Increasing the Efficiency of Data Mining Algorithms with Breadth-First Marker Propagation

John M. Aronis
Department of Computer Science
University of Pittsburgh
Pittsburgh, PA 15260
aronis@cs.pitt.edu

Foster J. Provost
NYNEX Science and Technology
400 Westchester Avenue
White Plains, NY 10604
foster@nynexst.com

Abstract

This paper describes how to increase the efficiency of inductive data mining algorithms by replacing the central matching operation with a marker propagation technique. Breadth-first marker propagation is most beneficial when the data are linked to hierarchical background knowledge (e.g., tree-structured attributes), or when the attributes describing the data have many values. We support our claims analytically with complexity arguments and empirically on several large data sets. We also point out other efficiency gains, including reduced memory management overhead, which facilitate mining massive tape archives.

Introduction

Inductive algorithms have proven to be valuable, practical tools for automated discovery in science and business, but users run into difficulties applying the algorithms to large, complex problems. For example, a large data set may have thousands of values for a location field (e.g., zip). Unfortunately, most existing algorithms are prohibitively inefficient when it comes to large value sets. One may also want to group these specific locations based on hierarchical knowledge, e.g., zip → city → state. Existing algorithms are also inefficient when it comes to even the most basic hierarchical background knowledge.

This paper is aimed at algorithm designers and implementors, and discusses how to increase the efficiency of these algorithms so that they will scale up to larger and more complex problems. We recommend the replacement of the central matching operation with *breadth-first marker propagation.* This techique is particularly effective when mining data described by attributes with large value sets or data linked to hierarchical background knowledge.

Several prior approaches implement or discuss the use of hierarchical background knowledge for propositional learning algorithms. The RL rule learning system (Clearwater and Provost 1990) extends the standard feature-vector-based system by allowing the possible values of attributes to be structured in ISA hierarchies. Núñez (1991) describes how ISA hierarchies can be used for decision tree learning, and Quinlan (1993) lists support of tree-structured attributes as a "desirable extension" to C4.5. He describes a scheme for encoding taxonomic information into flat attribute-value tables that standard inductive learning programs can use. The system described by Almuallim, et al., (1995), which we discuss in detail later, uses ISA hierarchies directly, and is shown to be more efficient than the techniques suggested by Quinlan.

Breadth-first marker propagation replaces, with a single pass through the data, the time-consuming generate-and-match operation common in many inductive algorithms, in which many hypothesis specializations are matched against the data one by one. We now describe breadth-first marker propagation, compare its efficiency analytically with standard approaches, and give an empirical demonstration on three very large, complex, real-world data mining problems.

Breadth-first marker propagation

Many inductive data mining algorithms, decision tree learners, and rule learners, in particular, build models through the iterative refinement of hypotheses. The fundamental operation is the specialization of a hypothesis by adding conjuncts, viz., attribute-value pairs, and counting the matches of the resulting specializations against the training database. These counts are used as input to a comparative evaluation function. We recommend the replacement of this generate-and-match method with a single operation based on breadth-first marker propagation to generate counts for all of a rule's specializations in one pass through the data. This *breadth-first marker propagation* approach is applicable to hierarchically structured attribute value sets, as well as to standard, flat attribute value sets.

Training examples are typically viewed as vectors of attribute-value pairs to be matched against. Instead, consider them to be vectors of bidirectional pointers into the value space. Given that you want to specialize a k-conjunct hypothesis R (e.g., a rule or a decision-tree branch), breadth-first marker propagation generates counts of matches for all possible specializations as follows. The data structure VALUESET contains the set of values with non-zero counts, which will be used as indices to retrieve the counts.

1. For each conjunct of R, mark the corresponding value with a *conjunct mark* (which we will denote &).
2. Following pointers, propagate these marks to the training instances, tallying how many marks accumulate on each instance.
3. For those instances with k conjunct marks, i.e., those that satisfy all k conjuncts of R, mark the instance with its class (e.g., + or −).
4. Now, for each instance, and for each attribute, propagate the instance's class mark to the attribute value present in the instance. At each attribute value, keep a running tally of the number of marks of each type. Add to VALUESET a pointer to each value marked.
5. For hierarchies of values, propagate *tallies of marks* in a breadth-first fashion from the leaves of the hierarchy to the root. Parent tallies are the sums of the corresponding child tallies. Add to VALUESET a pointer to each value visited.

