Naval Research Laboratory

Washington, DC 20375-5320

NRL/FR/5510--94-9707

(2)

Case-Based Sonogram Classification

DAVID AHA
PATRICK HARRISON

Naval Center for Applied Research in Artificial Intelligence
Information Technology Division

January 31, 1994

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (*Leave Blank*)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	January 31, 1994	

4. TITLE AND SUBTITLE

Case-Based Sonogram Classification

5. FUNDING NUMBERS

PE - 62234N
TA - RS34-C74-000
WU - DN2573

6. AUTHOR(S)

David Aha and Patrick Harrison

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Research Laboratory
Washington, DC 20375-5320

8. PERFORMING ORGANIZATION REPORT NUMBER

NRL/FR/5510--94-9707

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Office of Naval Research
Arlington, VA 22217

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (*Maximum 200 words*)

This report replicates and extends results reported by Naval Air Warfare Center (NAWC) personnel on the automatic classification of sonar images. They used novel case-based reasoning systems in their empirical studies, but did not obtain comparative analyses using standard classification algorithms. Therefore, the quality of NAWC results were unknown. We replicated the NAWC studies and also tested several other classifiers (i.e., both case-based and otherwise) from the machine learning literature. These comparisons and their ramifications are detailed in this report. Next, we investigated Fala and Walker's two suggestions for future work (i.e., on combining their similarity functions and on an alternative case representation). Finally, we describe several ways to incorporate additional domain-specific knowledge when applying case-based classifiers to similar tasks.

14. SUBJECT TERMS

Case-based reasoning
Classification
Sonograms
Empirical comparisons
Learning algorithms

15. NUMBER OF PAGES

17

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std 239-18
298-102

CONTENTS

CASE-BASED SONOGRAM CLASSIFICATION

1. INTRODUCTION AND MOTIVATION

Fala and Walker (1993) describe results of applying three novel case-based reasoning (CBR) algorithms to a submarine classification task that used data obtained from sonogram line readings. Although some of their algorithms appeared to perform well, they did not describe comparisons with alternative algorithms. It is difficult to assess the performance of their algorithms without these comparisons.

We replicated their studies, included comparisons with several other algorithms from the machine learning literature, and included two studies suggested as future work by Fala and Walker. We discovered strengths and weaknesses of their algorithms. We also found ways to improve their performance. This report details our studies and summarizes ways for incorporating additional domain-specific knowledge into case-based classifiers.

Section 2 details Fala and Walker's (1993) experiments. Our results with the same dataset are described in Section 3. Section 4 discusses the ramifications of these results and Section 5 provides suggestions on how to incorporate more domain-specific knowledge.

2. CONTEXT: NAWC'S SONAR ANALYSIS SYSTEM

Fala and Walker (1993) analyzed their CBR tools' ability to automatically classify acoustic sonar images of submarines. Their interviews with experts who do this task (i.e., three aviation anti-submarine warfare technicians) suggested that experts might use CBR-like classification strategies. This motivated Fala and Walker to create a CBR system that automates this process.

They began by collecting low-level features describing lines in sonogram readings. These were used to design a representation for cases. More specifically, 21 cases were compiled using an automated system for extracting lines from sonograms. Each case was then classified by an expert into one of five possible classifications (i.e., submarines). The raw cases contained between 16 and 49 lines, all taken at the same noise levels. The frequency resolution was 100 and the frequencies of the lines ranged between five and 400 Hz.

The raw data were not represented directly in the cases. Instead, Fala and Walker incorporated the notion that humans can visually separate lines as close as 1/32 of an inch. Their representation was based on counting, individually for each sonogram, the number of lines per *frequency boundary*. The frequency boundary for a given line was obtained using

$$\text{frequency(line)} = \text{truncate}\left(\frac{\text{line} - 5}{3 + \frac{1}{32}} - 1\right) \quad (1)$$

Manuscript approved Nov. 6, 1993.

When applied to the given 21 sonograms, this formula yields frequency boundaries in the range [0, 127]. Since this function can map many frequencies to a single frequency boundary, the 128 feature values constituting each case were non-negative integers.

Fala and Walker used the standard leave-one-out strategy (Weiss and Kulikowski 1991) to evaluate the classification accuracy of three novel variants of the nearest-neighbor algorithm (Fix and Hodges 1951; Cover and Hart 1967; Duda and Hart 1973). This strategy simply includes all but one case in the training set and uses the remaining case as the only one used during testing. This is repeated once for each case in the data set. Since one of the cases had a unique classification relative to the remaining cases, it was not included in their experiments. Thus, only 20 of the 21 cases were used in their experiments, which left only four classes represented by cases in the dataset.

