Induction of Condensed Determinations

Pat Langley*(LANGLEY@CS.STANFORD.EDU)
Robotics Laboratory, Computer Science Dept.
Stanford University, Stanford, CA 94305

Abstract

In this paper we suggest *determinations* as a representation of knowledge that should be easy to understand. We briefly review determinations, which can be displayed in a tabular format, and their use in prediction, which involves a simple matching process. We describe CONDET, an algorithm that uses feature selection to construct determinations from training data, augmented by a condensation process that collapses rows to produce simpler structures. We report experiments that show condensation reduces complexity with no loss of accuracy, then discuss CONDET's relation to other work and outline directions for future studies.

Introduction

Understandability is a major concern in knowledge discovery and data mining. Although it is important to discover knowledge that is accurate, in many domains it is also essential that users find that knowledge easy to interpret. Most researchers assume that logical rules and decision trees are more understandable than other formalisms, such as neural networks or stored cases. Although the evidence supporting this belief is mainly anecdotal, we will not argue with it here.

Rather, we will assume its validity and focus on a special class of logical rules, known as *determinations*, that we maintain are particularly understandable. This representation differs from other rule frameworks in that all rules in the knowledge base refer to the same attributes. As a result, they can be graphically displayed as a 'truth table', with one column for each attribute (including the class) and one row for each combination of attribute values. We anticipate that users will like this regular structure, especially given its similarity to widely used spreadsheet formats.

In the following sections, we review the representation of determinations and their use in classification, followed by an algorithm for inducing these structures based on recent work in feature selection. Next we present a technique for condensing induced determinations, aimed at further improving their understandability. After this, we present experimental studies of these techniques that evaluate the accuracy and complexity of the learned structures. We close with comments on related work and directions for future research.

*Also affiliated with the Institute for the Study of Learning and Expertise, 2164 Staunton Court, Palo Alto, CA 94306.

The Nature of Determinations

Davies and Russell (1987) introduced determinations as a form of background knowledge for use in analogical reasoning, but the idea has more general applications. Briefly, a determination expresses some functional dependency between a set of predictor attributes P and a set of predicted attributes Q, so that, given P, one can infer Q. Of course, such knowledge is useful only if one has information about particular combinations of those attributes' values. Davies and Russell proposed obtaining this information through analogy with stored cases. However, one can also envision a knowledge base containing a separate rule for each combination of predictor values, and we will assume such structures here.

Such determinations are interesting from the perspective of understandability because they can be displayed in a tabular format. Table 1 shows a determination for a simple artificial domain, originally used by Quinlan (1993) to illustrate decision trees, that involves deciding whether to pursue an outdoor activity. This domain includes four predictor attributes – OUTLOOK, HUMIDITY, WINDY, and TEMPERATURE – and one predicted attribute CLASS, which states whether to engage in the activity. This determination includes columns for only three of the predictor variables, because TEMPERATURE does not help to predict CLASS.

One can use a determination for prediction or inference in the same way as any other formalism that involves logical rules. For a given instance, one finds the row (i.e., rule) that specifies a combination of predictor values that match the instance, then infers the value specified for the predicted attribute(s). For now, we will assume that all attributes in a determination are discrete, and that any continuous variables have been transformed into discrete ones either by the knowledge base's developer or through some automatic process.

Although Davies and Russell focused on logical determinations that always held, one can adapt them to

Table 1: A simple determination for outdoor activities.

Outlook	Humidity	Windy	Class
Sunny	High	True	No
Sunny	High	False	No
Sunny	Normal	True	Yes
Sunny	Normal	False	Yes
Overcast	High	True	Yes
Overcast	High	False	Yes
Overcast	Normal	True	Yes
Overcast	Normal	False	Yes
Rain	High	True	No
Rain	High	False	Yes
Rain	Normal	True	No
Rain	Normal	False	Yes

situations in which each row's outcome is probabilistic. In such domains, the natural strategy is to predict the class most frequently associated with the matched row. Note that learned determinations may lack rows for certain combinations of attribute values if those combinations never occur in the training data. For such situations, Langley and Sage (1994) recommended basing predictions on the nearest matches, while Kohavi (1995) suggested predicting a default value associated with the entire table. We will incorporate the latter technique into the system we describe here.

