Small Disjuncts in Action: Learning to Diagnose Errors in the Local Loop of the Telephone Network

Andrea Pohoreckyj Danyluk
NYNEX Science & Technology, Inc.
500 Westchester Avenue
White Plains, NY 10604
danyluk@nynexst.com

Foster John Provost
Computer Science Department
University of Pittsburgh
Pittsburgh, PA 15260
foster@cs.pitt.edu

Abstract

Learning *general* rules is a basic goal of many concept learning systems. In a 1989 paper, Holte, Acker, and Porter pointed out that this bias toward generality had resulted in a problem with small disjuncts. The problem they discussed was that small disjuncts had high rates of misclassification, and that it was difficult to eliminate the error-prone small disjuncts without affecting the performance of other disjuncts. We describe a real domain based on NYNEX MAX, an expert system that diagnoses the local loop in a telephone network. We demonstrate with two inductive learning systems that a range of disjunct sizes is important for this domain despite the relatively high error rates of the small disjuncts. We conclude that the need for smaller disjuncts is a major reason that it is difficult to learn from errorful data in this domain.

1 INTRODUCTION

Learning *general* rules is a basic goal of many learning systems [Mitchell, 1980]. Inductive learning systems often make use of a bias that prefers large disjuncts to small disjuncts, where a large (small) disjunct is one that correctly classifies many (few) training examples.[1] Many believe in the notion that it is better to capture generalities than to have a knowledge base of specific cases from which you cannot extrapolate.[2] There are a number of reasons for this belief. Large disjuncts tend to have fewer conditions on their applicability. They are therefore simpler and more comprehensible, which also makes them more acceptable to experts who might be required to evaluate them. A related reason is Occam's razor–that is, the belief that given two results with similar accuracies on training data, the description with fewer conditions is more elegant (and will be more predictive on future data). Complex small disjuncts could be overfitting the training data, which would make them less applicable to future unseen cases. More practical considerations include the fact that having many small disjuncts may decrease the efficiency of a system, both in terms of space and the time it takes to consider a disjunct for each new input. In dealing with real application domains where one must anticipate errors in the data, it is undesirable to capture specific "odd" cases, as they may be the random or systematic errors in the domain.

Holte, et al., pointed out that the bias toward large disjuncts focuses attention away from small disjuncts, which deserve equal attention for many reasons [Holte, et al., 1989]. Many concepts include rare or exceptional cases and it is desirable for induced definitions to cover these cases. Second, small disjuncts collectively may match a large percentage of the examples that satisfy a definition. The problem with small disjuncts, as pointed out by Holte, et al., is that they are much more error prone than large disjuncts. This work was followed up by Quinlan [Quinlan, 1991].

According to Holte, et al., "The net effect of eliminating all small disjuncts is hard to predict..." They assume that small disjuncts are necessary for high accuracy and discuss methods for decreasing their error rates. We provide support for their basic assumptions by showing that in one novel, real-world domain, small disjuncts are necessary for high accuracy. More generally, we will show that a range of disjunct sizes is necessary for this domain.[3] We demonstrate the need for a range of sizes using two very different learning systems: c4.5 and RL.

In the next section we introduce the MAX expert system, which performs diagnosis of telephone troubles.

[1] We will refer to large disjuncts as "general" and small disjuncts as "specific".

[2] A notable exception is case-based reasoning, where specific instances are stored. But there the general knowledge is in the matching function. Learning this is not a trivial problem.

[3] That is, it is necessary given the description language.

We then introduce the learning task for the MAX domain. In subsequent sections we give results of test runs with c4.5 and RL. We will demonstrate that a range of disjuncts are important for this domain despite the relatively high error rates of the small disjuncts. We will conclude with a brief discussion of the impact of noise in such a domain and posit that the need for smaller disjuncts is a major reason that it is difficult to learn from errorful data in this domain.

2 THE NYNEX MAX DOMAIN

MAX is an expert system that was developed by NYNEX Science and Technology, Inc.[4] for the high level diagnosis of customer-reported telephone problems. Specifically, MAX concentrates on the *local loop*, the final segment of the telephone network which connects the customer to the central office. The learning task for this domain is to create the knowledge base inductively from trouble cases and their resolutions. Section 2.1 describes the expert system in more detail[5], and Section 2.2 discusses the learning task.

