attempted but failed it in the preceding year. The study commenced in the third week of the unit. Up to that point the students had been exposed to overview level discussions of knowledge acquisition and to programming in the CLIPS expert system language. During the study, the students received further lectures and laboratory sessions on CLIPS programming and two discursive lectures on knowledge acquisition principles and techniques. The student body comprised both Information Systems and Software Development students. Thus, the subjects, while having good computer skills, were, at best, novice knowledge engineers.

Both knowledge acquisition tasks were artificial. First a set of defining rules for a domain were created. These were intended to have a level of complexity sufficient to provide a spread of quality in the rules developed by different means. That is, they were not to be so simple as to enable any knowledge acquisition approach to develop perfect rules. Nor were they to be so complex as to ensure that all approaches developed rules of extremely low quality. These rules are presented in Figure 1.

The domains are defined by a dependent variable with four values (the classes) and 15 independent variables of which ten are ordinal ($a^A0 \ldots a^A9$) with integral values between 0 and 1000 inclusive, and five are categorical ($c^A0 \ldots c^A4$) with values of true or false. Of these 15 independent variables, eight were generated by independent random number generation processes while the remaining seven were each derived by random transformation of other variables. This inter-relationship between variables was used in an attempt to make the artificial task as realistic as possible. The data generation functions are presented in Table 1.

Two data sets were generated using the data generation functions presented in Table 1 and augmented by the dependent variable generated as per Figure 1. The first of these, called the training set, contained two hundred items while the second, called the evaluation set contained one thousand items. In addition, a body of background knowledge was defined. This was designed to provide the subject with a set of beliefs about the domain in order to simulate a real domain expert with extensive, but neither completely accurate nor exhaustive knowledge about the domain.

The two data sets and the background knowledge defined the base knowledge acquisition task.

IF $a^A0 \geq 450$
c^A0 is true
c^Al is true
THEN Class = 0

IF $a^A0 \geq 450$
$c^Al \leq 200$
c^A0 is false
c^Al is true
THEN Class = 1

IF $a^A0 \leq 449$
c^A0 is true
c^Al is true
THEN Class = 1

IF $a^A0 \geq 400$
c^A0 is true
c^Al is false
$a^A2 \leq 700$
THEN Class = 2

IF $a^A0 \geq 450$
$a^A1 \leq 400$
c^A0 is true
c^Al is false
THEN Class = 2

OTHERWISE Class = 3

Figure 1: Defining rules for the knowledge acquisition tasks

On the one hand it was desirable to give all subjects the same task and for each subject to use each version of the software on the one task. This would prevent experimental confounds being introduced by irrelevant differences between tasks. However, the straightforward use of a single task would introduce the risk of collaboration between subjects, especially collaboration between subjects in different treatments (those with access to machine learning could report the rules developed through machine learning to colleagues without access to those facilities). Further, if each subject used two different systems on the one task, rules developed using one system could be entered into the knowledge base for the other system.

To limit the potential for either of these occurrences, the base knowledge acquisition task was transformed for each subject. First, two scenarios were defined: the Gruwald's disease diagnosis scenario and the geochemical analysis scenario. Each scenario was defined by:

- a textual briefing:
- a set of names for the ordinal variables:
- a set of names for the categorical variables;
- a set of class names; and
- a set of transformation functions.

The latter were employed to transform the values of the ordinal variables from the base task. These scenario definitions are shown in Figures 2 and 3.

$$a^A 0 = rand(0..499) + rand(0..499)$$

$$a^A 1 = rand(0..499) + rand(0..499) - \frac{a^A 0}{3}$$

$$a^A 2 = \mathrm{a} + rand(0..499) — rand(0..499)$$

$$a^A 3 = \begin{cases} rand(0..199) + rand(0..199) & \text{if } \mathrm{c}^A 0 \\ rand(0..299) + rand(0..299) + rand(0..299) & \text{otherwise} \end{cases}$$

$$a^A 4 = \begin{cases} rand(0..299) + rand(0..299) + rand(0..299) & \text{if } \mathrm{c}^A 0 \wedge \neg\ \mathrm{c}^A 1 \\ rand(0..199) + rand(0..199) & \text{otherwise} \end{cases}$$

$$a^A 5 = rand(0..999)$$

$$a^A 6 = rand(0..999)$$

$$a^A 7 = rand(0..999)$$

$$a^A 8 = rand(0..999)$$

$$a^A 9 = rand(0..999)$$

$$c^A 0 = rand\ (true, false)$$

$$c^A 1 = rand(\ true, true, false)$$

$$c^A 2 = c^A 0\ \wedge\ rand\ (true, true, false)$$

$$c^A 3 = a^A \mathrm{l} + rand(0..99) \leq 170$$

$$c^A 4 = rand\ (true, false)$$

Table 1: Data generation functions

The definition of different scenarios reduced the risk of subjects realising that at an underlying level, the two tasks were identical. To further reduce this risk and to reduce the risk of subjects realising that they shared common tasks with other subjects (with whom they could freely

communicate) within each scenario, each subject was provided with an individual *surface task* by the addition of a random offset to the values for each variable.

As an additional measure to reduce the risk of different subjects realising that they shared tasks that were identical at an underlying level, the subjects were told that each had tasks defined by different sets of defining rules.

To minimise any effect whereby the subject's performance on one task might affect performance on the other, in particular due to time being apportioned unduly to one task at the expense of the other, the tasks were performed in sequence. The Gruwald disease diagnosis task was performed first. Subjects had to collect disks containing the software, data, briefing and manuals on one Tuesday and return it, with the completed project on the following Tuesday. When a subject submitted the first project he or she was provided with the disks for the second project. This, in turn, was submitted on the third Tuesday in sequence.

