Title

Comprehensible Knowledge-Discovery in Databases

Permalink

https://escholarship.org/uc/item/68d3r0p6

Journal

Proceedings of the Annual Meeting of the Cognitive Science Society, 19(0)

Authors

Pazzani, Michael J.
Mani, Subramani
Shankle, W. Rodman

Publication Date

1997

Peer reviewed

Comprehensible Knowledge-Discovery in Databases

Michael J. Pazzani (pazzani@ics.uci.edu)
Subramani Mani (mani@ics.uci.edu)
Department of Information and Computer Science

W. Rodman Shankle (rshankle@uci.edu)
Department of Neurology
The University of California
Irvine, CA 92697

Abstract

Large databases are routinely being collected in science, business and medicine. A variety of techniques from statistics, signal processing, pattern recognition, machine learning, and neural networks have been proposed to understand the data by discovering useful categories. However, to date research in data mining has not paid attention to the cognitive factors that make learned categories comprehensible. We show that one factor which influences the comprehensibility of learned models is consistency with existing knowledge and describe a learning algorithm that creates concepts with this goal in mind.

Introduction

Knowledge-discovery in databases is a field whose goal is to extract usable knowledge from a collection of data. It draws upon methods in statistics, signal processing, pattern recognition, information theory, machine learning, and neural networks to produce models that provide insight into data. Such models are expected to be accurate and are further expected to be comprehensible to experts in the field. For example, knowledge acquired through such methods on a medical database might be published in scientific journals. Knowledge acquired from analyzing a financial database might be taught in a management school. While it is important that such knowledge be an accurate summary of the data, it is equally important that the knowledge be comprehensible to experts in the domain. One factor that influences comprehensibility is being integrated with other knowledge in the domain.

In this paper, we concentrate on analyzing a database collected by the Consortium to Establish a Registry for Alzheimer's Disease (CERAD). The particular problem of interest is to identify patients with early signs of dementia. Most demented patients do not see a physician for the problem of memory loss until four years after symptom onset (Ernst and Hay, 1994). Community physicians commonly do not detect dementia or misidentify it in its earliest stages when patients are seeing them for other reasons (O'Connor, et al., 1989). A simple, unobtrusive method for detecting dementia early in the disease's course would encourage patients to seek early evaluation and treatment, resulting in preserved quality of life and reduced financial burden to family and health care providers.

In previous research, we have shown that a variety of machine learning and statistical methods can acquire models that have accuracy, specificity and sensitivity that exceed the average practitioner at screening for early stages of dementia. However, it is unlikely that the description of patients with early dementia created by any of the models so far would be widely adopted in practice. The decision procedure implied by some models (e.g., logistic regression) is too complex to follow, while the decision criteria explicitly stated in learned rules or decision trees make little sense to the neurologist or the practitioner since it differs drastically from the current practice.

In this paper, we concentrate on knowledge discovery from an electronic patient database containing data on the dementia status of each patient and the results of two commonly used cognitive tests for dementia screening, the Blessed Orientation, Memory and Concentration test (BOMC- Fillenbaum et al., 1987) and the Mini-Mental Status Exam (MMSE- Folstein et al., 1975). To understand why the results of current knowledge-discovery algorithms make little sense, it is necessary to describe how the tests are currently used for screening. In each test, the patient answers questions that assess orientation for time and place, registration, attention, short-term recall, language skills, and drawing ability. For example, the patient is first asked to remember a name and address ("John Brown, 42 Market Street, Chicago") and later asked to recall these items. The patient receives a score for each item in the test. For example, the number of times that the test giver repeats the name and address before the patient is able to repeat it immediately is recorded. Similarly, the number of errors in recalling the name and address several minutes later is recorded. An overall score is given to each patient by summing the score on each question. The tests are used in practice for screening for dementia by the use of a simple threshold on the total score.

The score on each question of the test plus the patient's age, sex, and years of education were used in our earlier work to predict whether a patient was "normal" or "mildly impaired" by the knowledge-discovery in database methods.[1] We suspected that such methods would be more effective than a simple threshold on the aggregate score because some questions seemed more important than others. All of the methods were tried, including decision trees, rules, logistic regression, neural networks, and a simple Bayesian classifier more accurate than the simple threshold and none of the methods were substantially more accurate than the others. In such a case, one might prefer to make decisions based upon rules or decision trees since such representations are easy to follow by a practitioner. Furthermore, following guidelines similar to decision trees or rules is becoming commonplace in health care management organizations where a patient first sees a "gatekeeper" who determines whether the patient should be seen by a specialist.

However, neither the trees produced by C4.5 (Quinlan, 1993) nor the rules produced by rule learners such as C4.5 rules or FOCL (Pazzani & Kibler, 1992) produced rules that would be acceptable in practice. In particular, some items which should be viewed as signs of being impaired are used as signs of being normal and vice versa. This does not match the original intent of BOMC and MMSE tests, does not agree with the currently used procedure of totaling the number of errors, and reduces the comprehensibility of the rules to the layperson and the trained neurologist. Table 1 shows an example of one such rule that was produced by FOCL when training on 300 patient records. Similar problems occur with other rule learners such as C4.5 rules and CN2 (Clark & Niblett, 1989).

If such violations of expectations were necessary to obtain accurate results, they could be tolerated. Such violations might even lead to new insights by focusing future research on explaining them. However, we shall show that on this problem, the same diagnostic performance can be achieved without these violations.

In the remainder of this paper, we first describe rule learning algorithms in detail using FOCL as an example to describe one source of incomprehensible rules. We describe a simple extension to FOCL that prevents it from learning rules that violate the expectations of a domain expert and show that this extension does not hurt the diagnostic value of the concepts that are learned. We present preliminary evidence that rules without these violations are preferred. We conclude by describing related work and commenting on directions for future research.

Background: Rule Learners

FOCL is derived from Quinlan's (1991) FOIL system. FOIL is designed to learn a set of clauses that distinguish positive examples of a concept from negative examples. Each clause consists of a conjunction of tests. For example, in the dementia domain a test might check to see whether the patient's age is less than a certain value, or whether the patient knows the day of the week.

Table 1: Sample rule with questionable tests underlined.

```
IF the years of education of the patient is > 5
AND the patient does not know the date
AND the patient does not know the name
    of a nearby street
THEN The patient is NORMAL

OTHERWISE IF the number of repetitions before
             correctly reciting the address is > 2

AND the age of the patient is > 86
THEN The patient is NORMAL

OTHERWISE IF the years of education of the
             patient is > 9
AND the mistakes recalling the address is < 2
THEN The patient is NORMAL

OTHERWISE The patient is IMPAIRED
```

FOIL operates by trying to find a clause that is true of as many positive examples as possible and no (or few) negative examples.[2] It then removes the positive examples explained by that clause from consideration and finds another clause to account for other positive examples. It repeats this clause learning process until all (or nearly all) of the positive examples are explained by some clause. Each clause can be viewed as a description of some subgroup of examples.

To learn a clause, FOIL first considers all possible clauses consisting of a single test. It selects the best of these according to an information-gain heuristic which essentially favors a test that is true of many positive examples and few negative examples. Next, FOIL specializes the rule using the same search procedure and information-based heuristic, considering how conjoining a test to the current clause would improve it by excluding many negative examples and few positives. This specialization process continues until the clause is not true of any negative examples, resulting in a single clause that is a conjunction of tests.

FOCL follows the same procedure as FOIL to learn a set of clauses. However, it learns a set of clauses for each class (such as normal and impaired) enabling it to also deal with problems that have more than two classes. The clause learning algorithm is run once for each class, treating the examples of that class as positive examples and the examples of all other classes as negative examples. This results in a set of clauses for each class.

FOCL has two methods for converting a set of clauses for each class into a single decision list such as that shown in Table 1. The first method simply orders the learned clauses by an estimate of accuracy and uses the most frequent class

[1] There are no severely impaired patients in this sample of data since it is easy to distinguish severely impaired patients from others without the use of such tests.