The nearest-neighbor algorithm has been extensively analyzed in the literature on pattern recognition (Dasarathy 1991) and machine learning (Aha, Kibler, and Albert 1991), where it is viewed as an *instance-based learning* (IBL) algorithm. For the purposes of this report, IBL algorithms can be thought of as consisting of the following three functions:

1. *Normalization*: Preprocesses the data, and is primarily used to equalize the relative influences of features in similarity computations.
2. *Similarity*: Used to compute the similarity between two cases.
3. *Prediction*: Given the results of the similarity computations, this function details how a classification prediction is made.

There is no standard normalization function used with the nearest-neighbor algorithm. However, it is defined as using the following Euclidean (dis)similarity function (assuming F features are used to describe each case x and y):

$$\text{(dis)Similarity}(x, y) = \sqrt{\sum_{i=1}^{F} (x_i - y_i)^2}. \quad (2)$$

The nearest-neighbor prediction function simply predicts that the given case's class is the same as that of its most similar case (i.e., the least distant case). Several studies on machine learning, case-based reasoning, statistics, pattern recognition, cognitive psychology, and other topics have used this algorithm in empirical comparison studies as a straw-man due to its simplicity and popularity (Sebestyen 1962; Reed 1972; Duda and Hart 1973; Shepard 1983; Breiman, Friedman, Olshen, and Stone 1984; Kibler and Aha 1987; Bareiss 1989; Dasarathy 1991; Weiss and Kulikowski 1991; Shavlik, Mooney, and Towell 1991; Aha, Kibler, and Albert 1991). It is well-known that while the nearest-neighbor algorithm is a relatively robust classifier, its primary drawbacks include an inability to tolerate irrelevant attributes, large storage requirements, and relatively high computational complexities for classifying new cases.

Fala and Walker's (1993) variants of the nearest neighbor function used no normalization function, used the nearest-neighbor prediction function, and did not involve repairs to these drawbacks. Instead, they used novel similarity functions, which they called *comparison operators*. The definitions of these three functions, which are listed in Table 1, were derived from their interviews with expert sonogram classifiers. More specifically, MATCHES was suggested by experts noting that a given frequency boundary of the two sonograms both contain or do not contain lines. The HITS function corresponds to the number of boundaries in the two sonograms that both contain lines; it sums the number of such lines in each such frequency boundary. Finally, the MISSES function

Table 1 – Similarity Functions Used by Fala and Walker (1993), Where x is the New Case and y is the Stored Case

NAME	DEFINITION
MATCHES	$\sum_{i=1}^{F}$ if $(x_i = y_i)$ or $((x_i \geq 1)$ and $(y_i \geq 1))$ then 1 else 0
HITS	$\sum_{i=1}^{F}$ if $((x_i \geq 1)$ and $(y_i \geq 1))$ then $\min(x_i, y_i)$ else 0
MISSES	$\sum_{i=1}^{F}$ if $((x_i \geq 1)$ and $(y_i = 0))$ then x_i else 0

corresponds to the number of boundaries in a stored case's sonogram that are missing lines visible in the new case.

Innumerable methods exist for making predictions when there is a tie among the most similar stored cases. Walker (1993) noted that the method used in their study was perfectly optimistic. If any of the most similar neighbor's classifications matched that of the test case, then the classification was deemed to be correct. We used this same optimistic tie-breaking method in our own experiments with nearest-neighbor variants.

Fala and Walker's algorithms compare two sonograms based on their number of lines per frequency boundary. However, each sonogram involved a different number of line readings (i.e., between 16 and 49). Therefore, they *scaled* their cases by using the *percentage* of lines in a given frequency boundary rather than their raw number. These case representations were obtained from the raw data by dividing each case's feature value by the number of total lines in that case.

Fala and Walker (1993) reported their leave-one-out results using this 128-feature representation for these three variants of the nearest-neighbor algorithm. The respective classification accuracies for MATCHES, HITS, and MISSES were were 14/20 (70%), 17/20 (85%), and 6/20 (30%). Guessing randomly among the four classes yields an accuracy of 25%, whereas always guessing the most frequent class yields an accuracy of 40%. Simply using nearest-neighbor prediction when representing the cases with their average line reading yields 55%.