2.1 THE MAX EXPERT SYSTEM

When customers have problems with their phone service, they call 611 to report the trouble. A phone company representative takes a report of the problem (the *trouble*) and also initiates electrical tests on the line (a *mechanized loop test*, or MLT).[6] The representative sends the information from the trouble report and the MLT to a maintenance administrator, who evaluates the trouble and determines how the company should take care of (how to *dispatch*) the trouble. The maintenance administrator also gets information from a screening decision unit, a primitive rule-based system for diagnosing problems based on a two-character *vercode*—a summary of the MLT results. In general, the vercode alone does not provide sufficient information for an optimal decision.

MAX (Maintenance Administrator eXpert) plays the role of a Maintenance Administrator, i.e., it uses the MLT test results, together with other information, to make a screening diagnosis. The only exception is that MAX has the option of referring certain problems to a human maintenance administrator. MAX diagnoses a problem based on the following information: results of the MLT, including the vercode, knowledge about the customer's line, and general knowledge about equipment.

MAX is currently running for over 55 maintenance centers. Some of its benefits are: minimal change to the maintenance center's work flow; consistency; speed; reduction of erroneous dispatches over the screening decision unit (see [Rabinowitz, et al., 1991] for details). Some of MAX's limitations are: it is running for many diverse sites, with parameters used to customize the knowledge base for a given site (and the parameters are difficult to set); and the knowledge base must be updated periodically due to changing conditions.

2.2 THE MAX LEARNING TASK

Due to the volume of troubles handled by MAX, even a small improvement in its accuracy is extremely valuable—each dispatch typically involves at least one hour of time by a highly trained worker. It is interesting to consider whether machine learning techniques can be useful to help in automating modifications to the knowledge base. The process of tuning and updating the MAX system is very tedious and time intensive, and must be done to account for differences that exist between sites, as well as for changes that occur over time. One direction we have been investigating is the possibility of creating inductively a MAX knowledge base from current data about troubles and resolutions.[7]

The features used by MAX for diagnosis are essentially the features used to describe the examples for learning, with only minor differences. The attributes are largely resistances and voltages. In all, there are 14 attributes, 12 of them numeric. In most (approximately 90%) cases, one or more features are missing from the examples.

The classes to be learned are the five possible dispatch diagnoses for MAX: (i) dispatch to the cable; (ii) dispatch to the customer premise; (iii) dispatch to the central office; (iv) request further testing; (v) send to a human. For the experiments described in this paper, the criterion used to evaluate the learned concept description was predictive accuracy. The relative cost of errors varies, however, and current work is addressing that issue. The classifications used in the following experiments are the diagnoses given by MAX. Training on MAX's classifications provides us with (a) a consistent data set; and (b) a data set that is analogous to a set of cases provided by a maintenance center expert. RL and c4.5 were trained on data from one maintenance center at a time as there may be variations in MAX's diagnoses from site to site. All results reported are from a single site. Comparable results have been obtained for other locations.

[4] NYNEX is the parent company of New York Telephone and New England Telephone.

[5] Some of the text in this section is paraphrased from [Rabinowitz, et al., 1991].

[6] MLT was developed by AT&T.

[7] Other approaches are also being investigated, including modification of the existing knowledge base [Pazzani, 1993].

Figure 1: Number of Disjuncts for Varying Disjunct Sizes. Most disjuncts are small. Histogram plots number of disjuncts in concept description learned by c4.5 for varying disjunct sizes. The disjunct size is the number of examples covered, based on 500 training examples. Each bar is the average over ten runs with randomly selected training and test sets.

3 C4.5 RESULTS

Several experiments were performed using c4.5 [Quinlan, 1992]. In all cases, c4.5 was run with its default parameter settings. The gain ratio criterion was used to select tests. All results reported are before pruning, though the differences between the results before and after pruning are negligible.

For the first set of runs we used a set of 5845 examples. These were troubles that had been diagnosed by MAX in one maintenance center in one month. For each run of c4.5, a random subset of n examples was selected from this set as a training set. The maximum value of n in this set of runs was 5000. For each run, a randomly selected set of 845 examples was used as the test set. The training and test sets were non-overlapping. Accuracy on the test set was quite high when training with as few as 500 examples (89.05%, with s.d. 1.0), and after training on 5000 examples it reached 97% (s.d. .56). Results were averaged over ten sets of runs.

