Expert Systems with Applications 38 (2011) 5036–5043

Contents lists available at ScienceDirect

Expert Systems with Applications

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

Identifying product failure rate based on a conditional Bayesian network classifier

Zhiqiang Cai [a,*], Shudong Sun [a], Shubin Si [a], Bernard Yannou [b]

[a] Ministry of Education, Key Laboratory of Contemporary Design and Integrated Manufacturing Technology, School of Mechantronics, Northwestern Polytechnical University, Xi'an 710072, China
[b] Laboratoire Genie Industriel, Ecole Centrale Paris, Chatenay-Malabry 92290, France

ARTICLE INFO

Keywords:
Maintenance management
Failure rate
Classifier
Bayesian network
Conditional independence

ABSTRACT

To identify the product failure rate grade under diverse configuration and operation conditions, a new conditional Bayesian networks (CBN) model is brought forward. By indicating the conditional independence relationship between attribute variables given the target variable, this model could provide an effective approach to classify the grade of failure rate. Furthermore, on the basis of the CBN model, the procedure of building product failure rate grade classifier is elaborated with modeling and application. At last, a case study is carried out and the results show that, with comparison to other Bayesian networks classifiers and traditional decision tree C4.5, the CBN model not only increases the total classification accuracy, but also reduces the complexity of network structure.

1. Introduction

In recent years, maintenance has been playing a more and more important role in industrial fields due to the high demand for system safety, operational efficiency and life cycle cost control. In China, we cooperated with some aircraft corporations to develop a maintenance management system for years. This maintenance system daily collects bulky failure data during the airplane operation which is in different formats. The challenge faced currently is how to discover the potential failure knowledge from these data for prediction and decision making.

Data mining, which is also referred to as knowledge discovery, means the process of extracting nontrivial, implicit, previously unknown and potentially useful information from databases (Witten & Frank, 2005). Depending on the types of knowledge derived, mining approaches may be classified as association rules mining, clustering, classification, prediction and others. In the area of product failure data mining, it has been used widely for the purpose of failure prediction, failure classification, and failure association. Al-Garni, Jamal, Ahmad, Al-Garni, and Tozan (2006) developed an artificial neural network (ANN) model for predicting the failure rate of De Havilland Dash-8 airplane tires utilizing the two layer feed-forward back-propagation algorithm. Using 6 years of data, the results show that the failure rate predicted by the ANN is closer to the actual data than the failure rate predicted by the Weibull regression model. Chen, Tseng, and Wang (2005) defined the root-cause machine set identification problem of analyzing correlations between combinations of machines and the defective products and then proposed the Root-cause Machine Identifier (RMI) method using the technique of associating rule mining to solve the problem efficiently and effectively. Han, Kim, and Sohn (2007) applied sequential association rules to extract the failure patterns and forecast failure sequences of Republic of Korea Air Force (ROKAF) aircrafts for various combinations of aircraft types, location, mission and season, which could improve the utilization of aircrafts by properly forecasting the future demand of aircraft spare parts.

Because of the variety of each failure dataset and the diversity of each knowledge discovery mission, researchers have to build proper data mining models and processes according to the characteristic of target dataset and request. In this study, we limit the focus to product failure rate classification. Traditional product failure rate enactment is used to theoretically calculate the system reliability thanks to a static mathematical formula that ignores the actual application of each batch of products. Using the historical product failure data, we could provide a more accurate and effective classification of failure rate according to the configuration and operation. With such results, this model could satisfy the expectations of maintenance scheduling, spare parts supply chain management and product operation optimization.

From recent classification literature, with the characteristics of causality and conditional independence, the Bayesian networks (BN) have been recommended as a comprehensive method of indicating relationships among and influences of variables in system reliability domains (Boudali & Dugan, 2005; Langseth & Portinale, 2007; Mahadevan, Zhang, & Smith, 2001; Muller, Suhner, & Iung, 2008; Weber & Jouffe, 2006). It is a powerful technique for handling system uncertainty and it shows a high performance in

* Corresponding author. Tel.: +86 137 0924 7090.
E-mail addresses: caizhiqiang@nwpu.edu.cn, zhiqiangcai@hotmail.com (Z. Cai), sdsun@nwpu.edu.cn (S. Sun), sisb@nwpu.edu.cn (S. Si), bernard.yannou@ecp.fr (B. Yannou).

0957-4174/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2010.09.146

prediction and classification tasks. Friedman, Geiger, and Goldszmidt (1997) evaluated approaches for inducing classifiers from data, based on the theory of learning general Bayesian networks (GBN) and put forward a tree augmented Naïve Bayes (TAN) method, which outperforms Naïve Bayes (NB), yet at the same time maintains the computational simplicity and robustness that characterize NB. Cheng and Greiner (1999) learned BN augmented Naïve Bayes (BAN) and GBN using a conditional-independence (CI) based BN learning algorithm and evaluated the algorithms with NB and TAN. Experimental results show that the obtained classifiers are competitive with (or superior to) the other two classifiers. Madden (2002) introduced a new partial Bayesian network (PBN) and describes its constructing algorithm. The algorithm constructs an approximate Markov blanket around a classification node and the results indicate that PBN performs better than other Bayesian network classification structures on some problem domains. Because of the variety of collected data and application domains, researchers also have to focus on the individual case and choose the most effective classifier and modelling process. Baesens et al. (2004) compared and evaluated several Bayesian network classifiers with statistical and artificial intelligence techniques for the purpose of classifying customers in the binary classification problem. The experimental evidence showed that Bayesian network classifiers offer an interesting and viable alternative for customer lifecycle slope estimation problem.