3. FOLLOWUP STUDIES

While the accuracies recorded by MATCHES and HITS are greater than that attainable by always guessing the most frequent class in the dataset, it is not obvious from this study alone whether they are "very good." Our first goal was to investigate this claim. We also tested several alternative preprocessing strategies that have been shown to dramatically alter classification performance on some problems (Aha 1990; Turney 1993). Finally, we investigated Fala and Walker's two suggestions for future work. Their first suggestion involves combining the effects of their algorithms. Their second suggestion involves examining the algorithms' behavior when using a more continuous case representation. These studies are detailed in the following subsections.

3.1 A Comparison Study

We selected our suite of algorithms from among several commonly known algorithms in the pattern recognition and machine learning literatures. In doing so, we also replicated Fala and Walker's experiments with their three case-based learning algorithms.

The first comparison algorithm we included is ¬MISSES, which is identical to MISSES except that it negates the computed sums before making classification predictions. We did this because

Table 2 - Leave-One-Out Results on the Sonar Database Using Scaled Data

Classifier	Fala and Walker's (1993) Results	Our Results
random guess	25%	25%
guess most frequent	40%	40%
MATCHES	70%	70%
HITS	85%	85%
MISSES	30%	30%
¬MISSES		75%
HAMMING		85%
EUCLIDEAN		70%
CUBIC		75%
C4.5		60%
CN2		40%
BACKPROP		85%

MISSES computes *dissimilarities* rather than similarities. This can be observed by noting that its values increase as the number of frequency boundaries differ. Fala and Walker's (1993) empirical results support the fact that this poor similarity function is outperformed by always guessing the most frequently occurring class in their dataset.

Given our familiarity with the nearest-neighbor classifier and its similarity with Fala and Walker's algorithms, it is natural to ask what accuracies it can attain. Therefore, we tested three additional variants of this algorithm. These variants differ only in their similarity function. The Euclidean similarity function shown in Eq. (2) is actually the Minkowski metric with $r = 2$:

$$(\text{dis})\text{Similarity}(x, y) = \sqrt[r]{\sum_{i=1}^{F}(x_i - y_i)^r}. \tag{3}$$

We used the Minkowskian dissimilarity function with $r = 1$ (HAMMING), with $r = 2$ (EUCLIDEAN), and with $r = 3$, which we'll refer to as CUBIC.

We also tested three common machine learning algorithms: a decision tree inducer named C4.5[1] (Quinlan 1993), a decision rule inducer named CN2[2] (Clark and Niblett 1989; Clark and Boswell 1991), and the Backpropagation algorithm[3] (Rumelhart, McClelland, and the PDP Research Group 1986).

Table 2 summarizes the results for the algorithms alongside the results from the original study and the baseline results for guessing randomly or always guessing the most frequent class in the dataset. As with the original study, we scaled the data before applying the algorithms. Four observations are noteworthy:

[1] C4.5 gave the same results when run both with and without its post-pruning option in effect.

[2] CN2 was tested on 24 combinations of its parameter settings. This included all four of its error estimating strategies, three values for its star size (i.e., 3, 5, and 7), and both with and without its maximum class prediction option. Its chi-square threshold value was always set to 0.

[3] BACKPROP was tested once for each of 384 combinations of its input parameters, including two methods for normalizing the input data (i.e., simple linear interval and z-score), four momentum values (i.e., 0.1, 0.4, 0.7, and 0.9), four learning rates (0.01, 0.1, 0.3, and 0.5), four temperatures (0.1, 0.5, 1.0, and 2.0), and three numbers of hidden units (i.e., 5, 10, and 25). Classification was based on the output node with the highest activation.

Table 3 – Leave-One-Out Results on the Sonar Database Using Scaled and Unscaled Data

Classifier	With Scaled Data	With Raw Data
random guess	25%	25%
guess most frequent	40%	40%
MATCHES	70%	70%
HITS	85%	80%
MISSES	30%	30%
¬MISSES	75%	70%
HAMMING	85%	70%
EUCLIDEAN	70%	75%
CUBIC	75%	65%
C4.5	60%	65%
CN2	40%	50%
BACKPROP	85%	90%

1. We replicated the original results.
2. As expected, the ¬MISSES similarity function easily outperformed MISSES and performed comparatively well with the other functions tested.
3. The Minkowski metric's results were somewhat sensitive to the value of r. The HAMMING distance function performed as well as HITS.
4. The machine learning algorithms fared poorly because they perform comparatively well only with larger-sized databases. Previous research also suggests that C4.5 and CN2 may not work well when the data is completely numeric (Aha 1992). As usual, BACKPROP performed well primarily because we tested it on a large number of values for its many parameters.