Briefing: Gruwalds Disease Background Briefing

The following is a summary of your knowledge of Gruwald's Disease accumulated from years of clinical experience.

1. Gruwald's Disease is highly dangerous. The mortality rate for patients with the acute form of the disease is very high.
2. Patients with Gruwald's Disease usually have raised Erythrocyte_Count and raised Liptuary.
3. Accurate diagnosis is currently only possible posthumously.
4. In your experience, patients with acute Gruwald's Disease usually have Erythrocyte_Count $\geq$ 450, Dyspnoea and Generalised_Oedema are present.
5. You believe that when Ervthrocyte_Count $\geq$ 450 and Generalised_Oedema is present but Dyspnoea is absent, patients have Advanced Gruwald's Disease when Haemoglobin $\leq$ 250. While this rule-of-thumb is indicative, you know that it is not definitive.
6. Finally, you believe that when Erythrocyte_Count $\geq$ 400, Generalised_Oedema is present but Dyspnoea is absent and when Liptuary $\leq$ 600 patients have Secondary Gruwald's Disease.

Ordinal variable names

Ervthrocvte_Count, Haemoglobin, Liptuary, Pheneral_Rate, Platelet_Count, Creatine_Clearance, Granular_Cell_Count. Urinary_Ervth_Count, Proteinuria_Value, White_Cell_Count.

Categorical variable names

Dyspnoea. Generalised_Oedema. Haemoptysis. Headache, Peripheral_Oedema.

Class names

Acute, Advanced, Secondary, Negative.

Transformation functions

For each subject a conversion factor was generated for each continuous attribute. Each factor was a random number between -100 and 100, inclusive. All attribute values, including those in the briefing above, were modified by addition of the conversion factor.

The order within the briefing of the two categorical attributes Generalised_Oedema and Dyspnoea was determined randomly for each subject. If Dyspnoea was selected first, the order in which these two attributes are mentioned in each of the clauses of the briefing would be reversed to that presented above.

Figure 2: Definition of the Gruwald's disease diagnosis scenario

Each subject performed one task with the machine learning enabled version of the software and the other task with the machine learning disabled version. To minimise order effects and confounds introduced by differing perceptions of the two scenarios, half of the subjects were assigned the machine learning enabled version for the first task while half were assigned it for the second task. This assignment was randomised through use of a random number generator.

The software, manuals and data were given to the subjects on a computer disk. The performance of the task was unsupervised. Subjects could use appropriate computers in the University's laboratories, at home, or elsewhere.

Briefing

The information that follows represents your background knowledge on mineral prospecting using Geochemical Analysis. This knowledge is the product of your undergraduate training.

Various metals can be detected using geochemistry. Many ore samples are taken over a specified area. The samples are tested for their chemical composition. Combinations of chemicals with certain compositions indicate the possibility of a particular mineral deposit, providing evidence to support further exploration.

Based on the knowledge you obtained from university studying Geology you believe the following information on geochemical analysis is valuable and could be used as the basis of the expert system you are required to develop. The expert system could feasibly save millions of dollars in exploration costs depending on how well it is constructed.

From your geochemical analysis class you know gold deposits are usually present when samples indicate that Chromium $\leq$ 450, Fluorite is present and Barite is present. Generally gold deposits are present when Chromium levels are low.

You believe that when Chromium $\leq$ 450 and Barite is present but Fluorite is absent and Cobalt $\geq$ 250 the sample indicates the presence of silver deposits.

Although of less importance, you believe that when Chromium $\leq$ 400 *value*, Barite is present but Fluorite is absent and when Copper $\geq$ 600 the sample indicates the presence of Zinc deposits.

Ordinal variable names

Chromium, Cobalt, Copper, Tin, Iron, Lead, Manganese, Molybdenum, Nickel, Vanadium.

Categorical variable names

Fluorite, Barite, Gypsum, Azurite, Malachite.

Class names

Gold, Silver, Zinc, Other.

Transformation functions

The continuous values for the geochemical analysis scenario were transformed in 3 ways.

1. A conversion factor was determined by adding a random number between 1201 and 1401.
2. Each value was subtracted from 1000, which reversed the order of the values.
3. Each data set was assigned a divisor of either 10, 100 or 1000. This enabled the data to range from 3 decimal places to 1 decimal place.

For each subject, the order in which the categorical attributes Barite and Fluorite were mentioned within the briefing was determined randomly. If Fluorite was selected first, the order of these two attributes would be reversed in each paragraph of the briefing.

Figure 3: Definition of the geochemical analysis scenario

The subjects received only minimal training in the use of the software. This training took the form of a half hour demonstration of the use of the software in class. They were able to ask questions of the experimenters at any stage during the experiment but responses were restricted to details directly relating to how to operate the software. Other than this, the only assistance that the subjects obtained was in the form of access to the system's help facilities and to the user manual.

3.1 Software employed

The Knowledge Factory is a Macintosh based software system. Previous experience had shown that there was a tendency for students to explore the full range of features provided by the software. As the software can support multiple modes of machine learning and multiple modes of rule interpretation (Webb, 1996), and as these issues did not bear directly upon the issues to be explored by this study, these facilities were disabled. The default machine learning and rule interpretation settings were employed with one exception.

By default, The Knowledge Factory applies rules in a mode that allows the system to make no decisions. This outcome occurs when no rule covers a case or when multiple rules for different classes cover a case. Such results make it extremely difficult to compare the performance of alternative expert systems as there is no definitive manner in which to compare a system that achieves an accuracy of x^A 1% on y^A 1% of cases for which it reaches a conclusion with a system that achieves x^A 2% accuracy on y^A 2% of cases.

