Engineering Applications of Artificial Intelligence 35 (2014) 71–83

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Improved decision tree construction based on attribute selection and data sampling for fault diagnosis in rotating machines

Nour El Islem Karabadji [a,*], Hassina Seridi [a], Ilyes Khelf [b], Nabiha Azizi [a], Ramzi Boulkroune [c]

[a] Electronic Document Management Laboratory (LabGED), Badji Mokhtar-Annaba University, P.O. Box 12, 23000 Annaba, Algeria
[b] Laboratoire de Mécanique Industrielle, Badji Mokhtar-Annaba University, P.O. Box 12, 23000 Annaba, Algeria
[c] URASM-CSC/ ANNABA Unité de recherche Appliquée en Sidérurgie Métallurgie Centre National de Recherche Scientifique et Technique en Soudage et Contrôle, Algeria

ARTICLE INFO

Article history:
Received 2 January 2014
Received in revised form
16 June 2014
Accepted 16 June 2014
Available online 8 July 2014

Keywords:
Decision tree construction
Pruning
Research graph
Attribute selection
Data sampling

ABSTRACT

This paper presents a new approach that avoids the over-fitting and complexity problems suffered in the construction of decision trees. Decision trees are an efficient means of building classification models, especially in industrial engineering. In their construction phase, the two main problems are choosing suitable attributes and database components. In the present work, a combination of attribute selection and data sampling is used to overcome these problems. To validate the proposed approach, several experiments are performed on 10 benchmark datasets, and the results are compared with those from classical approaches. Finally, we present an efficient application of the proposed approach in the construction of non-complex decision rules for fault diagnosis problems in rotating machines.

1. Introduction

In the industrial field, the risks of failure and disruption are increasing with the complexity of installed equipment. This phenomenon affects product quality, causes the immediate shut-down of a machine, and undermines the proper functioning of an entire production system. Rotating machines are a major class of mechanical equipment, and need the utmost care and continuous monitoring to ensure optimal operation. Traditionally, vibration analyses and many signal processing techniques have been used to extract useful information for monitoring the operating condition. Khelf et al. (2013) analysed the frequency domain to extract information and diagnose faults. Cepstral analysis has been used to construct a robust gear fault indicator (Badaoui et al., 2004), and a short-time Fourier transform representation was derived (Mosher et al., 2003). Other techniques have also been employed, such as the Wigner–Ville distribution (Baydar and Ball, 2001), continuous wavelet analysis (Kankar et al., 2011), and discrete wavelet analysis (Djebala et al., 2008).

Classification algorithms can be used in the construction of condition-monitoring diagnostic systems. For example, neural networks (Chen and Chen, 2011), support vector machines (Deng et al., 2011), and Bayesian classifiers (Yang et al., 2005) have all been applied. However, decision tree techniques are still preferred in engineering applications, because they allow users to easily understand the behaviour of the built models against the above-mentioned classifiers. Their use in such applications has been reported in numerous research papers, e.g. Sugumaran and Ramachandran (2007), Zhao and Zhang (2008), Sakthivel et al. (2010), and Sugumaran et al. (2007).

The construction of a decision tree (DT) includes growing and pruning stages. In the growing phase, the training data (samples) are repeatedly split into two or more descendant subsets, according to certain split rules, until all instances of each subset wrap the same class (pure) or some stopping criterion has been reached. Generally, this growing phase outputs a large DT that includes the learning examples and considers many uncertainties in the data (particularity, noise and residual variation). Pruning approaches based on heuristics prevent the over-fitting problem by removing all sections of the DT that may be based on noisy and/or erroneous data. This reduces the complexity and size of the DT. The pruning phase can under-prune or over-prune the grown DT. Moreover, many existing heuristics are very challenging (Breiman et al., 1984; Niblett and Bratko, 1987; Quinlan, 1987), but, unfortunately, no single method outperforms the others (Mingers, 1989; Esposito et al., 1997).

In terms of growing phase problems, there are two possible solutions: the first reduces DT complexity by reducing the number of learning data, simplifying the decision rules (Piramuthu, 2008).

* Corresponding author.
E-mail addresses: karabadji@labged.net (N.E.I. Karabadji), seridi@labged.net (H. Seridi), ilyeskhelf@gmail.com (I. Khelf), azizi@labged.net (N. Azizi), ramzi86@hotmail.com (R. Boulkroune).

http://dx.doi.org/10.1016/j.engappai.2014.06.010
0952-1976/

The second solution uses attribute selection to overcome over-fitting problems (Yildiz and Alpaydin, 2005; Kohavi and John, 1997). To overcome both the DT size and over-fitting risks, we propose to combine attribute selection and data reduction to construct an Improved Unpruned Decision Tree $\mathcal{IUDT}$. The optimal DT construction (DTC) problem will thus be converted into an exploration of the combinatorial graph research space problem. The key feature of this proposition is to encode each subset of attributes A_i and a samples subset X_j into a couple (A_i, X_j). All possible (A_i, X_j) couples form the research space graph. The results show that the proposed schematic largely improves the tree performance compared to standard pruned DTs, as well as those based solely on attribute selection or data reduction.

The rest of the paper is organized as follows: In Section 2, some previous studies on DTC are briefly discussed. Section 3 introduces the main notions used in this work. In Section 4, we describe our approach based on attribute selection and database sampling to outperform conventional DTC. Section 5 reports the experimental results using 10 benchmark datasets. In Section 6, $\mathcal{IUDT}$ is applied to the problem of fault diagnosis in rotating machines. Finally, Section 7 concludes the study.