The ability to display determinations in a tabular format has led Kohavi (1995) to refer to them as *decision tables*. Determinations are also equivalent to what Langley and Sage (1994) have called *oblivious decision trees*, in which each level of the tree involves tests on the same attribute. We will continue to use the term *determinations* here, primarily because it should be familiar to more readers.

Greedy Induction of Determinations

Given a set of predictive attributes, inducing a probabilistic determination from supervised training data is straightforward. For each observed combination of predictive values, one computes a histogram for the class values, then selects the most frequent class for entry in that row of the table. To determine default values, one also computes histograms for the entire training set, then selects the most frequent overall value.

However, the above procedure assumes that the predictive and predicted attributes have been specified. Many data-mining tasks involve supervised learning, so that one knows which attribute must be predicted, but determining the predictive attributes is another matter. Fortunately, some recent work on feature selection has dealt with determinations or closely related representations of knowledge.

For example, Schlimmer (1993) describes a systematic search algorithm that finds all minimal sets of features that predict the training data, though his method was not well suited for noisy data. Other work has also dealt with this task under different guises. Thus, Langley and Sage (1994) report a greedy algorithm that induces oblivious decision trees, Aha and Bankert (1994) take a similar approach to finding abstract cases for nearest-neighbor classification, and Kohavi (1995) describes a related scheme for creating decision tables.

Table 2: A condensed determination based on Table 1.

Outlook	Humidity	Windy	Class
Sunny	High	*	No
Sunny	Normal	*	Yes
Overcast	*	*	Yes
Rain	*	True	No
Rain	*	False	Yes

We have developed a system for learning determinations that operates along similar lines, which we will call ConDet. As Langley and Sage (1994) note, methods for feature selection must take a stance on four basic issues. First, they must specify the state from which search begins; ConDet starts with no features, since we believe that a bias toward simplicity will produce more understandable structures. Second, they require some scheme for organizing search; our system takes a greedy approach, both for purposes of efficiency and to reduce chances of overfitting. Third, they must have some means of evaluating alternative feature sets; ConDet takes a 'wrapper' approach to evaluation (John, Kohavi, & Pfleger, 1994), which invokes the histogram method described above for each candidate feature set considered, combined with an efficient version of leave-one-out to estimate its accuracy. Finally, they must indicate some halting criterion, and our system stops adding features when none of the candidates leads to an increase in the estimated accuracy.

Clearly, our approach to feature selection is far from new, in that ConDet draws heavily on earlier work. We review it here for purposes of completeness rather than novelty. This paper's main goal is to explore the advantages of learning determinations from the viewpoint of finding *understandable* knowledge structures. Again, we posit that determinations, especially when presented in tabular form, should fare better on this dimension than decision trees or arbitrary rule sets. However, this does not mean that their understandability cannot be improved further, as we will find shortly.

Condensing Induced Determinations

As we have seen, ConDet uses a feature-selection method, combined with a simple counting scheme, to induce a determination from data. This formalism has the same representational power as decision trees and arbitrary rule sets, but it may require more rules to encode the same knowledge, and this complexity may