It is important to mention that we studied the examples that c4.5 was unable to classify correctly after training on 5000 examples. We found that in 60% of the cases c4.5's classification was actually better than MAX's. In this study, we presented to an expert the troubles that c4.5 had misclassified. We also gave the expert the MAX answer and the c4.5 answer, though we did not tell him which was which. The order of the two choices was also randomized. We asked the expert to select the diagnosis (dispatch) that he thought was better. In approximately 60% of the cases, he selected c4.5's dispatch.

We next performed a set of test runs to determine the occurrence of different-sized disjuncts in the decision trees learned for this domain. We trained c4.5 ten times on randomly selected sets of 500 examples. Figure 1 shows a histogram of our results, averaged over the 10 runs. The x-axis gives the size of a disjunct, i.e., the number of training examples covered by a leaf. The y-axis indicates the number of disjuncts, i.e., leaves, of that size. As shown in the figure, a wide range of disjuncts are learned by c4.5, and a significant number of those are small disjuncts.

We next performed a set of runs to evaluate the accuracy of different-sized disjuncts. We performed 10 runs in which we trained on 500 examples and tested on 2457. The training and test sets used in each run were randomly selected and disjoint. Figure 2a shows the number of test set examples classified by disjuncts of varying sizes, while Figure 2b shows the number of those that were classified erroneously.

The results of our experiments indicate that c4.5 is able to learn good decision trees for the MAX domain. Our analysis indicates that both large and small disjuncts appear in the trees learned, and that small disjuncts occur with relatively high frequency. Our analysis also confirms Holte, et al.'s conclusion that small disjuncts are more prone to errors than are large disjuncts. Experiments performed with RL indicate that small disjuncts cover such a large percentage of instances in this domain that they are necessary in spite of their error rates. Section 5 will discuss this in more

Figure 2: Number of Matches and Errors for Varying Disjunct Sizes. Smaller disjuncts are more error prone.

detail.

4 RL RESULTS

The RL learning system [Clearwater & Provost, 1990], a descendent of Meta-DENDRAL [Buchanan & Mitchell, 1978], searches for a disjunctive set of conjunctive rules. These rules are intended to form the knowledge base for an evidence-gathering performance system (a system which combines the evidence from several rules to make a classification). Below we describe RL briefly, show that RL can learn a set of rules that classifies troubles with high accuracy, and go on to study the contributions of large and small disjuncts.

RL structures its hypothesis space in a general-to-specific hierarchy rooted at the (syntactically) most general rule (every example is an element of the concept). Two types of specialization operators are used: (i) adding a conjunct, and (ii) specializing an existing conjunct. These operations are performed based on information provided in a partial domain model (PDM), which contains descriptions of the attributes, their types, and possible values and value hierarchies for each attribute. The PDM also contains other information used to determine when a rule is satisfactory/too general/too specific, and/or to restrict the hypothesis space.

The learning procedure keeps statistics on the various tentative rules and compares these with the criteria specified in the PDM. If a rule is too general, it is

Figure 3: Number of Disjuncts for Varying Disjunct Sizes

specialized. If too specific, it is discarded along with the entire subtree rooted there. If satisfactory, it is saved. Several search methods are provided; a beam search was used for the results presented here. The evaluation function for the beam search is specified in the PDM; we used the default function, which rates node r_1 better than node r_2 if the ratio of true positives to false positives covered by r_1 is greater than that of r_2 (in a tie, the rule with the larger coverage wins). In summary, the search procedure is a beam search of the space of syntactically defined rules, in which sections of the rule space are pruned if they are guaranteed not to yield results that will prove satisfactory with respect to the criteria of the PDM. For this domain we used a multiclass version of RL. Throughout this section, a simple evidence-gathering system will be used for testing that classifies examples based on the rule with the highest certainty factor that matches the example (as in [Quinlan, 1987]).

Via the partial domain model, users can specify different biases in different domains. We were concerned with the effect of different disjunct sizes, which are represented in RL's partial domain model by coverage threshold levels (a rule is too general if it covers more negative examples than the negative threshold allows; a rule is too specific if it covers fewer examples than the positive threshold allows). In this domain, we knew that the data were very clean and that there existed special cases (that would appear infrequently). We therefore set RL's thresholds such that in order for a rule to be considered a "good" rule, it would have to cover at least one positive example, while covering no negative examples (a *perfect rules only* bias). With this bias, both large and small disjuncts would be learned.