Shape	Color	Class
Square	Red	−
Square	Blue	+
Triangle	Blue	+
Round	Blue	−

Figure 1: A Simple Data Mining Problem.

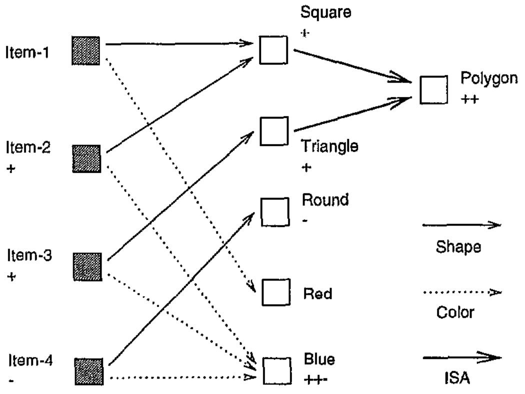

Figure 2: Network Representation of Simple Problem.

We will illustrate the algorithm on a simple problem. Consider the database given in Figure 1, corresponding to the network of pointers shown in Figure 2. Suppose the learner wants to specialize the hypothesis color=blue → +. We first mark blue with &, then move that marker down links onto items 2, 3, and 4. Since each of these items now has one & marker, corresponding to the single conjunct of the current hypothessis, we mark each item with its class (+ or −). Then, these markers are moved forward across links and tallied on each node. (This is the state the diagram illustrates.) Notice that the node Polygon accumulates two + markers and no − markers, indicating a perfect match of the positive examples.

Complexity Analysis

We first consider the complexity of hypothesis specialization in the case without ISA hierarchies. Assuming that there are e examples, a attributes, and (on average) v values for each attribute, even very efficient inductive algorithms based on matching require $O(e)$ matches for each of $O(av)$ potential specializations of each hypothesis for a time complexity of $O(eav)$, as described recently by Domingos (Domingos 1996).

Now consider a learner that uses breadth-first marker propagation to replace matching. After walking through the examples once, each of the possible specializations will have class counts tallying all the examples that match it. The counts can be retrieved by walking through VALUESET, which (with no value hierarchies) can have no more than ae elements. The overall time complexity, $O(ae)$, is independent of the number of values. Thus, marker propagation should scale better for problems with large sets of values.

Now consider the case where attributes can have hierarchical, tree-structured values. The state of the art in efficient learning with value hierarchies is described by Almuallim, et al. (1995). They show their method, which we will call the *AAK-direct* approach, to be more efficient than other methods. It differs from our marker propagation technique in that it walks each attribute value up the ISA hierarchy individually. With ISA hierarchies of depth d, computing counts for e examples and a attributes takes time $O(ead)$.

Because breadth-first marker propagation combines counts at each level and propagates *tallies* of markers up ISA hierarchies, the process takes $O(ea + s)$ time, where s is the total number of values visited. It is clear that the set of values visited by breadth-first marker propagation is the same as the set values visited by the AAK-direct approach. However, breadth-first marker propagation visits each value only once. Thus, in the worst case, where no two examples share a value, and no two values share intermediate tree nodes as ancestors, marker propagation is equivalent to the AAK-direct approach. In any non-degenerate case, where there exists at least one place in the visited ISA hierarchy where its branching factor is greater than one, marker propagation will be more efficient than the AAK-direct approach.

Moreover, for very large datasets, breadth-first marker propagation introduces efficiency benefits that are not apparent from the complexity analysis alone. Consider, again, the hypothesis specialization step. For a attributes and v values, matching methods typically make av passes through the set of e data items. Even more savvy programs, e.g., C4.5 (Quinlan 1993), make a passes through the set of e examples. Breadth-first marker propagation performs only one pass through the data, performing a operations on each item. This introduces a huge savings in disk accesses if the data set does not fit in main memory. For example, an n-level decision tree can be built with only n passes through the example set.