In general, it appears that the performance of the MATCHES and HITS algorithms selected by Fala and Walker performed well relative to this suite of algorithms.

3.2 Alternative Preprocessing Strategies

It is possible that, by using different normalization functions in the case-based classifiers, higher classification accuracies can be obtained. However, it is not obvious which case representation, scaled or raw, supports better classification performance.

Although Fala and Walker reasoned that their data should be scaled, it is not obvious what gains were obtained by doing so. Therefore, we repeated the experiments using the unscaled case representation. The results are displayed in Table 3. Four of the algorithms recorded lower accuracies using this representation while three recorded higher accuracies. While HITS's accuracy decreased slightly, the accuracies of the other new algorithms did not change. Therefore, it is not obvious which representation supports better classification performance. Given this, we retested the case-based classifiers using both case representations in our next study.

In the previous experiments, no normalization function was used in any of the algorithms. We were curious as to whether improved classification accuracies could be obtained by using normalization functions. Therefore, we compared our previous results with those obtained using two standard

Table 4 – Leave-One-Out Results on the Sonar Database Using Scaled and Unscaled Data

Classifier	With Scaled Data			With Raw Data		
MATCHES	70%	70%	70%	70%	70%	70%
HITS	**85%**	75%	**85%**	80%	75%	70%
MISSES	30%	20%	20%	30%	20%	25%
¬MISSES	75%	65%	70%	70%	65%	70%
HAMMING	**85%**	70%	55%	70%	60%	60%
EUCLIDEAN	70%	75%	80%	75%	70%	80%
CUBIC	75%	65%	**85%**	65%	75%	80%

normalization strategies. The first, named *linear interval*, normalizes value v of feature f based on its minimum and maximum across all cases. The normalized value is computed using

$$\text{Normalize}(f, v) = \frac{v - \text{minimum}(f)}{\text{maximum}(f) - \text{minimum}(f)}. \quad (4)$$

The second normalization function we tested is *z-score*. This function subtracts the feature's mean value from the feature value and divides by the feature's standard deviation. The results for all three normalization procedures and both case representations are shown in Table 4.[4] In summary, the scaled representation supports the highest accuracies, and using the linear interval normalization function yields lower accuracies than the other two strategies.

3.3 Combining the Original Algorithms

Fala and Walker (1993) suggested combining the effects of their algorithms. There are four obvious combinations to consider corresponding to combining pairs of the three algorithms or all of them at once. In each case, the combined algorithm simply invokes more than one of the similarity functions in Table 1. For example, when using the three-algorithm combination, similarities among pairs of cases are still computed by summing up pairwise similarities among the features. Thus, the MATCHES component adds one to the similarity when the two values are equal or both positive, the HITS component then adds the smaller value if both are positive, and the MISSES component *subtracts* the test case's value if it is positive and the stored case's value is zero. The other, pairwise combinations of the three similarity functions are computed similarly, but with only two similarity components rather than three.

We evaluated all four combinations by using both case representations. The results are shown in Table 5. In this case, two of the algorithms show improvement under some of the normalization functions. Perfect classification accuracy results for this dataset when combining the MATCHES and ¬MISSES similarity functions and using no normalization function on the raw data. High accuracies result under three conditions when combining all three similarity functions. In summary, combinations including both MATCHES and ¬MISSES yield better performance on this dataset.

[4] The second through fourth columns display the results using scaled case representations. The third column's results correspond to using a linear interval normalization procedure, while the fourth column's results are from computing z-score normalizations. The remaining columns include the same results when using unscaled case representations.

Table 5 - Leave-One-Out Results on the Sonar Database Using Scaled and Unscaled Data

Classifier	With Scaled Data			With Raw Data		
MATCHES–MISSES	65%	75%	75%	**100%**	75%	80%
HITS–MISSES	85%	70%	75%	80%	65%	70%
MATCHES+HITS	60%	70%	80%	80%	65%	80%
MATCHES+HITS–MISSES	60%	**95%**	75%	**90%**	**90%**	75%

Table 6 - Leave-One-Out Results on the Sonar Database Using the 126-Feature "Boundary-Overlapping" Representation