Across ten runs with random subsets of 500 training examples and 2457 test examples, the rule sets learned by RL averaged 86.5% accuracy (with a s.d. of 1.0). On a separate set of ten runs, where the rule sets were pruned (in a manner similar to that in [Quinlan, 1987]), the average accuracy was 87.8% (with a s.d. of 1.4). These results are comparable with those of c4.5. In similar runs, c4.5 averaged 88.9% accuracy (with a s.d. of 0.9).

Figure 3 shows that in the concept descriptions learned by RL, small disjuncts are abundant (Figure 3 is analogous to Figure 1). As observed with c4.5, RL's smaller disjuncts are more error prone than its larger disjuncts (see Figure 4, analogous to Figure 2). Disjuncts' error rates generally decrease with increased size. (As in Figures 1 and 2, data indicate averages over ten runs with randomly selected training sets of 500 examples and test sets of 2457.)

5 GENERALITY VS. ACCURACY

In order to test the hypothesis that learning small disjuncts was necessary for learning high-accuracy concept descriptions, we conducted a series of RL runs with different minimum levels of generality. The minimum level of generality was enforced by the threshold on the positive coverage of learned rules. A positive threshold of p restricts RL from learning rules that cover less than $p\%$ of the positive examples of each class. In all cases, the rules were not allowed to cover any negative examples. Figure 5 depicts the relationship between the level of generality and the corresponding test-set accuracy. Each point in the figure is an average over ten runs with randomly selected train-

Figure 4: Number of Matches and Errors for Varying Disjunct Sizes

ing sets of 500 examples, and test sets of 2457. Error bars indicate 95% confidence intervals using Student's t.

Figure 5 shows a (decreasing) linear relationship between the minimum size of the learned disjuncts and the accuracy of the corresponding concept description (a line can be fit to the data with R=0.99). This indicates that (with respect to the rule-based description language) there are special cases in the MAX data that appear very infrequently, but are important parts of the concept description. This is consistent with our prior domain knowledge, and the assumptions of previous investigators. In fact, the linear relationship indicates that disjuncts of all sizes are integral parts of the learned concept descriptions.

6 DISCUSSION: DISJUNCT SIZES AND NOISE

The results of this study suggest that the incorporation of good mechanisms for learning small disjuncts is one reason for the widespread success of c4.5 (though the version we used did not include improvements outlined by Quinlan in [Quinlan, 1991]). Our initial success with c4.5, as well as our ability to choose different inductive biases in RL, led us to examine the features of our learning systems that allowed us to learn the knowledge base. It confirmed the need for a range of disjunct sizes, especially the important role played by small disjuncts.

Though c4.5 and RL were able to learn good knowl-

Figure 5: Minimum Disjunct Size Versus Accuracy. Increasing the generality required of rules learned by RL decreases the corresponding test-set accuracy. Graph plots the accuracy of concept descriptions leaned by RL when a minimum level of generality was enforced. The minimum level of generality is a requirement that a learned rule must cover at least a certain percentage of the examples of the corresponding concept. Each point is an average over ten runs, error bars are 95% confidence intervals based on Student's t.

edge bases for MAX, neither system was able to learn a good knowledge base when trained on noisy data. The difficulties stem from two major sources. First, it is difficult to distinguish between noise and true exceptions. In the MAX domain approximately half of the example coverage comes from small disjuncts (exception-type rules covering fewer than 10 training examples), all of which can be significantly affected by the presence of noise. Second, in the MAX domain, errors in measurement and classification often occur systematically rather than randomly. Thus it is difficult to distinguish between erroneous consistencies and correct ones.

We performed a set of runs in which we trained on troubles analyzed by MAX, but rather than using the MAX diagnosis as the classification for a trouble, we used the diagnosis as determined by the technician who actually solved the trouble in the field. We then tested on other troubles as diagnosed by the field techs. There are a number of sources of errors in this data, and many of the errors differ from the type of noise that one generally expects. They include electronic faults in data collection and reporting devices and noise in transmission lines. Additional sources of error include:

- Errors in repair: The trouble may not have been repaired correctly. Or a problem may have been solved, but there may in fact have been multiple troubles that should have been attended to.
- Errors in translation of codes: The people who solve the troubles indicate their diagnosis using a different set of codes than those used to dispatch the troubles. There is not an exact one-to-one correspondence between the two sets of classifications. Thus translating one coding scheme to another will introduce errors, some due to pure mistranslation and others due to inherent ambiguities.
- Errors in coding: The codes used to indicate the trouble found are complicated and errors in coding will occur naturally. A repair person might also deliberately miscode a diagnosis.