Empirical Demonstrations

To demonstrate the analytical results, we replaced matching with breadth-first marker propagation in the RL (Clearwater and Provost 1990) rule-learning algorithm (BFMP-RL). For these studies we use a beam search and a depth limit to restrict the search space of rules, use the rule certainty factor defined by Quinlan (1993) to evaluate potential rules, and accept rules if their evaluation is above a user-defined threshold.

To test our first analytical result that, even without hierarchical background knowledge, breadth-first marker propagation is more efficient than conventional matching as the number of attribute values grows, we synthesized a sequence of problems consisting of 10,000 training examples with 10 attributes and an increasing number of values randomly assigned to these attributes. These tests were performed on a DECstation 5000 with 64Mbytes of memory.

Figure 3 compares BFMP-RL's run time with that of RL using generate-and-test matching (MATCHING-RL). Note that for these and the following experiments, the different systems performed identical searches and produced identical rule sets. As predicated analytically, the run time with breadth-first marker propagation remains nearly constant as the number of values increases, while the run time with matching increases linearly.

Our second analytical result predicts that breadth-first marker propagation will be more efficient than prior approaches when dealing with deep ISA hierarchies. Figure 4 shows the effect of increasing the depth of ISA hierarchies on breadth-first marker propagation compared with a version of RL using the AAK-direct approach. Again, the empirical results support the analytical results: breadth-first marker propagation is strikingly more efficient for deep ISA hierarchies.

To provide further support to our claim that breadth-first marker propagation is an efficient mechanism for learning with large data sets connected to background knowledge, we ran BFMP-RL on three real-world data sets with one million examples each, linked to large ISA hierarchies of background knowledge.

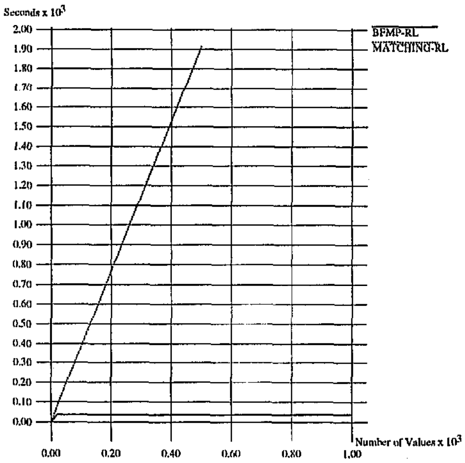

Figure 3: BFMP-RL vs. MATCHING-RL with Increasing Number of Values.

BFMP-RL's first learning task is to discover potential indicators of fraudulent cellular telephone calling behavior (Fawcett and Provost 1996). The training data comprise one million examples of cellular telephone calls. The data are linked to a hierarchy of domain knowledge about 1400 geographic locations of particular telephone numbers arranged in an ISA hierarchy of depth three. For these experiments we used 23 attributes with 18,000 total values.

We also analyzed a data set comprising U.S. Department of Health birth records linked with records of infant deaths. BFMP-RL was used to learn rules to predict infant mortality and survival. The database contains one million records with about twenty fields each, including demographic factors, birthweight, etc. The goal of the learning is to identify subgroups of the population with unusually high and unusually low infant mortality rates, in order to direct further research. The long-term goal of such work is to formulate policies that will reduce the nation's infant mortality rate (Provost and Aronis 1996).

Finally, we analyzed data describing incidents of potentially toxic exposures to plants. The database contains about one million records including symptoms, recommended actions, actual actions, outcome, as well as demographic and symptom information about the victims and information about the plant substances. We used 20 of these fields. These data were linked to background knowledge hierarchies describing geographic regions (1014 distinct areas), climate types (55 types), and botanical classifications (2400 individual species, genera, and families) (Krenzelok, Jacobsen, and Aronis 1995).