Classifier	With Scaled Data			With Raw Data		
MATCHES	85%	85%	85%	85%	85%	85%
HITS	80%	75%	80%	80%	80%	80%
MISSES	20%	20%	20%	20%	30%	25%
¬MISSES	65%	70%	70%	65%	70%	70%
MATCHES–MISSES	85%	**90%**	65%	80%	85%	70%
HITS–MISSES	80%	75%	75%	75%	75%	75%
MATCHES+HITS	85%	**90%**	80%	80%	80%	75%
MATCHES+HITS–MISSES	85%	80%	70%	75%	70%	75%
HAMMING	85%	85%	85%	70%	70%	80%
EUCLIDEAN	75%	75%	80%	75%	70%	75%
CUBIC	70%	65%	85%	85%	75%	75%
C4.5	80%			50%		
CN2	55%			40%		
BACKPROP	**90%**			**90%**		

3.4 An Overlapping Case Representation

Fala and Walker (1993) also suggested using an alternative case representation in which the frequency bounds overlap. This *coarse coding* representation (Rumelhart, McClelland, and The PDP Research Group 1986) modifies the original representation via an averaging process. Each feature in this representation corresponds to a sequence of frequency boundaries rather than to a single boundary. For example, we used an overlap of three so that the first feature's value is the sum of the values of the first three frequency boundaries. Similarly, feature i contains the sum of the values from frequency boundaries i, $i + 1$, and $i + 2$, where i ranges between zero and 125. Table 6 summarizes the results when using this 126-feature representation.

In general, this boundary-overlapping representation did not yield higher performances. None of the accuracies were over 90%. However, it is possible that alternative boundary-overlapping representations can support higher classification accuracies.

4. DISCUSSION

These results should not be interpreted as an indication that, for example, the combined algorithms will outperform the others in general. We plainly see that the algorithm's accuracies vary depending on the normalization function and case representation. Each algorithm has its own classification bias, and that bias can only be best for a finite set of databases (Utgoff 1986; Schaffer 1993).

However, it is interesting to note that a cognitively motivated similarity function, such as the combination of MATCHES and MISSES, was the only similarity function that attained perfect classification accuracy on this dataset. Tversky (1977) has argued for the psychological plausibility of such similarity definitions. His *contrast model* of similarity is similar to the combination of MATCHES and MISSES in that it is an increasing function of the cases' commonalities, a decreasing function of their differences, and computes these separately. However, one difference is that the contrast model subtracts differences in *both* directions rather than only *one* direction (i.e., test value minus the stored value). Thus a variant of Tversky's model would also subtract from the cumulative cases' similarity the value of the stored case's feature whenever it was positive and the test case's value for that feature was zero. We extended MISSES to include this property and replicated the experiments. The results were similar to those previously reported.

One interesting avenue for research concerns evaluating the generality of the hybrid MATCHES + MISSES classifier. While it performed well for this application, we would like to determine whether it has specific general benefits in comparison to more established algorithms.

While the performance of some individual algorithms improved when using the alternative normalization functions, in general normalization did not improve classification performance. This information is still useful in that it tells us that the good performance of Fala and Walker's classifiers is not primarily due to anomalies of the initial representation. After all, MATCHES performed equally well under all of the normalization methods that were tested while the performance of the other two algorithms decreased slightly when using the linear interval and z-score functions.

Naturally, there are several questions not addressed by this research, such as the relationship between the similarity function and the case representation and their contribution towards performance. Studying this requires varying one component of the case-based classifier while controlling the selection of the other components. For example, we would like to understand the comparative limitations of case-based classifiers and other approaches. This is partially addressed elsewhere (Aha 1992), but these experiments are beyond the scope of this report.

5. INCORPORATING DOMAIN KNOWLEDGE

Although it is comforting that perfect classification accuracy could be achieved on this sonogram dataset via a combination of the MATCHES and MISSES algorithms, we do not know whether this result will scale up. That is, the current dataset is quite small - only 20 cases - and this hybrid algorithm may not perform well on larger datasets.

The algorithms we have described so far are *knowledge-poor* in that they use only a minimum of domain-specific knowledge. In practical applications, there is no substitute for such knowledge, and we believe that to attain equally good classification accuracies with larger sonogram datasets, more *knowledge-intensive* case-based classifiers will be required.

There are at least five ways to incorporate domain-specific knowledge into a case-based classifier:

1. Use a more appropriate case representation,
2. Improve the normalization function,
3. Improve the similarity function,
4. Improve the prediction function, and
5. Add a postprocessing function.

Several of these methods have overlapping effects. For example, some similarity functions effectively modify the case representation.