To obviate this problem The Knowledge Factory was set to a mode whereby when no rule applied to a case, the most common class from the training set (in this experiment. Class D) was assigned, and when multiple rules covered a case the highest quality rule (in terms of performance on the training set) was assigned. For this experiment the quality of a rule was judged by the function

$$quality = \begin{cases} -1 & \text{if } n > 0 \\ p & \text{otherwise} \end{cases}$$

where p is the number of cases correctly classified by the rule and n is the number of cases incorrectly classified. With this evaluation function the specific to general search used in this learning algorithm avoids rules that cover any negative cases. In consequence, there is no need to distinguish between the quality of alternative rules that cover negative cases.

Further features of the system that did not directly bear upon the experimental question but which had potential to seriously degrade performance if misused were also disabled. These were –

editing of the model: All facilities for adding, deleting or otherwise transforming attributes were disabled as subjects had access to no source of knowledge that could warrant such actions.

adding example cases: Subjects had no knowledge by which to generate new reliable example cases and hence the ability to generate new cases was disabled.

importing example cases and rules from external files: The ability to load from external files either additional example cases or sets of rules could not be used in a sensible manner within the scope of the defined scenarios and hence was also disabled.

deleting example cases: Subjects were informed that all example cases were accurate and hence had no basis on which to sensibly delete existing cases. Hence this facility was also disabled.

evaluation set: The Knowledge Factory supports the division of the available example cases into a training and an evaluation set. The latter is kept separate from the training data, is not accessed by the machine learning component and is not available to the user when developing rules. The number of example cases made available to the students was too small to enable this facility to be used in a useful manner. Hence, it was also disabled.

In addition, to prevent subjects from exchanging data between versions of the system or using other data analysis tools, the students were prevented from outputting the data in any form other than as a project file, the system's internal data representation format. Further, for ease of analysis, the 'Save As' facility was disabled ensuring that the one project name was used throughout the project.

To simplify the task of tracking progress, subjects were presented with a computer disk containing the appropriate version of the system along with a project file pre-loaded with the training data. The software was modified so as to require the system to be run from that disk and only on the original project file (although that file could be updated by the system under the user's direction).

The software was also modified to ensure that projects saved by one version of the system could not be input into another.

3.2 Experimental manipulation

Two versions of the software were created. The machine learning enabled version had the full functionality of The Knowledge Factory software other than the disabled features noted above. The machine learning disabled version was identical to the machine learning enabled version except that the following commands were disabled -

Develop New Rules: This command deletes any existing rules and then applies the DLG machine learning algorithm (Webb & Agar, 1992) (a variant of AQ (Michalski, 1984)) to the training examples to form a new set of rules.

Revise Current Ruleset: This command applies the DLGref2 inductive refinement algorithm (Webb, 1993) to refine the current set of rules. DLGref2 seeks to modify each of the existing rules the least amount necessary in order to optimise the preference criterion. The preference criterion defined by equation 1 was used in this study. The user is able to specify that selected rules are not to be modified in this process. After all existing rules have been processed new rules are added to the ruleset to cover any example cases not covered by the modified ruleset.

Revise Rules for Current Decision: This command is identical to Revise Current Ruleset except that only rules for the class of the currently selected rule are modified or added to the ruleset.

Form Alternative Rules: This command takes an existing rule and presents a set of alternative rules that correctly classify all example cases correctly classified by the original rules and incorrectly classify no example cases not incorrectly classified by the original rule.

It should be emphasised that while the machine learning disabled version of the soft ware did not contain the machine learning facilities described above, it still retained a comprehensive set of rule specification, editing and evaluation facilities.

3.3 Performance measures

The primary criterion that was used to measure performance was accuracy, when applied to the 1000 with-held cases, of the rule set submitted by the subject. One secondary measure was the complexity of the knowledge base developed. This was measured by the number of rules developed. Another secondary measure was the total time taken to complete the assignment. This was measured in terms of total running time of the software.

3.4 Tracking performance

Evaluation of the primary measure, predictive accuracy, was straight-forward. The example cases set aside for evaluation from the base task were transformed as appropriate into the subject's surface task. The submitted rule set was then applied to the transformed evaluation set and a simple score of the number of evaluation cases correctly classified was obtained.

To enable tracking of subject performance, a record was maintained of their actions during each task. Keeping track of performance during the task was not straight forward, however. Subjects were given disks containing the software and data. They were required to run the system from that disk only. A single project file had to be used throughout the task. As a result, some tracking could be performed by maintaining records within the project file.

However, it was possible for the subjects to either

- quit from the system after working with the project but without saving to the project file, or
- duplicate the project file and then at a later date substitute the saved copy for the modified original, hence effectively undoing all intervening work and deleting any records maintained in the modified original project file.

Both of these actions would prevent the recording of the subject's interactions with the system during the time in question. While these interactions could not directly impact upon the expert system that was developed (because any changes made would not be retained in the final project file), the subject's interactions could affect their understanding of the knowledge acquisition task and hence the actions could indirectly impact upon the final result. Due to these considerations, in addition to maintaining records in the project file, a log file was also kept. This was an independent file that was opened each time that the system was run and to which a record of each action was added immediately that the action was performed. Each action, including system activation was time stamped.

The records added to the project files took the form of a simple tally of the number of times that the action was performed.