[2] FOIL uses the minimum description length principle to trade-off the complexity of a rule with the number of examples covered and excluded. This is intended to prevent it from learning an overly complex rule to explain just a few exceptions.

as a default to be used if no clause applies. Using the same examples to learn the initial set of clauses and to create the ordered decision list can cause a problem since the learned rules rarely make errors on the data used to learn the rules. In our experiments, we always divide the training data into a learning set consisting of 2/3 of the training data for learning clauses and an ordering set consisting of the remaining 1/3 of the training data for creating the decision list.

The second method for creating a decision list is an optimization procedure that selects an ordered subset of the original clauses. The algorithm initializes the decision list to a default clause that predicts the most frequent class. Next, it iteratively tries to improve upon the current decision list with an operator that replaces the default rule with a learned clause and a new default clause. The impact is calculated of adding each remaining clause to the end of the current decision list and assigning the examples that match no clause to the most frequent class of the unmatched examples. The change that yields the highest impact in accuracy is made and the process is repeated until no change results in an improvement. Typically, only a few clauses are selected by this process resulting in a relatively short decision procedure.

One further detail is needed to understand how FOCL arrives at a decision list using rule optimization. When adding clauses to the decision list, FOCL also has the option to choose a prefix of a learned clause. That is, if a clause such as X & Y & Z was learned, FOCL considers using X or X&Y in addition to X&Y&Z as a clause in the decision list.[3] This can result in shorter, more general clauses. Such a clause optimization step has been shown to significantly simplify the learned concepts (e.g., Cohen, 1995). The decision list shown in Table 1 was learned using this optimization procedure.

Table 2 shows the accuracy of C4.5, C4.5 rules, FOCL with rules ordered by accuracy, and FOCL with optimized rules on the CERAD data. The accuracy is averaged over 50 trials of dividing the data into a training set of size 210 and a test set of size 105. The test set does not contain any examples from the training set.

Table 2: Accuracy at identifying impaired patients.

Algorithm	Accuracy
C4.5	86.7
C4.5 rules	82.6
FOCL (Accuracy order)	86.0
FOCL (Optimized order)	90.6

The results show that FOCL's optimized rule orde substantially more accurate than the other learr algorithms. This result is significant at least at the 01 lc using two-tailed t-tests. We now turn our attentior improving the comprehensibility of FOCL's output.

Monotonicity Relationships

Some clauses in the learned category descriptions vio the intent of the BOMC and MMSE examinations. particular, getting some questions right is used as evide that one is impaired and getting some questions wron used as evidence that one is not impaired. A relati simple change to FOCL eliminates such tests fr consideration. For variables with numeric relationships, user declares whether the variable has a known monotc relationships with each class.[4] A monotonic relationshi one in which increasing the value of the variable alw increases or decreases the likelihood category membersl When considering tests to add to a clause, the tests violate these relationships are removed from considerati For example, when learning a description of the nor patients, FOCL with monotonicity constraints only check see if the number of errors recalling the address is less t some number. When learning clauses describing impaired category, it only tests to see if this variable above some threshold.

These constraints on tests may also be used on Bool and nominal variables. In this case, the user specifies wh values are possibly indicative of membership in a class. example, a value of true for the variable "knows the da may be used as a sign for normal, while the value false r be used as a sign for impaired.

For the CERAD data, and for many medical data s the data is coded such that an increase in a variables valu an incorrect response to a question increases the chance one has a particular disease or syndrome. We encoded knowledge as monotonicity relationships to FOCL. We a added constraints indicating that the likelihood that one impaired increases with age and decreases with educatic level. Table 3 shows an example of a rule learned with th constraints.

[3] Prefixes of learned clauses are selected rather than subsets for efficiency reasons. There are only N prefixes of a clause of length N, while there are 2^N subsets. The tests at the end of a clause are more likely to cause problems since they are learned after more examples have been excluded, increasing the chances of "coincidental" regularities.

[4] Some variables may be left unconstrained, in which case tests with that variable are considered.