5.1 Case Representations

The simplest and most effective way to improve classification performance often involves using a better case representation for the given task. This relies on having experts available to suggest alternative representations. For example, Fala and Walker (1993) suggested that experts classify sonogram readings using only a subset of the sonogram rather than its entirety. Thus, perhaps these cases could more profitably be represented using only a subset of their features.

Alternatively, if there is sufficient data and expertise available to statistically analyze the data, then a data modelling approach could be used to repeatedly propose and test alternative case representations.

A final alternative is to have the algorithm itself propose alternative case representations. In the machine learning literature, several algorithms implementing *feature construction* and *constructive induction* have been used to modify the given case representation (Birnbaum and Collins 1991). While few such algorithms have been described for use with case-based classifiers (e.g., Aha 1991), constructive induction algorithms have greatly improved classification behavior on a limited set of applications. Future research should include an investigation to determine whether they are useful for similar sonogram classification tasks.

5.2 Normalization Functions

Although we examined three simple normalization strategies in this report (i.e., none, linear interval, and z-score), they are certainly knowledge-poor. Recently, Turney (1993) proposed using several *contextual normalization* functions to exploit the context of the application. One of Turney's approaches normalizes data by estimating each feature's expected value and variance using some standard prediction function on "healthy baseline" data and then normalizes data using a function of these estimates. His algorithm improved the accuracy of a simple case-based classifier by 13% on a gas turbine classification task. Future work should include studying whether similar functions could be used for sonogram classification tasks.

5.3 Similarity Functions

As mentioned previously, a primary weakness of the nearest-neighbor function is that it is sensitive to the presence of irrelevant features in the case representation. This is because its similarity function, the Euclidean distance metric, assumes that all features are equally relevant. That is, each feature has equal impact on similarity computations.

Dynamic feature selection algorithms alleviate this problem. Most of them assign weights to each feature. The most relevant features are assigned the highest weights. For example, a typical weighted-Euclidean similarity function is

$$(\text{dis})\text{Similarity}(x, y) = \sqrt{\sum_{i=1}^{F} w_i \times (x_i - y_i)^2}, \quad (5)$$

where W_i is the weight of feature i. Using this function, features with weights of zero are effectively ignored during similarity computations, whereas features whose weights are high have the most impact on determining similarity. Several weight-learning methods have been proposed, including algorithms based on incremental training (Salzberg 1990; Aha 1989), genetic algorithms (Kelly and Davis 1991), decision trees (Kibler and Aha 1987; Cardie 1993), information theory (Bakiri 1991), ones for symbolic-valued attributes (Stanfill and Waltz 1986), and several others.

All of these algorithms can be run in a knowledge-poor fashion. However, knowledge-intensive algorithms are often more appropriate, especially when only a small amount of data is available for an application with a large instance space. For example, Cain, Pazzani and Silverstein (1991) demonstrated that a simple set of explanation-based learning trees can be used to determine, for each case, which attributes are relevant for their application. By using this additional knowledge, they increased their classification accuracy by 18% in their database on foreign trade negotiations. Future research should include analyzing how well similar algorithms improve performance on sonogram classification tasks.

Jabbour (et al. 1987) and his colleagues have published studies on yet another method for incorporating knowledge into a similarity function. Their power load forecasting system, **ALFA**, uses an eight-nearest-neighbor function to predict power load for the Niagra Mohawk Power Company (NIMO) of central New York State. Cases consist of meteorological data from three cities in New York. To prevent similarities from being computed on possibly misleading cases, thresholds are placed on the tolerated amount of difference allowed on the day of week, hour of day, and month of year features. If these thresholds are not met, then the similarity between two cases is deemed to be zero (i.e., effectively, similarities are not computed for large portions of their huge database). Similar domain-specific thresholds may prove useful for sonogram classification tasks.

5.4 Prediction Functions

The only prediction function that we have discussed has been the single nearest-neighbor function. Alternative functions should be considered in future research tasks. The most obvious alternative is k-nearest-neighbor where $k > 1$. Many studies have suggested that its bias is beneficial, and it is well-known that linear increases in k yield exponential decreases in the difference between the learning rates of k-nearest-neighbor and the Bayes optimal learner (Cover and Hart 1967; Cover 1968; Duda and Hart 1973).

Additionally, when $k > 1$, alternative similarity functions should be considered, especially those in which similarity decreases exponentially with distance (Nosofsky 1986; Hintzman 1988; Aha and Goldstone 1992). These studies on human concept formation were all motivated by Shepard's (1987) findings that subjects tend to generalize two stimuli based on an exponentially decreasing function of the stimuli's distance in a psychological space. This observation may soon prove useful for improving the performance of automated classification algorithms.