3.5 Experimental design

The experimental design was matched pairs. The experimental units were subject-scenario tuples. Each subject participated in two such tuples, one in each treatment. Thus, each tuple could be matched to another by subject. Order effects and confounds due to effects of differing scenarios were minimised by having half of the subjects receive each treatment for each scenario.

This matched pairs experiment was followed by the administration of a questionnaire designed to elicit user's subjective evaluation of the alternative approaches. This questionnaire is described below.

4 Results

Twenty-eight students consented to participate at the commencement of the study and none withdrew thereafter.

Initial analysis (detailed below) showed that the machine learning enabled treatment resulted in higher average accuracy than the machine learning disabled treatment. However, this difference was not statistically significant.

Further analysis showed that a large number of the machine learning enabled predictive accuracies were identical to those obtained by rulesets created by application of machine learning alone to the training data. Inspection of the log files revealed that a large number of subjects had either:

- Started their assignment by applying the Develop New Rules command to learn a set of rules from the training data and then having discovered that these rules correctly handled all the training cases and, not attending to the briefing with which they were provided, had proceeded to ignore their simulated expertise.
- Having applied their simulated background knowledge, then applied the Develop New Rules command, thereby removing all influence of rules already defined. As a result, the subjects either ignored their simulated expertise or incorrectly believed it to have been taken into account by the induction process.

It was clear that for both of these sets of subjects; the experimental manipulations had failed to establish the desired experimental treatments. These machine learning enabled subjects had access to facilities that enabled the integration of machine learning with knowledge acquisition from experts but were developing rules through machine learning alone.

To enable analysis of only experimental units for which the experimental treatments were successfully established, subjects that employed the Develop New Rules command in the machine learning enabled treatment were discarded (from both treatments). However, there was some difficulty adequately identifying such subjects. When the Develop New Rules command is selected The Knowledge Factory presents a dialog in which it briefly explains that executing the command will delete all existing rules and asks the user whether they wish to continue or cancel. Unfortunately, the count maintained in the project file of the number of times that the command was executed was incremented even if the command was cancelled as a result of this dialog. Thus, if the count was zero then it could be concluded with certainty that the command had not been used (at least not in a sequence of interactions that had led directly to the formation of the set of rules in the submitted project file). However, if the count was not zero, it was still possible that the command had been cancelled and hence not executed. For such subjects it was possible to inspect the log files, in which cancellations were recorded. Unfortunately, it was not possible to determine from the log files whether a particular session had contributed directly to the submitted project file or not, as subjects could have replaced the project file created by that session with a copy saved previously. It was possible, nonetheless to ignore Develop New Rules commands for which the results were not saved. This was the case if there was no save in the session after the command was executed. To do this, the subject would have to explicitly tell the system not to save the changes to the project file when they quit. One other subject was also retained who had used Develop New Rules once only and who had deleted all the rules immediately after they were generated by the command, effectively restarting their project from the beginning.

Due to this process, 15 subjects were excluded leaving 13 subjects in the analysis. Detailed results both with and without this exclusion are reported below.

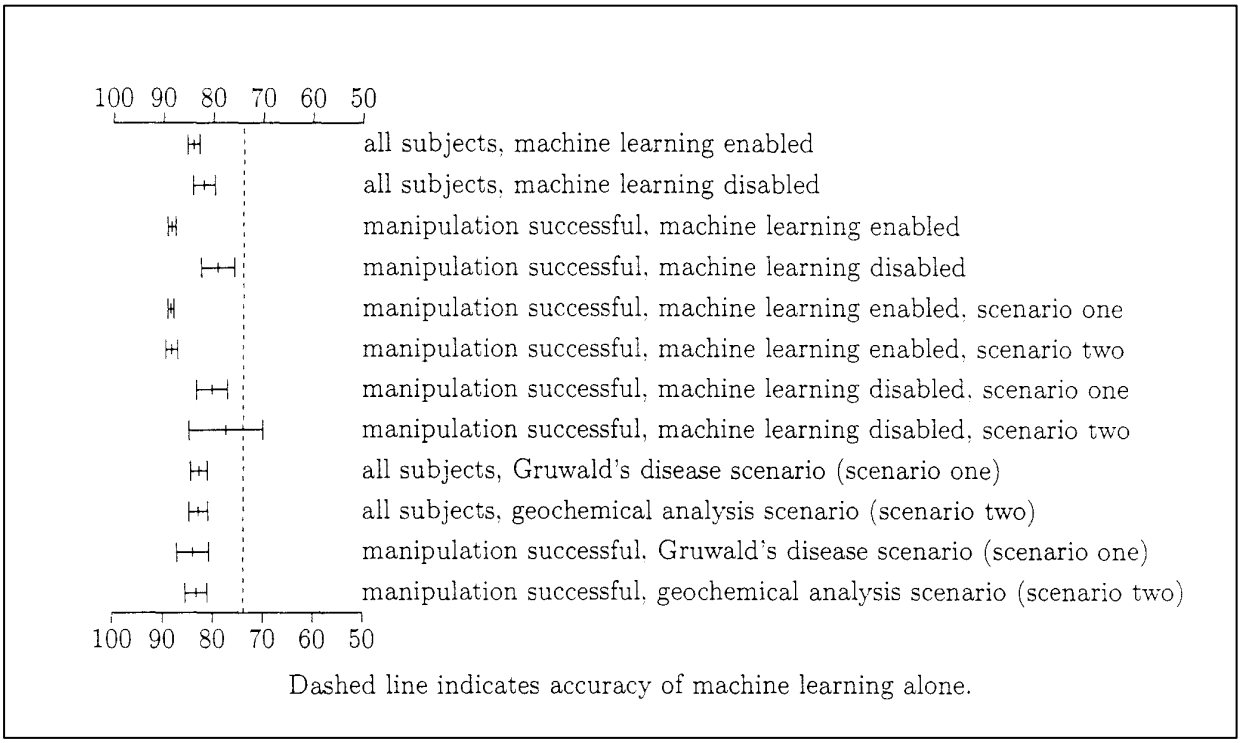

Figure 4: Predictive Accuracy

4.1 Predictive accuracy

Figure 4 summarizes the mean predictive accuracy obtained by each of various sub groups of the subjects. Accuracy is measured by the percentage of cases correctly classified. Error bars indicate one standard error. These outcomes are discussed in detail below.