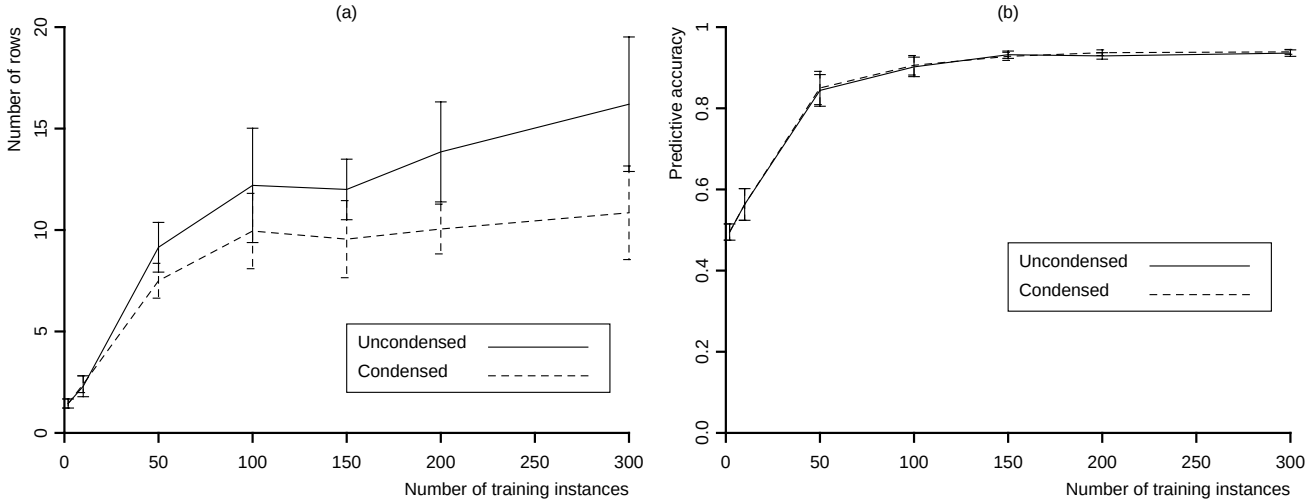

Figure 1: Learning curves for for inducing determinations on chess endgames, with and without its condensation mechanism, measuring (a) complexity of the learned determinations and (b) accuracy on separate test sets.

decrease the inherent comprehensibility. For example, Quinlan's (1986) decision-tree encoding of the twelve-row determination in Table 1 involves only five terminal nodes, which is certainly simpler in some respects.

Fortunately, there exists a compromise that retains the tabular format but allows simpler structures. These *condensed* determinations still display knowledge in terms of rows and columns, but they allow wildcard symbols to indicate that some rows have been collapsed. For example, Table 2 shows a condensed determination that makes the same predictions as the original table. The new structure includes a wildcard '$*$' for selected values of HUMIDITY and WINDY, which reduces the total number of rows from twelve to five.

CONDET incorporates a mechanism to condense determinations in this manner. The basic operator involves combining rules (rows) that differ on only one predictive attribute into a rule in which that attribute's values have been replaced with a wildcard. For tractability's sake, we restrict this operation in certain ways. Rather than focusing on pairs of rules, CONDET combines *all* rules that share a set of attribute values. Also, when the system combines one set of rules that share values, it tries to condense all other sets that have common values on those attributes.

Another constraint aims to maintain the predictive accuracy of the original determination. Here, CONDET combines only sets of rules that predict the same class. When all possible rows of the determination are represented in the training data, this scheme does not alter the deductive closure of the knowledge base. However, the closure can change when some rows are missing, since the situation they describe may now be covered by the condensed rule, which has precedence over the majority class. For this reason, CONDET evaluates each candidate condensation against the training set, retaining it only if it does not hurt the overall accuracy.

In terms of search organization, CONDET takes a greedy approach to condensing its determinations, as it does in constructing them. The system tentatively generates a new table that results from condensing along each attribute, in each case combining all rules that differ on that attribute but have the same class. It selects the condensed table with the highest training set accuracy and continues this process, halting when accuracy decreases. The resulting table may not be condensed in the optimal way, but it provides a reasonable compromise given limited computational resources.

Experiments with Condensation

Our aim in developing CONDET was to improve the comprehensibility of learned determinations without decreasing their accuracy. We posit that determinations with fewer rows will be easier to understand than ones with more rows; we have no hard evidence for this claim, but it seems intuitively plausible and we will assume it here. Thus, to evaluate our system's behavior, we needed two dependent measures – the accuracy of the induced determinations and the complexity (specifically, the number of rows) of this knowledge structure.

We tested CONDET on four domains from the UCI repository, focusing on data sets with only nominal attributes. For each domain, we generated 20 random training sets and 20 associated test sets. We ran CONDET on all 20 training sets, measuring accuracy on the test sets and complexity of the learned determination, then computed average scores. Because we were interested in the effects of condensation, we collected the same statistics when this process was absent.

Moreover, we hypothesized that differences between the condensed and uncondensed determinations would increase with greater numbers of training cases, because the data would tend to encourage the inclusion of more attributes and thus increase the number of un-

compressed rows. For this reason, we collected learning curves, which measure system behavior as one increases the number of training cases. We expected that their accuracies would remain the same throughout the course of learning, while their complexities would diverge for larger numbers of training instances.