5.5 Postprocessing Functions

Finally, postprocessing functions have also been shown to improve the performance of case-based classifiers. For example, after ALFA generated a prediction, it consulted a set of rules to adjust for annual population drift and account for days on which power load requirements would differ greatly from their norm (e.g., Super Bowl Sunday). This allowed ALFA to attain predictive accuracies similar to those attained by NIMO's experts.

Another way to incorporate knowledge during postprocessing was demonstrated in CABERESS by Clark, Feng, and Matwin (1993). They simply averaged the predictions from a case-based classifier with those derived from a domain-specific model of their classification task. Similar approaches should be useful for sonogram classification studies.

6. CONCLUSION

This report describes followup studies to Fala and Walker's (1993) study of three CBR algorithm's ability to classify sonar data. We replicated their experiments, extended them, compared their results with those from several other algorithms, and investigated other representations and normalization functions. We also tested Fala and Walker's suggestion to combine their similarity functions and found that, under some conditions, perfect or near-perfect classification performance could be obtained when using their algorithms. Our future interests include investigating whether existing knowledge-intensive learning strategies for case-based reasoners can improve performance on more challenging sonogram classification tasks. Therefore, we outlined many possible ways to explore these issues.

7. ACKNOWLEDGMENTS

We thank Midshipman/Lieutenant Ahmed Williamson for discussions concerning the technical aspects of sonogram classification and specific details regarding our followup study. We also thank John Walker for many invaluable discussions on his experiments and Saul Oresky for his editorial assistance.

REFERENCES

Aha, D. W. (1989). Incremental, Instance-Based Learning of Independent and Graded Concept Descriptions. In *Proceedings of the Sixth International Workshop on Machine Learning.* Ithaca, NY: Morgan Kaufmann, 387–391.

Aha, D. W. (1991). Incremental Constructive Induction: An Instance-Based Approach. In *Proceedings of the Eighth International Workshop on Machine Learning.* Evanston, IL: Morgan Kaufmann, 117–121.

Aha, D. W. (1992). Generalizing from Case Studies: A Case Study. In *Proceedings of the Ninth International Conference on Machine Learning.* Aberdeen, Scotland: Morgan Kaufmann, 1–10.

Aha, D. W. (1990). *A Study of Instance-Based Learning Algorithms for Supervised Learning Tasks: Mathematical, Empirical, and Psychological Evaluations* (Technical Report 90-42). Irvine, CA: University of California, Department of Information and Computer Science.

Aha, D. W., and Goldstone, R. L. (1992). Concept Learning and Flexible Weighting. In *Proceedings of the Fourteenth Annual Conference of the Cognitive Science Society.* Bloomington, IN: Lawrence Erlbaum, 534–539.

Aha, D. W., Kibler, D., and Albert, M. K. (1991). Instance-Based Learning Algorithms. *Machine Learning* **6**, 37–66.

Bakiri, G. (1991). *Converting English Text to Speech: A Machine Learning Approach.* Doctoral dissertation, Department of Computer Science, Oregon State University, Corvallis, Oregon.

Bareiss, R. (1989). The Experimental Evaluation of a Case-Based Learning Apprentice. In *Proceedings of a Case-Based Reasoning Workshop.* Pensacola Beach, FL: Morgan Kaufmann, 162–167.

Birnbaum, L. A., and Collins, G. C. (Ed.). (1991). *Proceedings of the Eighth International Workshop on Machine Learning.* Evanston, IL: Morgan Kaufmann.

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1984). *Classification and Regression Trees.* Belmont, CA: Wadsworth International Group.

Cain, T., Pazzani, M. J., and Silverstein, G. (1991). Using Domain Knowledge to Influence Similarity Judgement. In *Proceedings of the Case-Based Reasoning Workshop.* Washington, DC: Morgan Kaufmann, 191–202.

Cardie, C. (1993). Using Decision Trees to Improve Case-Based Learning. In *Proceedings of the Tenth International Conference on Machine Learning.* Amherst, MA: Morgan Kaufmann, 25–32.

Clark, P. E., and Boswell, R. (1991). Rule Induction with CN2: Some Recent Improvements. In *Proceedings of the Fifth European Working Session on Learning.* Porto, Portugal: Springer-Verlag, 151–163.