Including all subjects, the predictive accuracy obtained for the machine learning enabled treatment was $\bar{x}$ = 83.9, s = 6.5. The predictive accuracy for the machine learning disabled treatment was $\bar{x}$ = 81.8, s = 11.6. A one-tailed matched-pairs t-test revealed that this difference was not significant at the 0.05 level. t = -0.80. p = 0.200.

For those 13 subjects remaining after excluding those identified above, the predictive accuracy obtained for the machine learning enabled treatment was $\bar{x}$ = 88.2, s = 2.8 and for the machine learning disabled treatment, $\bar{x}$ = 78.9, s = 11.9. A one-tailed matched pairs t-test revealed that this difference was significant at the 0.05 level (t = 2.69, p = 0.010).

More of the subjects that had machine learning enabled for the first scenario were excluded (9) than of those for the second scenario (6). It is conceivable that this is because subjects were less experienced with the software during the first scenario and hence more likely to use the system less effectively. To eliminate the possibility that this uneven exclusion might have confounded the results, within each condition mean predictive accuracies were determined for those receiving the treatment in each scenario. For the machine learning enabled condition, those 5 subjects receiving the condition for the first scenario obtained predictive accuracies $\bar{x}$ = 88.3, s = 1.4 and those 8 subjects that received the condition for the second scenario obtained $\bar{x}$ = 88.1, s = 3.5. A two-tailed pooled variance t-test revealed no significant difference between these results. t = 0.48, p = 0.363. For the machine learning disabled condition, those 8 subjects receiving the condition for the first scenario obtained predictive accuracies $\bar{x}$ = 80.0, s = 8.9 and those 5 subjects that received the condition for the second scenario obtained $\bar{x}$ = 77.2, s = 16.6. A two-tailed pooled variance t-test revealed no significant difference between these results, t = 2.03. p = 0.335. The small magnitude and lack of significance of the differences within each treatment between those subjects who received the treatment in each scenario suggests that this factor has not a significant confound, although the failure to find a significant difference must be treated with caution as the power of the pooled variance t-test for such small numbers is low.

An evaluation of the differences between treatments within each scenario provides further support for the proposition that the exclusion of more subjects with machine learning enabled for the first scenario than for the second has not confounded the results. For the first scenario, a one-tailed pooled variance t-test reveals that the machine learning enabled $\bar{x}$ = 88.3, s = 1.4 is significantly greater than the machine learning disabled $\bar{x}$ = 80.0, s = 8.9 (t = 9.35, p 0.006). For the second scenario, a one-tailed pooled variance t-test likewise reveals that the machine learning enabled $\bar{x}$ = 88.0, s = 3.4 is significantly greater than the machine learning disabled $\bar{x}$ = 77.2, s = 16.6 (t = 8.77, p = 0.042).

Rules developed by application of the machine learning system alone to the training data obtained a predictive accuracy of 73.8 when applied to the evaluation set. A one- tailed t-test revealed that the mean predictive accuracy obtained under the machine learning enabled condition ($\bar{x}$ = 83.9, s = 6.5) was significantly greater (t = 8.24, p < 0.005) than this value as was that for the machine learning disabled condition ($\bar{x}$ = 81.8, s = 11.6, t = 3.64, p < 0.005). Similar results were obtained when only those subjects remaining after the exclusions described above were considered (machine learning enabled: $\bar{x}$ = 88.2, s = 2.8, t = 27.06, p < 0.005; machine learning disabled: $\bar{x}$ = 78.9, s = 11.9, t = 2.24, p < 0.025). These results show the integration of machine learning with knowledge elicitation is providing an advantage that is not due solely to either the machine learning or the knowledge elicitation.

Another comparison of interest is the accuracies obtained for each scenario. For all subjects these accuracies were –

Gruwalds disease scenario: $\bar{x}$ = 82.6, s = 8.8;
Geochemical analysis scenario: $\bar{x}$ = 82.8, s = 10.0.

A two-tailed matched pairs t test comparison of these outcomes reveals no significant difference (t = -0.24, p = 0.468).

For those subjects remaining after exclusion of those for whom it could not be established that the background knowledge was employed, the accuracies were –

Gruwald's disease scenario: = 83.9, s = 11.4;
Geochemical analysis scenario: $\bar{x}$ = 83.2, s = 8.1.

A two-tailed matched pairs t-test comparison of these outcomes reveals no significant difference (t 0.43, p = 0.761). These results suggest that the surface knowledge acquisition tasks defined by the scenarios do not differ significantly in difficulty.

4.2 Complexity

The second major variable analysed was knowledge base complexity. For each expert system developed, the number of rules was recorded. For all subjects these numbers were –

Machine learning enabled: $\bar{x}$ = 15.7, s = 3.5;
Machine learning disabled: $\bar{x}$ = 7.4, s = 2.7.