2. Related work

This section describes post-pruning approaches that have been proposed to improve DTC. Their common aim was to decrease (1) the tree complexity and (2) the error rate of an independent test dataset. Pruning methods have various differences that can be summarized as follows:

1. the necessity of the test dataset;
2. the generation of a series of pruned sub-trees or the processing of a single tree;
3. the pruning determination criteria.

Breiman et al. (1984) developed error-complexity pruning, which uses the cost-complexity risk. The pruning measure uses an error rate penalty based on the sub-tree size. The errors and the size of the tree's leaves (complexity) are both considered in this pruning method. The cost-complexity risk measurement of all possible sub-trees in an initial DT T_0 is calculated as the training error $R(t)$ added to the product of a factor α and the number of leaves $|\tilde{t}|$ in the sub-tree t, i.e. $RC_\alpha(t) = R(t) + \alpha(|\tilde{t}|)$. A series of sub-decision trees with the smallest value of α are selected to be pruned. Finally, the correctly pruned sub-tree t is selected from the α sequence of sub-trees using an independent test dataset. The final selection is based on the error rate or standard error (assuming a binomial distribution).

Reduced-error pruning, proposed by Quinlan (1987), produces a series of pruned DTs using the test dataset. A complete DT T_0 is first grown using the training dataset. A test dataset is then used, and for each node in T_0, the number of classification errors made on the pruning set when the sub-tree t is kept is compared with the number of classification errors made when t is turned into a leaf. Next, the positive difference between the two errors is assigned to the sub-tree root node. The node with the largest difference is then pruned. This process is repeated until the pruning increases the misclassification rate. Finally, the smallest version of the most accurate tree with respect to the test dataset is generated.

In contrast to reduced-error pruning, the necessity of separate test datasets can be avoided using pessimistic error pruning (PEP, Quinlan, 1987). This uses the binomial continuity correction rate to obtain a more realistic estimate of the misclassification rate. The misclassification correction depends on the number of leaves and misclassifications.

Error-based pruning (EBP, Quinlan, 1993) is an improved version of PEP that traverses the tree according to a bottom-up post-order strategy. No pruning dataset is required, and the binomial continuity correction rate of PEP is used. Therefore, the difference is that, in each iteration, EBP considers the possibility of grafting a branch t_y in place of the parent of y itself. The estimation errors t_x, t_y are calculated to determine whether it is convenient to prune node x (the tree rooted by x replaced by a leaf), replace it with t_y (the largest sub-tree), or keep the original t_x.

Recently, Luo et al. (2013) developed a new pruning method based on the structural risk of the leaf nodes. This method was developed under the hypothesis that leaves with high accuracies mean that the tree can classify the training data very well, and a large volume of such leaves implies generally good performance. Using this hypothesis, the structural risk measures the product of the accuracy and the volume of leaf nodes. As in common pruning methods, a series of sub-trees are generated. The process visits each node x on DT T_0 (t_x is a sub-tree whose root is x). For each sub-tree t_x, feasible pruning nodes are found (their two children are leaves), and the structural risks are measured. Finally, the sub-tree that maximizes the structural risk is selected for pruning.

Additional post-pruning methods have been proposed, such as critical value pruning (Mingers, 1987), minimum error pruning (Niblett and Bratko, 1987), and DI pruning (which balances both the Depth and the Impurity of nodes) (Fournier and Crémilleux, 2002). The choice of DT has also been validated (Karabadji et al., 2012), and genetic algorithms used to pull out the best tree over a set of different models (e.g. BFTree, J48, LMT, Hall et al., 2009). To select the most robust DT, all models were generated and their performances measured on distinct training and validation sets. In this work, the main objective is to construct DTs without under-pruning or over-fitting the training dataset, and without choosing between different pruning methods. Two prior works have shown that unpruned DTs give similar results to pruned trees when a Laplace correction is used to calculate the class probabilities (Bradford et al., 1998; Provost and Domingos, 2003).

The identification of smaller sets of highly predictive attributes has been considered by many learning schemes. Attribute selection shares the same objective as pruning methods, namely the elimination of irrelevant, redundant, and noisy attributes in the building phase to produce good DT performance. Many studies have investigated and improved classification models (Bermejo et al., 2012; Macaš et al., 2012). In these works, wrapper techniques have been applied to attribute selection. A target learning algorithm is used to estimate the value of attribute subsets. The process is driven by the binary relation "$\subseteq$" between attribute subsets. The search process can be conducted on a depth-first or breadth-first basis, or a combination of both (e.g. "A star" (A*) algorithm). Wrappers are generally better than filters, but the improved performance comes at a computational cost—in the worst case, 2^m subsets of attributes must be tested (m is the number of attributes) (Kohavi and John, 1997).

Similar to attribute selection, DTs can be improved by reducing the data complexity, as well as reducing the effects of unwanted data characteristics. Data reduction essentially involves dimensionality reduction and/or example reduction (Piramuthu, 2008). Generally, reduction methods use sampling (e.g. random, stratified) to select examples for consideration in the learning phase (Ishibuchi et al., 2001; Liu, 2010).

In conclusion, different pruning techniques have been studied, but none is adequate for all varieties of problem. There has been a recent focus on attribute selection and sampling data to improve DTC. To realize a better DT for a specific application, we propose the $\mathcal{IUDT}$ algorithm, which combines a novel scheme of random