Figure 5 shows the effect on BFMP-RL of increasing the number of data items for these three real-world

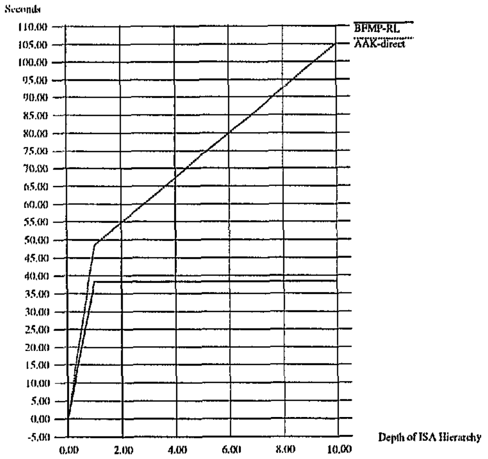

Figure 4: BFMP-RL versus AAK-direct with Increasing ISA Depth.

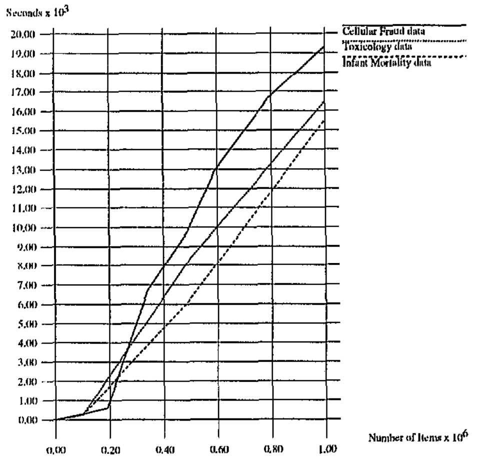

Figure 5: BFMP-RL with 10,000 to 1,000,000 Items.

data sets up to 1,000,000 items. BFMP-RL searched for rules of up to 5 conjuncts, using a beam width of 50. We note that, with these settings, MATCHING-RL took nearly two hours to learn with 100,000 examples and no ISA hierarchy. Furthermore, it is practically impossible to run MATCHING-RL on this workstation with many more than 100,000 items due to memory-management thrashing. On 100,000 cellular fraud examples BFMP-RL performed a relatively thorough search of the rule space defined by 23 features with 18,000 total values in an ISA hierarchy of depth 3 in under 5 minutes minutes on a desktop workstation.

Conclusions

We have shown that breadth-first marker propagation is an efficient alternative to existing approaches when learning problems contain hierarchically structured values, and that even without such structures the technique is an efficient replacement for matching. Furthermore, minimizing the number of passes through the example set avoids memory-management thrashing, and provides a method to mine archived datasets. Finally, breadth-first marker propagation links data mining to basic ideas from knowledge representation. In order to focus on efficiency gains, we have limited the discussion in this paper to ISA-hierarchical background knowledge. However the use of marker propagation provides a means to learn with more complex networks of background knowledge and multitable databases (Aronis, Provost, and Buchanan 1996).

Acknowledgements

This research was supported by National Science Foundation grant IRI-9412549.

References

Almuallim, H.; Akiba, Y.; and Kaneda, S. 1995. On handling tree-structured attributes in decision tree learning. In Proceedings of the Twelfth International Conference on Machine Learning, 12–20. Morgan-Kaufmann.

Aronis, J.; Provost, F.; and Buchanan, B. 1996. Exploiting background knowledge in automated discovery. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, 335–358. AAAI Press.

Clearwater, S.; and Provost, F. 1990. RL4: A tool for knowledge-based induction. In Proceedings of the Second International IEEE Conference on Tools for Artificial Intelligence, 24–30. IEEE C.S. Press.

Domingos, P. 1996. Linear-Time Rule Induction. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, 96–101. AAAI Press.

Fawcett, T.; and Provost, F. 1996. Combining data mining and machine learning for effective user profiling. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, 8–13. AAAI Press.

Krenzelok, E.; Jacobsen, T.; and Aronis, J. 1995. Jimsonweed (datura stramonium) poisoning and abuse ...an analysis of 1,458 cases. Presented at 1995 North American Congress of Clinical Toxicology, abstract in *Clinical Toxicology*, 33(5).

Núñez, M. 1991. The use of background knowledge in decision tree induction. *Machine Learning* 6:231–250.

Provost, F.; and Aronis, J. 1996. Scaling up inductive learning with massive parallelism. *Machine Learning*, 23:33–46.

Quinlan, J. 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufmann.