

An Efficient Hybrid Classification Algorithm - an Example from Palliative Care

Tor Gunnar Houeland, Agnar Aamodt

Department of Computer and Information Science,
Norwegian University of Science and Technology,
NO-7491 Trondheim, Norway
{houeland,agnar}@idi.ntnu.no

Abstract. In this paper we present an efficient hybrid classification algorithm based on combining case-based reasoning and random decision trees, which is based on a general approach for combining lazy and eager learning methods. We use this hybrid classification algorithm to predict the pain classification for palliative care patients, and compare the resulting classification accuracy to other similar algorithms. The hybrid algorithm consistently produces a lower average error than the base algorithms it combines, but at a higher computational cost.

Keywords: hybrid reasoning systems, classifier combination, case-based reasoning, random decision trees

1 Introduction

Case-based reasoning (CBR), including instance-based methods, represents a unique approach to learning and problem solving compared to generalization-based methods. It is therefore often a choice of one method in a hybrid system, complementary to generalization-based and inductive methods. Examples include using an ensemble of different inductive methods to perform adaptation in CBR [12], and a neural network approach for adaptation, revision, and retention of cases [5]. As a lazy learning method that postpones the generalization step until problem solving time [1], CBR has the advantage of including contextual information that an eager approach would not have access to, thereby adapting the reasoning to the particular characteristics of the problem to solve. Eager methods, on the other hand, have the advantage that parts of the problem solving behaviour can be precomputed during training, which enables reduced storage space and faster query processing.

A path of our research is to explore the combination of model-based reasoning, starting from a predefined model that make some top-down commitments, with case-based reasoning that make very few high level commitments but rather grows its knowledge base (case base) in a bottom-up fashion. An example is the combination of Bayesian Networks with case-based reasoning [4]. In this paper we examine a hybrid approach that uses a modified version of an eager method,

Random Decision Trees (RDT), that can be partially precomputed and partially adapted to the particular problem query.

As of today, there is no consensus about the set of classes that should be used for pain classification in palliative care [9]. Our domain is open and changing, which is why we study methods of machine learning and decision support that are able to produce useful results without making very strong commitments about the domain.

In an earlier study we examined the problem of determining case similarity in our palliative care domain, and created a hybrid RDT approach to locate the most similar case in the case base [7]. In the work presented here we extend our approach by developing algorithms for classifying cases. We do this by predicting the *average pain* and *worst pain* values for the third week after the first consultation, based on the information collected for the first two weeks. These values are important because the objective is to minimize the patient’s pain, and the doctor’s approach for relieving the patient’s pain is applied in full during the second week. In the third and following weeks, the patient is mainly observed and pain medication is modified according to needs.

In the next section we review some earlier relevant research, which is followed in section 3 by a description of our RDT-based experiment and the algorithms used in the experiment. In section 4 we compare the algorithms and their parameters and discuss empirical results from running the algorithms on a case base of palliative care patients. Concluding remarks end the paper.

2 Related Research

Studies of ensembles of random decision trees have been extensive. Among the most well-known is the Random Forest (RF) classifier [3] which grows a number of trees based on bootstrap samples of the training data. For each node of a tree, m variables are randomly chosen and the best split based on these m variables is calculated based on the bootstrap sample. Each decision tree results in a classification and is said to cast a vote for that classification. The ensemble classifier returns the class that receives the most votes. RF can also compute *proximities* between pairs of cases that can be used for clustering and data visualization, and hence as similarity measures for case-based reasoning.

In a thorough study of ensemble method types it was found that the performance of an ensemble learning approach varies substantially across applications. Bian et. al. [2] studied homogeneous and heterogeneous ensembles and found connections between diversity and performance, and an increased diversity for heterogeneous ensembles.

A contribution to the analysis of the laziness vs. eagerness distinction, which corresponds with the distinction between global and local approximations to the target function, was made by Hendrickx and den Bosch [6]. They studied several hybrid methods as well as their single components. The analysis showed that the k-NN method outperformed the eager methods, while the best hybrid methods outperformed any single methods on combined generalization performance and

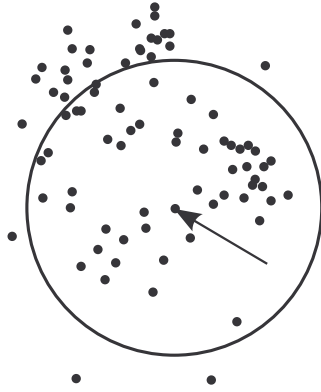


Fig. 1. Using a similarity-based local case subset as training data. A set of neighbors of the marked problem query to solve are shown as the cases that lie within the circle. That set is used to train an independent learning algorithm.

statistical error bias. A combined approach for optimizing the combined learning and classification time of lazy and eager learners was developed by Mohebpour et.al. [11], a problem also addressed by Veloso and Meira [14].

A particular problem relates to the utility of learned knowledge. The "utility problem" occurs when additional knowledge learned decreases a reasoning system's performance instead of increasing it [10, 13]. Theoretically this will always occur in a CBR system when the system's case base increases without bound. The utility problem is not necessarily observed in practice for real-world CBR systems with moderately-sized case bases, however.

Based on one of our own studies [8], we suggest that the usefulness of an optimization should be measured by the effect it has on the reasoning system's overall utility. We measured an example system's total solution time to show that case base size reduction methods can be counterproductive because the methods were more computationally demanding than simply reasoning using the larger unreduced case base.

3 Random Decision Tree Classification Experiment

The hybrid random decision tree (RDT) algorithm presented here is an approach to combining machine learning methods with case-based reasoning. We retrieve the most similar half of the available cases using a domain-specific relevance measure (a general illustration of this approach is shown in figure 1). We then run our RDT algorithm as a computationally efficient machine learner using this subset of cases as training data. This approach combines the lazy and locally specific characteristics of a CBR retrieval with the more eager and global characteristics often seen in traditional machine learning algorithms.

In the presented research we expand the use of our RDT algorithm from being a pure similarity measure to also predicting the classification of unseen

cases. As part of its internal computations, each decision tree in our algorithm is partitioning the cases in the case base between its leaf nodes. This is conceptually similar to how indexing trees used for efficient retrieval in CBR are constructed. We exploit this insight to create a classification algorithm where each tree classifies a new problem query based on the previous cases that lead to the same leaf node as the new problem query.

If each tree classifies cases as the arithmetic mean of the classification of previous cases in the same leaf node, and the average of each tree is used as the combined classification, then it is not necessary to enumerate the specific subsets of cases present in each leaf node. It is sufficient to know *how many times* each case shares a leaf node with the problem query, and then the combined classification can be determined by taking the weighted average, where each case is assigned a weight equal to the number of times it shares a leaf node with the problem query.

The number of times a case shares a leaf node with the problem query is precisely the *proximity* of the case, for which we have previously developed an efficient computational method while exploring the use of RDTs to determine similarity [7].

Using this proximity-weighted averaging approach, we have implemented a purely RDT-based classifier and a hybrid RDT+CBR classifier. We explore their characteristics related to the palliative pain classification domain. For comparison purposes we also test a k -NN classifier corresponding to the CBR part of the hybrid, and a simple and very fast algorithm based only on averaging. We compare the results obtained from these algorithms according to their computational complexity.

Our data set consists of 1486 cases with numerical features based on patients in the palliative care domain. The problem description we use for input queries to be solved consists of 55 numerical features based on measurements and classifications obtained during the first two weeks after the first consultation. Examples of these features include the patient’s age, the reported average pain for week 1 on a scale from 0-10, the total opioid dose given as pain relief for week 2 as a floating point number, and similar features for other aspects such as insomnia, cognitive functioning and use of antidepressants. As the solution to predict we use 2 classifications related to the patient’s pain for the third week: the reported *average pain* and *worst pain* on scales from 0-10.

3.1 Algorithms

COMPUTED-AVERAGE computes the mean *average pain* and *worst pain* values based on the cases encountered so far, and uses these computed means as the predicted classifications for the new problem query. It is a simple and fast algorithm which only learns from the problem solutions. It performs this limited task well, and is used as a baseline comparison for the other algorithms which attempt to also learn domain knowledge from the more complicated problem descriptions.

CBR- k -NN selects the average of the k most similar previously encountered cases. Similarity is measured using a simple CBR-DIFFERENCE-MEASURE function that was provided as a rough relevance estimate. This estimate is based on differences in 8 values in the data set that correspond to the variables a domain expert considers most important. For $k = 1$ this is the same as retrieving and copying the solution from the most similar case, while $k \geq 2$ performs averaging as a simple and knowledge-lean multi-case adaptation step during reuse.

N -RANDOMTREES-CLASSIFIER is based on our presented approach for classification by efficiently evaluating random decision trees on case subsets. N trees are grown and the *average pain* and *worst pain* values are predicted as the average of all cases in the case base weighted by their computed proximity to the problem query.

N -HYBRID-CLASSIFIER is our hybrid combination of the CBR relevance measure and using our RDT algorithm for classification. For every input problem query, the CBR-DIFFERENCE-MEASURE function is used to narrow the case base down to the most similar half. Then N trees are used to compute the average and worst pain as in the N -RANDOMTREES-CLASSIFIER algorithm, but based only on the cases from this most relevant half of the case base.

4 Results and Discussion

To achieve a fair comparison we generate 10 versions of the input where the same patient cases are used, but in 10 different randomly shuffled orders. We evaluate the algorithms by their average result from each of these modified case bases. We use this approach because the results of a single run-through of the case base can vary, due to intrinsic randomness in the RDT-based algorithms and differences caused by the order in which the cases are presented. For each algorithm we measure the root mean square error (RMSE) for solving each of the 10 permuted case bases, and report the average RMSE value.

Figure 2 shows the measured average root mean square error for the different algorithms, compared according to the time (computational resources) required. The result for the COMPUTED-AVERAGE algorithm is shown as a single point, as there is no varying parameter and the execution time for a given set of inputs remains constant apart from small random fluctuations in the computing environment. The exact time required depends on the type of computing device that is used to run the algorithms, but we focus on the relative differences between these algorithms which is primarily determined by their computational complexity.

The results for CBR- k -NN are not as sensitive to the exact value of k as an initial reading of the graph might suggest, because the value of k has a relatively small effect on the time required to run the algorithm. In fact the visible line for CBR- k -NN in the graph spans from around $k = 5$ to $k = 1000$.

Additional details are shown in table 1, with numerical values for a subset of the results. The results shown in the table are marked as points in figure 2.

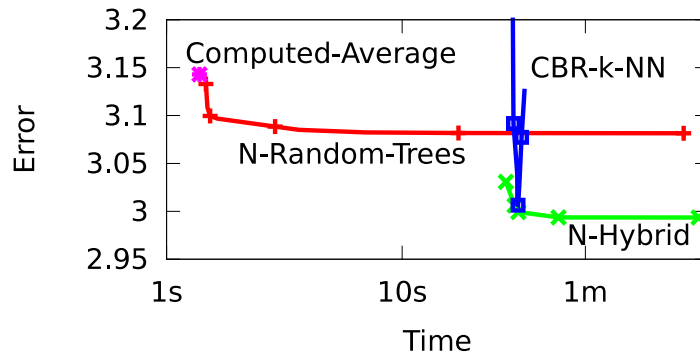


Fig. 2. Experimental results for the different algorithms and parameters, compared to the computational resources required. (Lower error and faster time is better.)

The underlying CBR-DIFFERENCE-MEASURE function is not in itself particularly potent as a direct similarity measure. CBR-1-NN produces an error of 4.14. This is worse than a trivial classifier that always predicts 5 as the solution, which produces an error of 3.62 using the same experimental setup. However, the variables identified by the domain expert are indeed relevant, as a completely random similarity measure that retrieves a case at random produces an error of 4.46.

This indicates that the similarity measure is helpful for locating the most relevant cases, but that predicting the pain values based on only a single similar patient is unlikely to work well in this domain. A relatively large k value of around 75 produces the best result for the CBR- k -NN algorithm in this experiment.

For our RDT approaches a higher number of trees N produces better results. Unlike how k affects CBR- k -NN, there is no particular sweet spot for N in either the RDT trees or the hybrid approach above which the results start deteriorating. However, the improvements flatten out to become negligible compared to the increase in computational resources required when using more than around 1000 trees.

N-HYBRID-CLASSIFIER has the lowest overall error but a comparatively high computation cost, while *COMPUTED-AVERAGE* and *N-RANDOM-TREES-CLASSIFIER* are good choices to produce results very quickly.

This illustrates an important trade-off between speed and accuracy when choosing a classifier. In this experiment, our approach to combining lazy and eager classifiers to make a hybrid classifier produced better predictions, but at an increased computational cost. Whether the increased accuracy is worth the additional complexity and increased resource cost depends on the exact application and usage of the reasoning system. Given a time limit for a particular application, the algorithm that produces the best results can e.g. be determined as the lowest line at that point in a graph such as the one shown in figure 2.

Table 1. Numerical results for our algorithms in the palliative care domain, showing the computation time required and the average root mean square error.

Algorithm	Time	Error
COMPUTED-AVERAGE	1.4 seconds	3.14
CBR-1-NN	26 seconds	4.14
CBR-10-NN	30 seconds	3.09
CBR-75-NN	31 seconds	3.01
CBR-500-NN	32 seconds	3.08
1-RANDOM-TREES-CLASSIFIER	1.5 seconds	3.13
10-RANDOM-TREES-CLASSIFIER	1.6 seconds	3.10
100-RANDOM-TREES-CLASSIFIER	2.9 seconds	3.09
1000-RANDOM-TREES-CLASSIFIER	17 seconds	3.08
10000-RANDOM-TREES-CLASSIFIER	150 seconds	3.08
1-HYBRID-CLASSIFIER	28 seconds	3.03
10-HYBRID-CLASSIFIER	30 seconds	3.01
100-HYBRID-CLASSIFIER	31 seconds	3.00
1000-HYBRID-CLASSIFIER	46 seconds	2.99
10000-HYBRID-CLASSIFIER	180 seconds	2.99

5 Conclusions and further research

In this paper we have presented an approach for classifying unseen cases in the palliative care domain by extending our efficiently computable random decision tree (RDT) algorithm. We have developed methods for predicting the *average pain* and *worst pain* values for palliative care patients. We used a case-based k -NN method using a domain-specific relevance measure, a knowledge-lean implementation of our RDT method and a hybrid combination of the relevance measure and the RDT approach. The base RDT approach produced results very quickly, while the hybrid approach produced better results than either of the base algorithms at a comparable computational cost to running the k -NN method.

In the palliative care domain, where patients receive treatment over several months and a better result can potentially result in reduced suffering, using the best possible algorithm is usually worthwhile. However, in this domain, increasing the parameter for the number of trees in the hybrid algorithm above around 1000 increases the computational cost with negligible improvements in accuracy.

In our ongoing and future work, we are experimenting with using meta-level reasoning as part of the problem solving process. Our goal is to automatically determine which algorithm produces the best results for a given data set, and to use that algorithm for solving future problem queries.

Acknowledgments This research is partly conducted within the project TL-CPC (Transactional Research in Lung Cancer and Palliative Care), a nationally funded project in cooperation with the Medical Faculty of our university and the St. Olav Hospital in Trondheim.

We wish to thank Cinzia Brunelli for providing the data set, Anne Kari Knudsen for interpreting and analysing the data from a clinical perspective, and Tore Bruland for his analysis of the data from a data structure and machine learning perspective.

References

1. Aamodt, A., Plaza, E.: Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI Communications* 7(1), 39–59 (March 1994), <http://portal.acm.org/citation.cfm?id=196108.196115>
2. Bian, S., Wang, W.: On diversity and accuracy of homogeneous and heterogeneous ensembles. *Int. J. Hybrid Intell. Syst.* 4, 103–128 (April 2007), <http://portal.acm.org/citation.cfm?id=1367006.1367010>
3. Breiman, L.: Random forests. *Machine Learning* 45, 5–32 (2001), <http://dx.doi.org/10.1023/A:1010933404324>
4. Bruland, T., Aamodt, A., Langseth, H.: Architectures integrating case-based reasoning and bayesian networks for clinical decision support. In: Shi, Z., Vadera, S., Aamodt, A., Leake, D.B. (eds.) *Intelligent Information Processing. IFIP*, vol. 340, pp. 82–91. Springer (2010)
5. Corchado, J.M., Lees, B., Aiken, J.: Hybrid instance-based system for predicting ocean temperatures. *International Journal of Computational Intelligence and Applications* pp. 35–52 (2001)
6. Hendrickx, I., van den Bosch, A.: Hybrid algorithms with instance-based classification. In: Gama, J., Camacho, R., Brazdil, P., Jorge, A., Torgo, L. (eds.) *ECML. LNCS*, vol. 3720, pp. 158–169. Springer (2005)
7. Houeland, T.G.: An efficient random decision tree algorithm for case-based reasoning systems. In: *FLAIRS Conference. AAAI Press* (2011), to appear
8. Houeland, T.G., Aamodt, A.: The utility problem for lazy learners - towards a non-eager approach. In: Bichindaritz, I., Montani, S. (eds.) *Case-Based Reasoning. Research and Development, LNCS*, vol. 6176, pp. 141–155. Springer (2010), <http://dx.doi.org/10.1007/978-3-642-14274-1>, [10.1007/978-3-642-14274-1](http://dx.doi.org/10.1007/978-3-642-14274-1)
9. Knudsen, A., Aass, N., Fainsinger, R., Caraceni, A., Klepstad, P., Jordhy, M., Hjermsstad, M., Kaasa, S.: Classification of pain in cancer patients a systematic literature review. *Palliative Medicine* 23(4), 295–308 (2009), <http://pmj.sagepub.com/content/23/4/295.abstract>
10. Minton, S.: Quantitative results concerning the utility of explanation-based learning. *Artif. Intell.* 42(2-3), 363–391 (1990)
11. Mohebpour, M.R., Adznan B. J., Saripan, M.I.: Grid Base Classifier in Comparison to Nonparametric Methods in Multiclass Classification. *Pertanika J. Sci. & Technol.* 18(1), 139–154 (2010)
12. Policastro, C., Delbem, A., Mattoso, L., Minatti, E., Ferreira, E., Borato, C., Zanus, M.: A Hybrid Case Based Reasoning Approach for Wine Classification. *ISDA* pp. 395–400 (2007)
13. Smyth, B., Cunningham, P.: The utility problem analysed - a case-based reasoning perspective. In: *Proceedings of the Third European Workshop on Case-Based Reasoning*. pp. 392–399. Springer Verlag (1996)
14. Veloso, A., Meira, Jr., W.: Eager, lazy and hybrid algorithms for multi-criteria associative classification. In: *Proceedings of the Data Mining Algorithms Workshop. Uberlandia, MG.* (2005)