Improving the Naïve Bayes Classifier

Liwei Fan
National University of Singapore, Singapore

Kim Leng Poh
National University of Singapore, Singapore

INTRODUCTION

A **Bayesian Network** (BN) takes a relationship between graphs and probability distributions. In the past, BN was mainly used for knowledge representation and reasoning. Recent years have seen numerous successful applications of BN in classification, among which the Naïve Bayes classifier was found to be surprisingly effective in spite of its simple mechanism (Langley, Iba & Thompson, 1992). It is built upon the strong assumption that different attributes are independent with each other. Despite of its many advantages, a major limitation of using the Naïve Bayes classifier is that the real-world data may not always satisfy the independence assumption among attributes. This strong assumption could make the prediction accuracy of the Naïve Bayes classifier highly sensitive to the correlated attributes. To overcome the limitation, many approaches have been developed to improve the performance of the Naïve Bayes classifier.

This article gives a brief introduction to the approaches which attempt to relax the independence assumption among attributes or use certain pre-processing procedures to make the attributes as independent with each other as possible. Previous theoretical and empirical results have shown that the performance of the Naïve Bayes classifier can be improved significantly by using these approaches, while the computational complexity will also increase to a certain extent.

BACKGROUND

The **Naïve Bayes classifier**, also called simple Bayesian classifier, is essentially a simple BN. Since no **structure learning** is required, it is very easy to construct and implement a Naïve Bayes classifier. Despite its simplicity, the Naïve Bayes classifier is competitive with other more advanced and sophisticated classifiers such as decision trees (Friedman, Geiger & Goldszmidt, 1997). Owing to these advantages, the Naïve Bayes classifier has gained great popularity in solving different classification problems. Nevertheless, its **independence assumption** among attributes is often violated in the real world. Fortunately, many approaches have been developed to alleviate this problem.

In general, these approaches can be divided into two groups. One attempts to relax the independence assumption of Naïve Bayes classifier, e.g. Semi-Naïve Bayes (SNB) (Kononenko, 1991), Searching for dependencies (Pazzani, 1995), the Tree Augmented Naïve Bayes (TAN) (Friedman, Geiger & Goldszmidt, 1997), SuperParent Tree Augmented Naïve Bayes (SP-TAN) (Keogh & Pazzani, 1999), Lazy Bayes Rule (LBR) (Zheng & Webb, 2000) and Aggregating One-Dependence Estimators (AODE) (Webb, Boughton & Wang, 2005).

The other group attempts to use certain **pre-processing procedures** to select or transform the attributes, which can be more suitable for the assumption of the Naïve Bayes classifier. The Feature selection can be implemented by greedy forward search (Langley & Sage, 1994) and **Decision Trees** (Ratanamahatana & Gunopulos, 2002). The transformation techniques include **Principal Component Analysis** (PCA) (Gupta, 2004), **Independent Component Analysis** (ICA) (Prasad, 2004) and CC-ICA (Bressan & Vitria, 2002). The next section describes the main ideas of the two groups of techniques in a broad way.

IMPROVING THE NAÏVE BAYES CLASSIFIER

This section introduces the two groups of approaches that have been used to improve the **Naïve Bayes classifier**. In the first group, the strong independence assumption is relaxed by restricted structure learning. The second group helps to select some major (and approximately independent) attributes from the original attributes or transform them into some new attributes, which can then be used by the Naïve Bayes classifier.

Relaxing the Independence Assumption

Relaxing the **independence assumption** means that the dependence will be considered in constructing the network. To consider the dependencies between attributes, Kononenko (Kononenko, 1991) proposed the Semi-Naïve Bayes classifier (SNB), which joined the attributes based on the theorem of Chebyshev. The medical diagnostic data were used to compare the performance of the SNB and the NB. It was found that the results of two domains are identical but in the other two domains SNB slightly improves the performance. Nevertheless, this method may cause overfitting problems. Another limitation of the SNB is that the number of parameters will grow exponentially with the increase of the number of attributes that need to be joined. In addition, the exhaustive searching technique of joining attributes may affect the computational time. Pazzani (Pazzani, 1995) used Forward Sequential Selection and Joining (FSSJ) and Backward Sequential Elimination and Joining (BSEJ) to search dependencies and join the attributes. They tested the two methods on UCI data and found that BSEJ provided the most improvement.

Friedman et al. (Friedman, Geiger & Goldszmidt, 1997) found that Kononenko's and Pazzani's methods can be represented as an augmented Naïve Bayes network, which includes some subgraphs. They restricted the network to be Tree Augmented Naïve Bayes (TAN) that spans over all attributes and can be learned by tree-structure learning algorithms. The results based on problems from the UCI repository showed that the TAN classifier outperforms the Naïve Bayes classifier. It is also competitive with C4.5 while maintains the computational simplicity. However, the use of the TAN classifier is only limited to the problems with discrete attributes. For the problems with continuous attributes, these attributes must be prediscretized. To address this problem, Friedman et al. (Friedman, Goldszmidt & Lee, 1998) extended TAN to deal with continuous attributes via parametric and semiparametric conditional probabilities. Keogh & Pazzani (Keogh & Pazzani, 1999) proposed a variant of the TAN classifier, i.e. SP-TAN, which could result in better performance than TAN. The performance of SP-TAN is also competitive with the Lazy Bayes Rule (LBR), in which the lazy learning techniques are used in the Naïve Byes classifier (Zheng, & Webb, 2000; Wang & Webb, 2002)

Although LBR and SP-TAN have outstanding performance on the testing data, the main disadvantage of the two methods is that they have high computational complexity. Aggregating One-Dependence Estimators (AODE), developed by Webb et al. (Webb, Boughton & Wang, 2005), can avoid model selection which may reduce computational complexity and lead to lower variance. These advantages have been demonstrated by some empirical experiment results. It is also empirically found that the average prediction accuracy of AODE is comparative to that of LBR and SP-TAN but with lower variance. Therefore, AODE might be more suitable for small datasets due to its lower variance.

Using Pre-Processing Procedures

In general, the **pre-processing procedures** for the Naïve Bayes classifier include feature selection and transforming the original attributes. The Selective Bayes classifier (SBC) (Langley & Sage, 1994) deals with correlated features by selecting only some attributes into the final classifier. They used a greedy method to search the space and forward selection to select the attributes. In their study, six UCI datasets are used to compare the performance of the Naïve Bayes classifier, SBC and C4.5. It is found that selecting the attributes can improve the performance of the Naïve Bayes classifier when there are redundant attributes. In addition, SBC is found to be competitive with C4.5 in terms of the datasets by which C4.5 outperforms the Naïve Bayes classifier. The study by Ratanamahatana & Gunopulos (Ratanamahatana & Gunopulos, 2002) applied C4.5 to select the attributes for the Naïve Bayes classifier. Interestingly, experimental results showed that the new attributes obtained by C4.5 can make the Naïve Bayes classifier outperform C4.5 with respect to a number of datasets.

Transforming the attributes is another useful pre-processing procedure for the Naïve Bayes classifier. Gupta (Gupta, 2004) found that **Principal Component Analysis** (PCA) was helpful to improve the classification accuracy and reduce the computational complexity. Prasad (Prasad, 2004) applied **Independent Component Analysis** (ICA) to all the training data and found