Figure 1 shows the comparative learning curves for the domain of chess endgames, which involves two classes and 36 attributes. The results were consistent with our predictions; Figure 1 (a) indicates that, later in the learning curve, condensation consistently leads to simpler determinations, whereas Figure 1 (b) reveals that this process does not reduce accuracy. We observed similar results (which we do not have room to report here) on domains involving mushroom classification and Congressional voting records, with condensation not affecting accuracy but simplifying the determinations. Here the size reduction was smaller, since feature selection left only a few tabular rows to condense. We also tested our algorithm on DNA promoters, a domain that typically gives trouble to induction methods that create axis-parallel splits. Yet the predicted effect occurred even here; condensation led to simpler determinations without reducing accuracy.

Recall that our experiments were not designed to show that ConDet is a particularly effective induction algorithm. Other implementations of the same basic approach may produce simpler determinations and higher accuracies, though the accuracies for ConDet and C4.5 were nearly identical on the domains used in our studies. Rather, our aim was to illustrate that determinations are a viable representation for use in knowledge discovery, that feature selection combined with a simple counting procedure can produce accurate determinations for some natural domains, and that a straightforward condensation process can simplify (and make more understandable) these knowledge structures with no loss in accuracy.

Related and Future Work

The approach to induction described in this paper has clear relations to earlier research. We have noted its strong debt to work on feature selection; nor are we the first to study methods for learning determinations from data, as Schlimmer (1993), Langley and Sage (1994), and Kohavi (1995) have worked on very similar tasks and, in some cases, very similar methods.

At first glance, the condensation process appears more novel, but it holds features in common with compression techniques intended to reduce matching costs and with postpruning methods designed to avoid overfitting. An even stronger connection exists with work in the rough sets community, which often uses tabular representations of knowledge. Shan, Ziarko, Hamilton, and Cercone (1995) report an operation called *value reduction* that reduces the rows in a table by replacing values with wildcards. Their algorithm differs from the one used in ConDet, but the spirit is much the same.

We have made no claims that our particular approaches to the induction and simplification of determinations are the best possible. Rather, this paper's contribution has been to highlight determinations as a promising representation of discovered knowledge, to note that algorithms exist for inducing such descriptions, and to show there are methods that can increase their understandability with no loss in accuracy.

We believe that the most important direction for future work on this topic lies not in developing more refined algorithms, but in testing our predictions about the ease of understanding condensed determinations relative to other formalisms. This will require experiments with human subjects, including measures of their ability to understand knowledge bases, before we can draw firm conclusions about alternative notations.

Acknowledgements

Thanks to S. Sage, R. Kohavi, and G. John for discussions that led to the ideas in this paper. This research was funded by AFOSR Grant No. F49620-94-1-0118.

References

Aha, D. W., & Bankert, R. L. (1994). Feature selection for case-based classification of cloud types. *Working Notes of the AAAI94 Workshop on Case-Based Reasoning* (pp. 106–112). Seattle, WA.

Davies, T. R., & Russell, S. J. (1987). A logical approach to reasoning by analogy. *Proceedings of the Tenth International Joint Conference on Artificial Intelligence* (pp.264–270).Milan: Morgan Kaufmann.

John, G. H., Kohavi, R., & Pfleger, K. (1994). Irrelevant features and the subset selection problem. *Proceedings of the Eleventh International Conference on Machine Learning* (pp. 121–129). New Brunswick, NJ: Morgan Kaufmann.

Kohavi, R. (1994). The power of decision tables. *Proceedings of the 1995 European Conference on Machine Learning* (pp. 174–189). Heraklion, Crete.

Langley, P., & Sage, S. (1994). Oblivious decision trees and abstract cases. *Working Notes of the AAAI94 Workshop on Case-Based Reasoning* (pp. 113–117). Seattle, WA: AAAI Press.

Quinlan, J. R. (1993). *C4.5: Programs for machine learning*. San Francisco: Morgan Kaufmann.

Schlimmer, J. C. (1993). Efficiently inducing determinations: A complete and efficient search algorithm that uses optimal pruning. *Proceedings of the Tenth International Conference on Machine Learning* (pp. 284–290). Amherst, MA: Morgan Kaufmann.

Shan, N., Ziarko, W., Hamilton, H. J., & Cercone, N. (1995). Using rough sets as tools for knowledge discovery. *Proceedings of the First International Conference on Knowledge Discovery and Data Mining* (pp. 263–268). Montreal: Morgan Kaufmann.