A two-tailed matched pairs t-test comparison of these outcomes reveals that the group with access to machine learning developed significantly more rules (t = l3.3, p = 0.000). For those subjects remaining after exclusion of those for whom it could not be established that the background knowledge was employed, the complexities were –

Machine learning enabled: $\bar{x}$ = 14.6, s = 3.0;
Machine learning disabled: $\bar{x}$ = 7.3, s = 2.6.

A two-tailed matched pairs t-test comparison of these outcomes reveals that the use of machine learning resulted in knowledge bases containing significantly more rules than obtained when machine learning was not employed (t = 9 69, p= 0.000).

4.3 Knowledge acquisition time

The other major variable analysed was knowledge acquisition time. It was predicted that the subjects in the machine learning enabled condition would take less time to complete their projects than those in the machine learning disabled condition. A one-tailed matched pairs t-test was used to evaluate whether there was significant support for this prediction. The log files were analysed to determine the total time spent using each version of the software by each subject. The difference (machine learning enabled: $\bar{x}$ = 99 minutes, s = 115 minutes; machine learning disabled: $\bar{x}$ = 257 minutes, s = 179) was significant at the 0.05 level (t = -4.0l, p = 0.000). After exclusion of the 15 subjects for which it could not be determined that the background knowledge was employed, as detailed above, the difference (machine learning enable d: $\bar{x}$ = 51 minutes, s = 48; machine learning disabled: $\bar{x}$ = 237 minutes, s = 175) was still significant at the 0.05 level (t = -3.49, p = 0.006).

5 Questionnaire

In addition to measuring the three main variables, predictive accuracy, number of rules and knowledge acquisition time, the subjects were presented with a questionnaire. This is reproduced in Figure 5.

This questionnaire was distributed to the subjects when they collected the disks for the first scenario and was collected when they handed in the completed disks for the second scenario.

It was designed to evaluate a number of issues. The first four questions were designed t evaluate the subject's perception of the ease of use of the respective versions of The Knowledge Factory and to enable evaluation of how the version of the system employed for each task affected the perceived difficulty of the tasks and the perceived ease of use of the system. Questions 1 and 2 could be reinterpreted as la, *"How easy was it to use the system for the Gruwald's disease task?*". and 2a, *"How easy was it to use the system for the geochemical analysis task?"*. Likewise, questions 3 and 4 could be reinterpreted as 3a. *"How difficult was it to create an expert system for the task for which you used TKF_induction_on?* and 4a *"How difficult was it to create an expert system for the task for which you used TKF_induction_off?"* These reinterpretations are achieved as follows. For subjects given machine learning enabled for the Gruwald's disease task, the results for la, 2a, 3a and 4a are respectively the responses for questions 1. 2. 3 and 4. For the remaining subjects, the results for la, 2a, 3a and 4a are respectively the responses for 2. 1. 4 and 3.

Questions five and six were designed to evaluate the effect of the system employed on the subject's perception of the quality of the knowledge base developed. To reduce the influence on these results of the subject's attributions with respect to the two versions of the software, the systems were referred to by task rather than by system. However, the subject's perceptions with respect to 5a, *"How accurate do you think the expert system you created using TKF_induction_on will be when applied to the additional 1000 unseen evaluation cases?"* and 6a, *"How accurate do you think the expert system you created using TKF_induction_on will be when applied to the additional 1000 unseen evaluation cases?*", could be evaluated as follows. For a subject given machine learning enabled for the Gruwald's disease task, the result for 5a was the response to question 5 and the result for 6a was the response to question 6. For any other subject, these pairings were reversed.

Questions seven and eight were designed to evaluate the subject's perception of the relative usefulness of the two versions of the system. Question nine was designed to elicit the subject's perception, after using each version of the software, of the value of the main distinguishing feature between the two versions.

It was predicted that the responses would be higher for question 1 than 2 (subjects would find it easier to develop an expert system with the aid of machine learning). One tailed matched pairs t-tests support this prediction (all: $t = 7.15$. $p = 0.000$; retained: $t = 4.17$, $p = 0.001$). For the same reasons, it was predicted that the result for 3a would be lower than the result for 4a. However, one tailed matched pairs t tests failed to confirm this prediction (all: $t = 0.00$, $p = 0.500$, retained: $t = -0.26$, $p = 0.399$).

No predictions were made with respect to differences between questions 3 and 4 and la and 2a because it was not known whether, despite being based on the same underlying task, the subjects would perceive the scenarios to have different levels of difficulty.

Two tailed matched pairs t-tests show no significant differences. (Questions 3 and 4 - all: $t = 0.54$, $p = 0.558$; retained: $t = 1.76$, p 0.104. Questions la and 2a - all: $t = 0.44$, $p = 0.583$; retained: $t = -1.44$, $p = 0.175$).

Subjects were expected to anticipate higher predictive accuracy when using machine learning (5a) than when not (6a). This was confirmed by one tailed matched pairs t tests (all: $t = 3.17$, $p = 0.000$: retained: $t = 3.39$, p = 0.003). No prediction was made with respect to whether the subjects would expect a difference in predictive accuracy between scenarios (questions 5 and 6). Two tailed matched pairs t-tests show no significant differences. (all: $t = 0.09$, $p = 0.887$; retained: $t = -1.72$, $p = 0.110$).

Subjects were expected to find the version of the software that provided machine learning facilities more useful than the version that did not (questions 7 and 8). This was confirmed by one tailed matched pairs t-tests (all: $t = 3.47$, $p = 0.000$; retained: $t = 4.37$, $p = 0.004$).

Subjects were expected to provide high ratings for the value of machine learning for knowledge acquisition. One-tailed t-tests show that the mean responses were significantly higher than the middle value, 3 (all: $t = 9.36$, $p =< 0.005$; retained: $t = 6.50$, $p =< 0.005$).