This paper is organized as follows. In Sections 2.1 and 2.2, we discuss the principle of Bayesian networks and common Bayesian network classifiers. To deal with the weakness of present BN classifiers, a new conditional Bayesian network (CBN) classifier and its modeling process are described in Sections 2.3 and 2.4. In Section 3, the case study, the performance criteria and the comparison results are presented. Finally, Section 4 concludes the paper.

2. Modeling

2.1. Bayesian networks

Bayesian networks are directed acyclic graphs (DAGs) used to represent uncertain knowledge in artificial intelligence (Jensen, 1996). A Bayesian network is defined as a couple: BN = (S, $\mathbf{\Theta}$), where S = (**N**, **A**) represents the network structure.

N describes the set of all the nodes in a BN. Each node represents a discrete variable having a finite number of mutually exclusive states. In our example, a node may be failure cause, failure mode or other factors.

A is the set of all edges in a BN. Each edge represents the relationship of father and child by linking two nodes. In our example, an edge interprets as a causal relation such as failure cause node affects failure mode node.

$\mathbf{\Theta}$ represents the set of probability distributions that are associated with each node. When node is a root node (i.e. it does not have a parent), $\mathbf{\Theta}$ corresponds to the prior probability distribution of the node states. When a node is not a root node (i.e. when it has some parent nodes), $\mathbf{\Theta}$ corresponds to a conditional probability distribution that quantifies the probabilistic dependency between that node and its parents. It is represented by a conditional probability table (CPT).

Fig. 1 illustrates the nodes, edges and probability distribution through an example. The piston valve has one failure mode which is locked in a closed position. The high temperature and high vibration are two failure causes of valve locked closure, and their joint probability of leading to a locked valve is given by CPT. At last, the closure of valve will result in high gas pressure as a failure effect. From the CPT of high gas pressure, we can see that this node is separated from high temperature and high vibration by the node of valve locked close, which means they are conditionally independent.

Through the complex application of the Bayesian probability theory, Bayesian networks are designed to obtain probabilities of unknown variables from known probabilistic relationships. It is believed that they are well suited for prediction and classification research.

With the network structure and probability distributions mentioned in Fig. 1, it is convenient to compute the posterior probability of target variable. For example, according to the Bayesian theory $P(A/B) = \frac{P(B/A)P(A)}{P(B)}$, where $P(A)$ is prior probability, $P(B/A)$ is conditional probability, $P(A/B)$ is the posterior probability, we could compute $P(C = True/G = True)$ by $\frac{P(G=True/C=True)P(C=True)}{P(G=True)}$.

Then, we calculate the original variable probability distributions as follows:

$$\begin{aligned} P(C = True) &= \sum_{TE,V} P(TE, V, C = True) = \sum_{TE,V} P(TE)P(V)P(C \\ &= True/TE, V) = P(TE = True)P(V = True)P(C = True/TE \\ &= True, V = Ture) + P(TE = True)P(V = False)P(C \\ &= True/TE = True, V = False) + P(TE = False)P(V \\ &= True)P(C = True/TE = False, V = Ture) + P(TE \\ &= False)P(V = False)P(C = True/TE = False, V = False) \\ &= 0.318 \end{aligned} \tag{1}$$

$$\begin{aligned} P(G = True) &= \sum_{TE,V,C} P(TE, V, C, G = True) \\ &= \sum_{C} P(G = True/C) \sum_{TE,V} P(TE)P(V)P(C/TE, V) \\ &= \sum_{C} P(C)P(G = True/C) = P(C = True)P(G = True/C = Ture) \\ &\quad + P(C = False)P(G = True/C = False) = 0.3172 \end{aligned} \tag{2}$$

Finally, we got the posterior distribution easily as $P(C = True/G = True) = \frac{0.8\times0.318}{0.3172} = 0.802$.

In another case, suppose we have detected the evidence of high temperature and low vibration, the probability of bringing a high gas pressure is shown as:

$$\begin{aligned} P(G = true/TE = true, V = false) &= P(G = true/C = true) \\ &\quad \cdot P(C = true/TE = true, V = false) + P(G = true/C = false) \\ &\quad \cdot P(C = false/TE = true, V = false) \\ &= 0.8 * 0.4 + 0.1 * 0.6 = 0.38 \end{aligned} \tag{3}$$

But with the hypothesis of conditional independence, we could ignore the affections of (TE, V) on node G when C is detected. So, if we already know $(V = true, C = false)$, the result could be calculated directly as $P(G = True/V = True, C = False) = P(G = True/C = False) = 0.1$.

Because of the characteristics of causality and conditional independence, the Bayesian network provides a comprehensive method of representing relationships and influences among variables. It is also a powerful technique for handling uncertainty and shows a high performance in prediction domain. In addition, by presenting with graphical diagrams of nodes and edges, Bayesian network models can be more easily understood than many other techniques (Lee & Abbott, 2003).

2.2. Bayesian network classifiers

The application of Bayesian network classifiers is divided into two stages. First, the structure and parameters of Bayesian network are derived based on learning algorithm and some constraints. Secondly, the inference process is applied to compute the conditional probability of the target variable and classify it into certain classes based on the probability threshold. The time cost of classifier