We ran 50 trials of FOCL using rule optimization with and without monotonicity constraints. There is not a substantial or significant difference in accuracy using the constraints. FOCL is 90.7% accurate when using monotonicity constraint and 90.6% accurate when unconstrained.[5] On average, a decision list formed without constraints contains a total of 4.65 tests and 2.13 violations of the monotonicity constraints. With the constraints, an average of 4.30 tests are used in a decision list, none of which violate the constraints. This raises two questions that we will address below:

1. Why does a learning algorithm create tests that violate these constraints?
2. Are rules that do not violate monotonicity constraints to be preferred in practice?

If we assume that the constraints are correct, then there are two factors which contribute to a test that violate these constraints being used in a rule. First, while the test appeared best according to an information-based selection procedure, this procedure detected a "spurious correlation" in the data due to sampling biases, noise in category label (i.e., a patient may be misdiagnosed) or noise in a variable's value (i.e., a question may have been recorded or scored improperly or a patient may have guessed the correct answer to a question such as guessing the day of the week). Such problems are more likely to occur near the end of a clause or the leaves of a decision tree. In these cases, the sample of data used by the information-based heuristic is reduced to those examples that are true of the conditions in the initial portion of the clause. A smaller sample is more likely to have a test that is uninformative appear to be informative. Pruning algorithms such as the FOIL's MDL method, FOCL's rule optimization procedure, and various decision tree pruning algorithms mitigate this problem. However, they are only a partial solution at best. For example, the rules produced by accuracy ordering in unconstrained FOCL contain an average of 18.63 tests and 12.13 monotonicity constraint violations. The rule optimization procedure reduces this to 4.65 tests and 2.13 violations.

The second factor that accounts for the selection of tests that violate the monotonicity constraints is that the selection procedure selects a single best test. It is often the case that several tests are equally or statistically indistinguishably informative. Under these circumstances, a decision procedure could be found that is both accurate and comprehensible to an expert by eliminating from consideration tests that violate these constraints.

Note that the goal of knowledge-discovery in databases is sometimes viewed as finding "the model" of the data, while in reality there are often many possible models of the data that are not significantly different according to any statistical procedure on the training example. For example, Murphy and Pazzani (1994) used a massively parallel computer to find all decision trees consistent with a set of 20 training examples. A total of over 25,000 trees were found. Many of these trees were very complex. However, on average there were 20 trees with 5 or fewer tests. We advocate imposing other constraints, such as monotonicity constraints on the model selection process so that accurate and comprehensible models are produced.

Table 3: A rule learned with monotonicity constraints.

```
IF the years of education of the patient is > 5
AND the mistakes recalling the address is < 2
THEN The patient is NORMAL

OTHERWISE
IF the years of education of the patient is > 11
AND the errors made saying the months backward
    is < 2
THEN The patient is NORMAL

OTHERWISE
IF the years of education of the patient is > 17
THEN The patient is NORMAL

OTHERWISE The patient is IMPAIRED
```

Monotonicity Constraints and the Adoption of Clinical Guidelines

We have conducted surveys of two neurologists to determine whether monotonicity constraints influence the willingness to follow guidelines. We generated 16 decision lists such as that shown in Table 1 by using unconstrained FOCL and 16 decision lists such as that shown in Table 3 by using FOCL with monotonicity constraints on 16 randomly selected subsets containing 200 examples from the CERAD database. In both cases, the rule optimization procedure of FOCL was used to ensure that concise descriptions were learned. Each rule was printed on a separate sheet of paper and presented in a random order to each neurologist. We asked each neurologist to rate on a scale of 0-10 "How willing would you be to follow the decision rule in screening for cognitively impaired patients". We hypothesized that the neurologists would be more willing to use rules that were generated by FOCL when it used monotonicity constraints.