SCC376 Assignment 1 - Questionnaire.

Rate each of the following from 1 - not at all to 5 - very. Circle the appropriate value for each question. If you are not certain of an answer do not circle any value.

1. How easy was it to use "TKF_Induction_On"?

not at all				very
1	2	3	4	5

2. How easy was it to use "TKF_Induction_Off"?

not at all				very
1	2	3	4	5

3. How difficult was it to create an expert system for Gruwald's disease?

not at all				very
1	2	3	4	5

4. How difficult was it to create an expert system for geochemical analysis?

not at all				very
1	2	3	4	5

5. How accurate do you think the expert system you created for Gruwald's disease will be when applied to the additional 1000 unseen evaluation cases?

not at all				very
1	2	3	4	5

6. How accurate do you think the expert system you created for geophysical analysis will be when applied to the additional 1000 unseen evaluation cases?

not at all				very
1	2	3	4	5

7. Do you think that "TKF_Induction_On" is a useful tool for building expert systems?

not at all				very
1	2	3	4	5

8. Do you think that "TKF.Induction_Off" is a useful tool for building expert systems?

not at all				very
1	2	3	4	5

9. How valuable do you think machine learning is for knowledge acquisition?

not at all				very
1	2	3	4	5

Notes

a) The question numbers were not included in the original questionnaire. They have been added above for ease of reference.

b) The names "TKF_Induction_On" and "TKF_Induction_Off" referred to the machine learning enabled and machine learning disabled versions of The Knowledge Factory software respectively.

Figure 5: Questionnaire

Table 2 lists the mean responses to these questions for all subjects and for those retained after exclusion for failure to employ background knowledge in the machine learning enabled treatment.

Question	All subjects	Retained subjects
1	4.50	4.54
2	2.86	3.00
3	3.18	3.31
4	2.96	2.38
5	3.39	3.15
6	3.36	3.69
7	4.32	4.46
8	3.29	3.08
9	4.21	4.38
la	3.57	3.38
2a	3.79	4.15
3a	3.07	2.77
4a	3.07	2.92
5a	3.79	3.85
6a	2.96	3.00

Table 2: Questionnaire results

5.1 Summary of questionnaire results

The questionnaire results show that the subjects

- believed the machine learning facilities to be useful.
- found knowledge acquisition easier when the machine learning facilities were avail able. and
- had greater confidence in the expert systems developed with the aid of machine learning.

While questions 3 and 4 were intended to provide cross-validation for questions 1 and 2, when reinterpreted as 3a and 4a. the responses to the two pairs of questions appeared to be at odds. While the machine learning enabled software was considered easier to use than the machine learning disabled software, the task performed using the machine learning enabled software was not considered less difficult than the task performed using the machine learning disabled software. However, it is possible that the subjects interpreted questions 3 and 4 as relating directly to the knowledge acquisition task and were able to disassociate the difficulty of performing the task with the tools at hand from the underlying difficulty of the task.

6 Discussion

In an effort to demonstrate that The Knowledge Factory can be used effectively by those with extremely minimal knowledge engineering skills, the subjects in this project had almost no knowledge of knowledge engineering and were given very little training in the use of the software. In view of this extremely limited expertise, it is, in retrospect not very surprising that a large number of subjects failed to appreciate the implications of the Develop New Rules command. This has resulted in a failure to find a statistically significant difference in the accuracies obtained by those treatments in which subjects had access to both machine learning and knowledge acquisition from experts and those in which only the latter was available (although. even so. the mean accuracy in the former group was higher).

It is tempting to interpret the decrease in the number of subjects using the Develop New Rules command in the second scenario as evidence that even the small amount of experience involved in the first scenario was sufficient to improve their use of the software. However, the numbers involved are too small to judge with any confidence whether this decrease in the use of the command was significant or not (a binomial rest fails to reveal a significant difference, but with only 13 observations, its power is very limited).

The subjects were deliberately provided minimal training in the use of the software before the experiment commenced. This was intended to prevent the experimenters from unduly guiding the subjects and hence confounding the results. As a result of this paucity of training, in many cases the software was not used in the manner that the experiment was designed to investigate: integrating machine learning with knowledge acquisition. Where it was used in this manner, however, significantly more accurate rules were obtained in significantly less time.

Another aspect of the study that can be seen to have influenced the outcome was the limited number of variables and limited number of training examples in the base task. It was apparent that a number of subjects in the machine learning disabled treatment were able to use the extensive data analysis facilities in The Knowledge Factory to provide the same effect as the use of machine learning. One of the perceived benefits offered by the machine learning component of the system is its ability to perform exhaustive data analysis (Webb, 1996). Due to the small numbers of variables and training examples, such analysis was also feasible without the use of machine learning. The use of machine learning could thus be expected to be more advantageous for more complex knowledge acquisition tasks.

It should also be acknowledged that the study is constrained by its use of artificial scenarios. It is impossible to accurately evaluate how closely these artificial scenarios mirror real knowledge acquisition tasks.

6.1 Future Research

In view of the issues identified above it is intended that a further study be conducted. This new study will retain most of the design of the current study. However if possible the following differences will be included

- Subjects will be provided with more training in the use of the software.
- More complex data will be employed so as to capture types of task for which the integration of machine learning with knowledge acquisition was expected to deliver maximal benefit.
- Natural tasks will be used.

In the longer term, it would be desirable to map out in detail the types of knowledge acquisition task for which the integration of machine learning with knowledge acquisition from experts is beneficial. While the current study has demonstrated benefit in one context, this provides little evidence about the range of contexts for which it will be beneficial.