Many of these types of errors may occur systematically. This is especially true in the case of deliberate miscodings. We have no exact estimates of the number of errors occurring in the MAX data, where the trouble diagnoses are those actually given by the technicians. When trained and tested on this data, however, neither c4.5 nor RL was able to achieve accuracy above 60% even after training on 1000 examples.

Noise is a fundamental problem that has been addressed in many systems. The MAX domain has pointed out new issues to consider, however. In a domain where small disjuncts play a critical role in the overall performance of a system, it is imperative to find a mechanism to distinguish true exceptions from noise. Furthermore, in a domain such as ours, it is important to be able to distinguish between general rules that are correct, and erroneous trends. In order to achieve greater accuracy in MAX we will almost certainly require a good model of probable errors (more robust than the one we currently have) to perform intelligent instance selection.

7 CONCLUSIONS

Learning general rules is a fundamental goal of a number of learning systems. For a variety of reasons, ranging from comprehensibility to efficiency, many systems are biased to concentrate on general rules rather than on special cases. Holte, et al., first identified the fact that this bias drew attention away from another important problem, that of ensuring the accuracy of small disjuncts. We have described one domain, the NYNEX MAX domain, in which learning large disjuncts alone is not sufficient. However, learning a range of disjuncts is. Two learning systems, c4.5 and RL, were able to learn very accurate rules. c4.5 incorporated criteria for including both large and small disjuncts into its bias. The RL work showed that while learning a combination of large and small disjuncts was sufficient for learning concept descriptions with high accuracy, learning only large disjuncts was not. We have shown this to be true even though the error rate of disjuncts increases as the size decreases.

We agree with previous authors that the quality of both large and small disjuncts is important to the performance of a learning system. We show MAX to be a domain in which both are essential. We believe that there are other domains in the real world that require the ability to learn both generalities and special cases. This must be a capability of learning systems, if they are to succeed in these areas. Balancing the tradeoffs between true special cases and anomalies due to errors is still a great—and crucial—challenge. This is especially true since errors may be systematic as well as random.

References

Buchanan, B. & Mitchell, T. (1978). Model-directed Learning of Production Rules. In D. A. Waterman & F. Hayes-Roth (ed.), *Pattern Directed Inference Systems*, p. 297-312. New York: Academic Press.

Clearwater, S. & Provost, F. (1990). RL4: A Tool for Knowledge-Based Induction. In Proceedings of the Second International IEEE Conference on Tools for Artificial Intelligence, p. 24-30. IEEE C.S. Press.

Holte, R. C., Acker, L. E. & Porter, B. W. (1989). Concept Learning and the Problem of Small Disjuncts. In Proceedings of the Eleventh International Joint Conference on Artificial Intelligence, p. 813-818. San Mateo, CA: Morgan Kaufmann.

Mitchell, T. M. (1980). The Need for Biases in Learning Generalizations. Technical Report CBM-TR-117, Rutgers University.

Pazzani, M. J. (1993). Finding accurate frontiers: A new approach to analytic learning. In Proceedings of the National Conference of Artificial Intelligence. San Mateo, CA: Morgan Kaufmann.

Quinlan, J. (1987). Generating Production Rules from Decision Trees. In Proceedings of the Tenth International Joint Conference on Artificial Intelligence, p. 304-307. San Mateo, CA: Morgan Kaufmann.

Quinlan, J. R. (1991). Improved Estimates for the Accuracy of Small Disjuncts. *Machine Learning*, 6(1), p. 93-98.

Quinlan, J. R. (1992). *C4.5: Programs for Machine Learning*. San Mateo, CA: Morgan Kaufmann.

Rabinowitz, H., Flamholz, J., Wolin, E. & Euchner, J. (1991) NYNEX MAX: A Telephone Trouble Screening Expert. In R. Smith & C. Scott (ed.), *Innovative Applications of Artificial Intelligence 3*, p. 213-230. Menlo Park, CA: AAAI Press.