Neurologist 1 has been involved in this project for approximately one year and is aware that the focus of the research is to create comprehensible rules. Neurologist 2 is not affiliated with this project and is unaware of its goals. For Neurologist 1, the average score of rules generated by FOCL without the monotonicity constraints was 3.25, while the average score of rules generated with the monotonicity constraints was significantly higher 5.56 $t(15) = 6.60$, $p < .001$. For Neurologist 2, these scores were 0.25 and 2.38 $t(15) = 5.09$, $p < .001$. Although it is clear that the neurologists were using different scales, in each case higher average ratings were given to the category descriptions generated with these constraints in mind. We also show the correlation between the number of monotonicity constraint violations and the willingness to follow the rule. Table 4

[5] We also inverted all of the monotonicity constraints (e.g., by stating that increasing age decreases the likelihood of dementia) and this significantly decreased the accuracy of FOCL to 88.1%.

shows the correlation between these variables for each neurologist. For comparison purposes, we also show the correlation between the willingness to follow a rule and the number of tests and number of clauses in the rule, two commonly used measures of rule complexity. We did not attempt to balance for the size of the rules in the two conditions, but the average number of tests and clauses was within 10% between the two conditions.

Table 4: Correlations between properties of learned models and neurologists' willingness to use the models.

Correlation	Neurologist 1	Neurologist 2
Violations	.433	.623
Number of tests	.208	.020
Number of clauses	.278	.011

These results show that both neurologists were sensitive to the violations of monotonicity constraints and these violations affect the willingness to follow the rule. The size of the rules did affect the judgment of one of the neurologists but to a lesser extent than the number of constraint violations.

Related Work

Most work in producing understandable rules has focused on syntactic properties of the rules, particularly the size of rules. Such work simply equates size with comprehensibility and seeks to minimize the size of learned relationships. For example, Karalic's (1996) paper, "Producing more comprehensible models while retaining their performance" might just as well be entitled "Producing smaller models while retaining their performance" since it describes the use of the minimum decription length principle to learn shorter rules. Craven's (1996) research on extracting comprehensible models from neural networks has focused on creating concise representations such as short rules. In contrast, we have focused on how the relationship between learned knowledge and existing knowledge affects comprehensibility and have shown that there are differences other than size that affect the willingness of experts to use rules.

Pazzani (1991) introduced the notion of influence theories that affect the causal induction process. This earlier research focused on the direct causes of a state change, showing that some aspects of the causal induction process could be explained by the fact that people have knowledge of potential causes but must learn which combinations of these causes are necessary and sufficient (cf. Kelley, 1971). While the current work differs in that the factors of interest are effects rather than causes of the phenomenon of interest, the general idea is similar in that the learning algorithm is constrained by knowledge of potential influences.

Clark and Matwin (1993) show how learning may be constrained by a qualitative model of a physical phenomenon. Although it would be difficult to represent knowledge of the causes of dementia as a qualitative model, Clark and Matwin's work does show that rule learning can be constrained to be consistent with prior knowledge.

A variety of techniques to constrain the coefficients of linear regression are summarized in Leblanc and Tibshirani (1993). Although the focus of the research on methods to regularize weights is to increase the predictive inference capabilities of the models, they may also make the models easier to interpret. For example, constraining the coefficients of each variable to be positive or negative could simulate the effect of declaring there to be a monotonic relationship between a variable and a class.

Future Directions

Our future plans include collecting feedback from additional practitioners who use the MMSE and BOMC on the comprehensibility of learned category descriptions in this domain and further psychological investigation on the factors that influence the willingness of experts to use learned models.

In the current implementation, there is no way to add a condition that violates a monotonicity constraint to a category description regardless of the amount of statistical support for that condition. In the future we plan on softening such constraints by preferring conditions that do not violate constraints but permitting a condition with a violation if there are no sufficiently informative conditions that do not violate constraints. We will also investigate methods for learning monotonicity constraints so that the knowledge required by this system may automatically be acquired from the data.