There is also much scope for similar evaluation of alternative approaches to the integration of machine learning with knowledge acquisition from experts.

7 Conclusions

Integration of machine learning with knowledge acquisition from experts has considerable intuitive appeal. These two approaches to knowledge acquisition have different and apparently complementary features. However, despite the development of many techniques for integrating the two, there has been little formal evaluation of their effectiveness.

The current study has demonstrated that the integration of machine learning with knowledge acquisition from experts can increase the accuracy of the knowledge bases developed and reduce the knowledge base deve1opment time. The knowledge bases developed through the integrated use of both machine learning and knowledge acquisition from experts were both more accurate than those developed by the isolated use of either machine learning or knowledge acquisition from experts. Questionnaire results indicated a very positive response to the manner in which machine learning was integrated into the software in question.

A number of different techniques for integrating machine learning with knowledge acquisition from experts have been developed. Those examined in this study are distinguished by being oriented for direct use by domain experts with little knowledge engineering expertise. As the experiments employed subjects of this type considerable support has been obtained for the efficacy of these techniques in this context.

Acknowledgments

This research has been supported by the Australian Research Council and the Apple University Development Fund. We are grateful to Tim Menzies for encouraging us to use undergraduate students as experimental subjects for knowledge acquisition research. We thank Zijian Zheng for providing useful comments on previous drafts of this paper.

References

[1] Attar Software (1989). *Structured Decision Tasks Methodology for Developing and Integrating Knowledge Base Systems*. Leigh, Lancashire, Attar Software.

[2] Boose, J. H. (1986). **ETS: A system for the transfer of human expertise**. In Kowalik, J.S. (Ed.), *Knowledge Based Problem Solving.* New York, Prentice-Hall.

[3] Compton, P., Edwards. G., Srinivasan. A., Malor. R., Preston. P., Kang, B. and Lazarus. L. (1992). **Ripple down rules: Turning knowledge acquisition into knowledge maintenance**. *Artificial Intelligence in Medicine*. 4:47-59.

[4] Davis, R. & Lenat, D. B. (1982). *Knowledge-Based Systems in Artificial Intelligence.* New York, McGraw-Hill.

[5] De Raedt, L. (1992). *Interactive Theory Revision*. London. Academic Press.

[6] Le Grand, A. & Sallantin, J. (1994). **A framework to improve knowledge acquisition based on machine learning**. In *ECAI 94: 11th European Conference on Artificial Intelligence*, pages 493-497. John Wiley.

[7] Michalski, R. S. (1984). **A theory and methodology of inductive learning**. In Michalski. R. S., Carbonell, J. G. & Mitchell, T. M. (Eds.), *Machine Learning: An Artificial Intelligence Approach*. Pages 83-129. Berlin. Springer-Verlag.

[8] Morik, K. Wrobel, S., Kietz, J-U. & Emde, W. (1993). *Knowledge Acquisition and Machine Learning: Theory, IV and Applications.* London, Academic Press.

[9] Nedellec, C. & Causse, K. (1992). **Knowledge refinement using knowledge acquisition and machine learning methods**. In *Current Developments in Knowledge Acquisition - EKAW'92*. Pages 171-190. Berlin. Springer-Verlag.

[10] O'Neil, J. L. & Pearson, R. A. (1987). **A development environment for inductive learning systems**. *In Proceedings of the 1987 Australian Joint Artificial Intelligence Conference*. Pages 673-680 Sydney.

[11] Schmalhofer, F. & Tschaitschian, B. (1995). **Cooperative knowledge evolution for complex domains**. In Tecuci, G. & Kodratoff, Y., (Eds.), *Machine Learning and Knowledge Acquisition: Integrated Approaches*. Pages 146-166. London. Academic Press.

[12] Smith. R. G., Winston. H. A., Mitchell. T. M. & Buchanan, B. G. (1985). **Representation and use of explicit justifications for knowledge base refinement**. In Proceedings of the *Ninth International Joint Conference on Artificial Intelligence*. Pages 673-680. San Mateo, Ca. Morgan Kaufmann.

[13] Tecuci, G. (1995). **Building knowledge bases through multistrategy learning and knowledge acquisition**. In Tecuci, G. & Kodratoff, Y., (Eds.), *Machine Learning and Knowledge Acquisition: Integrated Approaches*, pages 13-50. London, Academic Press.

[14] Tecuci, G. & Kodratoff, Y. (1990). **Apprenticeship learning in imperfect domain theories**. In Kodratoff, Y. & Michalski, R., (Eds.), *Machine Learning: An Artificial Intelligence Approach*, pages 514-551. San Mateo, Ca. Morgan Kaufmann.

[15] Webb, G. I. (1993). **DLGref2: Techniques for inductive knowledge refinement**. In *Proceedings of the IJC Workshop W16*. Pages 236-252. Chambery, France.

[16] Webb, G. I. (1996). **Integrating machine learning with knowledge acquisition through direct interaction with domain experts**. *Knowledge-Based Systems*, 9(4):253-266.

[17] Webb, G. I. & Agar. J. W. M. (1992). **Inducing diagnostic rules for glomerular disease with the DLG machine learning algorithm.** *Artificial Intelligence in Medicine*. 4:3-14.

[18] Wilkins, D. C. (1988). **Knowledge base refinement using apprenticeship learning techniques**. In *AAAI-88: Proceedings of the Seventh National Conference on Artificial Intelligence*. Pages 646-651, San Mateo. CA. Morgan Kaufmann.