Clark, P., Feng, C., and Matwin, S. (1993). *Design for a Case-Based Expert-System for Remote Sensing* (Technical Report 93-08). University of Ottawa, Department of Computer Science.

Clark, P. E., and Niblett, T. (1989). The CN2 Induction Algorithm. *Machine Learning* **3**, 261–284.

Cover, T. M., and Hart, P. E. (1967). Nearest Neighbor Pattern Classification. *IEEE Trans. Inf. Theory* **13**, 21–27.

Cover, T. M. (1968). Estimation by the Nearest Neighbor Rule. *IEEE Trans. Inf. Theory* **14**, 50–55.

Dasarathy, B. V. (Ed.). (1991). *Nearest Neighbor(NN) Norms: NN Pattern Classification Techniques.* Los Alamitos, CA: IEEE Computer Society Press.

Duda, R. O., and Hart, P. E. (1973). *Pattern Classification and Scene Analysis.* New York, NY: Wiley.

Fala, G. and Walker, J. (1993). Using Case-Based Reasoning to Automate Acoustic Submarine Classification: Detailed Summary of Technical Progress. In *ONR Computer Science Division Program Summary, Fiscal Year 1992.*

Fix, E., and Hodges, J. L., Jr. (1951). *Discriminatory Analysis, Nonparametric Discrimination, Consistency Properties* (Technical Report 4). Randolph Field, TX: United States Air Force, School of Aviation Medicine.

Hintzman, D. L. (1988). Judgments of Frequency and Recognition Memory in a Multiple-Trace Memory Model. *Psychological Review* **95**, 528-551.

Jabbour, K., Riveros, J. F. V., Landsbergen, D., and Meyer W. (1987). ALFA: Automated Load Forecasting Assistant. In *Proceedings of the 1987 IEEE Power Engineering Society Summer Meeting.* San Francisco, CA.

Kelly, J. D., Jr., and Davis, L. (1991). A Hybrid Genetic Algorithm for Classification. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence.* Sydney, Australia: Morgan Kaufmann, 645–650.

Kibler, D., and Aha, D. W. (1987). Learning Representative Exemplars of Concepts: An Initial Case Study. In *Proceedings of the Fourth International Workshop on Machine Learning.* Irvine, CA: Morgan Kaufmann, 24–30.

Nosofsky, R. M. (1986). Attention, Similarity, and the Identification-Categorization Relationship. *J. Experimental Psychology: General* **15**, 39–57.

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning.* San Mateo, CA: Morgan Kaufmann.

Reed, S. K. (1972). Pattern Recognition and Categorization. *Cognitive Psychology* **3**, 382–407.

Rumelhart D. E., McClelland, J. L., and The PDP Research Group (Eds.), (1986). *Parallel Distributed Processing: Explorations in the Microstructure of Cognition* (Vol. 1). Cambridge, MA: MIT Press.

Salzberg, S. L. (1990). *Learning with Nested Generalized Exemplars.* Boston, MA: Kluwer.

Schaffer, C. (1993). Overfitting Avoidance as Bias. *Machine Learning* **10**, 113–152.

Sebestyen, G. S. (1962). *Decision-Making Processes in Pattern Recognition.* New York, NY: Macmillan.

Shavlik, J. W., Mooney, R. J., and Towell, G. G. (1991). Symbolic and Neural Learning Algorithms: An Experimental Comparison. *Machine Learning* **5**, 111–145.

Shepard, B. (1983). An Appraisal of a Decision Tree Approach to Image Classification. In *Proceedings of the Eighth International Joint Conference on Artificial Intelligence.* Karlsruhe, West Germany: William Kaufmann, 473–475.

Shepard, R. N. (1987). Toward a Universal Law of Generalization for Psychological Science. *Science* **237**, 1317–1323.

Stanfill, C., and Waltz, D. (1986). Toward Memory-Based Reasoning. *Commun. Assoc. Computing Machinery* **29**, 1213–1228.

Turney, P. D. (1993). Exploiting Context when Learning to Classify. In *Proceedings of the European Conference on Machine Learning.* Vienna, Austria: Springer-Verlag, 402–407.

Tversky, A. (1977). Features of Similarity. *Psychological Review* **84**, 327–352.

Utgoff, P. E. (1986). *Machine Learning of Inductive Bias.* Hingham, MA: Kluwer.

Walker, J. (1993). Personal communication.

Weiss, S. M., and Kulikowski, C. A. (1991). *Computer Systems that Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems.* San Mateo, CA: Morgan Kaufmann.