Conclusions

We have argued that to be truly useful, the knowledge discovered in databases must both be accurate and comprehensible. We have further argued that one factor that influences the comprehensibility of learned knowledge is the use of conditions as evidence for belonging to some category when prior knowledge indicates that these conditions are evidence that an example does not belong to that category. We have shown that existing knowledge discovery systems learn rules with such conditions and created an enhancement to one algorithm that prevents these conditions from being added to learned models. Finally, we have presented preliminary evidence that experts prefer rules that do not contain violations of prior knowledge.

Acknowledgements

The authors gratefully acknowledge CERAD for collecting and disseminating the database used in this study. This research was funded in part by the Alzheimer's Association Pilot Research Grant, PRG-95-161 and the National Science Foundation grant IRI-9310413. Comments by Dennis Kibler and Dorrit Billman on an earlier draft of this paper help to clarify some issues and their presentation.

References

Clark, P. & Matwin, S. (1993). "Using Qualitative Models to Guide Inductive Learning". The Proceedings of the *10th International Conference on Machine Learning*. Amherst, MA, 49-56.

Clark, P. & Niblett, T. (1989). The CN2 Induction Algorithm. *Machine Learning, 3*, 261-284.

Craven, M. W. (1996). Extracting Comprehensible Models from Trained Neural Networks. Ph.D. thesis, Department of Computer Sciences, University of Wisconsin-Madison. (Also appears as UW Technical Report CS-TR-96-1326).

Cohen, W. (1995). Fast effective rule induction. In Proceedings of the Twelfth International Conference on Machine Learning, Lake Tahoe, California.

Duda, R. & Hart, P. (1973). *Pattern classification and scene analysis.* New York: John Wiley & Sons.

Ernst, R. and Hay, J. (1994). The US economic and social costs of Alzheimer's disease revisited. *American Journal of Public Health,* 84(8):1261-4.

Fillenbaum, G., Heyman, A., Wilkinson, W., and Haynes, C. (1987). Comparison of two screening tests in Alzheimer's disease: the correlation and reliability of the mini-mental state examination and the modified Blessed test. *Archives of Neurology,* 44(9):924-7.

Folstein, M., Folstein, S., and McHugh, P. (1975). Mini-mental state-a practical method for grading the cognitive state of patients for the clinician. *Journal of Psychiatric Research, 12*(3):189-98.

Karalic, A. (1996). Producing More Comprehensible Models While Retaining Their Performance, Information, Statistics and Induction in Science. Melbourne, Australia.

Kelley, H. (1971). Causal schemata and the attribution process. In E. Jones, D. Kanouse, H. Kelley, N. Nisbett, S. Valins, & B. Weiner (Eds.), *Attribution: Perceiving the causes of behavior* (pp 151-174). Morristown, NJ: General Learning Press.

Leblanc, M. & Tibshirani, R. (1993). Combining estimates in regression and classification. Dept. of Statistics, University of Toronto, TR.

Murphy, P. & Pazzani, M. (1994). Exploring the Decision Forest: An Empirical Investigation of Occam's Razor in Decision Tree Induction. *Journal of Artificial Intelligence Research,* 1, (pp. 257-275).

O'Connor, D., Pollitt, P., Treasure, F., Brook, C., and Reiss, B. (1989). The influence of education, social class and sex on mini-mental state scores. *Psychological Medicine,* 19:771-776.

Pazzani, M. (1991). The influence of prior knowledge on concept acquisition: Experimental and computational results. *Journal of Experimental Psychology: Learning, Memory & Cognition, 17, 3,* 416-32.

Pazzani, M. and Kibler, D. (1992). The utility of knowledge in inductive learning, (9):57-94.

Quinlan, J.R. (1987). Simplifying decision trees. *International Journal of Man-Machine Studies, 27,* 221-234.

Quinlan, J.R. (1990). Learning logical definitions from relations. *Machine Learning, 5,* 239–266.

Quinlan, J. (1993). *C4.5: Programs for Machine Learning.* Morgan Kaufmann, Los Altos, California

Shankle, W.R., Mani, S., Pazzani, M., & Smyth, P. (1997). *Detecting very early stages of dementia from normal aging with machine learning methods.* The Proceedings of the 6th Conference on Artificial Intelligence in European Medicine.