Neurocomputing 149 (2015) 1647–1657

Contents lists available at ScienceDirect

Neurocomputing

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

A new fast reduction technique based on binary nearest neighbor tree

Juan Li [a,b], Yuping Wang [a,*]

[a] School of Computer Science and Technology, Xidian University, Xi'an 710071, China
[b] School of Distance Education, Shaanxi Normal University, Xi'an, China

ARTICLE INFO

Article history:
Received 18 June 2013
Received in revised form
26 July 2014
Accepted 15 August 2014
Communicated by Hung-Yuan Chung
Available online 27 August 2014

Keywords:
Supervised classification
Binary nearest neighbor tree
K-nearest neighbor rule
Prototype selection
Prototype generation

ABSTRACT

The K-nearest neighbor (*KNN*) rule is one of the most useful supervised classification methods, and is widely used in many pattern classification applications due to its simplicity. However, it faces prohibitive computational and storage requirements when dealing with large datasets. A reasonable way of alleviating this problem is to extract a small representative subset from the original dataset without reducing the classification accuracy. This means the most internal patterns are removed and the boundary patterns that can contribute to better classification accuracy are retained. To achieve this purpose, a new algorithm based on binary tree technique and some reduction operations is presented. The key issues of the proposed algorithm are how to build binary nearest neighbor search tree and design reduction strategies to keep the high classification accuracy patterns. In particular, firstly, we utilize several tree control rules and *KNN* rule to build a binary nearest neighbor tree of each random pattern. Secondly, according to the node locations in each binary nearest neighbor tree and the strategies of selection and replacement, different kinds of patterns as prototypes are obtained, which are close to class boundary regions or locate in the interior regions, and some internal patterns are generated. Finally, experimental results show that the proposed algorithm effectively reduces the number of prototypes while maintaining the same level of classification accuracy as the traditional *KNN* algorithm and other prototype algorithms. Moreover, it is a simple and fast hybrid algorithm for prototype reduction.

1. Introduction

Classification has been a very well-studied research topic and many research works on it have appeared since the 1960s. One of the most widely studied non-parametric classification approaches is the K-Nearest Neighbor (*KNN*) decision rule [1]. Due to its simplicity and effectiveness, it is widely used in a large number of pattern classification problems, such as intrusion detection, text classification, and information retrieval, etc [1–5]. However, since *KNN* classifier could not be established in advance and plays an important rule when a new unlabeled pattern comes, it may cause a prohibitive time and memory burden especially when the scale of dataset is large and its dimension is high. One possible approach for this problem is the nearest prototype algorithm which is developed by reducing the scale of the training set and meanwhile guaranteeing little loss of classification accuracy simultaneously. The other possible approach for this problem is the dimension reduction algorithm which is also developed by reducing the memory burden and reducing the classification time of *KNN* classifier.

The nearest prototype algorithms can realize above requirement and have been studied for many years. There are two main research branches of them, i.e. prototype selection and prototype generation, and each aims at obtaining an acceptable scale of the prototype set as well as getting higher or no worse classification performance than one obtained from the original training set. There are two kinds of the traditional nearest prototype algorithms: condensation and edition strategy.

The first and most popular algorithm is the Condensing Nearest Neighbor (*CNN*) that uses a kind of prototype selection techniques and was first to propose by Hart [6] in 1968 for reducing the dataset size by using the nearest neighbor decision. Since *CNN* is very sensitive to noise patterns and the scan sequence of dataset, thereafter, a series of improved versions were proposed based on *CNN*. In 1976, Tomek [7] defined Tomek Link and used a generic algorithm useful for noisy datasets by potentially removing the noise patterns. But there are some drawbacks of Tomek's method, such as the very large time consumption, the higher sensitivity of scan sequence and the incorrect deletion of the potentially useful patterns, and so on. In 2006, Fu Chang et al. [8] proposed an adaptive prototype learning algorithm *GCNN* which employs the same *CNN* scheme to determine the number and location of prototypes, but differs to *CNN* with new replacement patterns as prototypes, and a new assumption of distance measures. In 2007,

* Corresponding author.
E-mail address: ywang@xidian.edu.cn (Y. Wang).

http://dx.doi.org/10.1016/j.neucom.2014.08.028
0925-2312/

Angiulli [9] further proposed what he called Fast Nearest Neighbor Condensation (*FCNN*) which can cope with large data sets classification. However, it is well known that *CNN* and some *CNN−based* algorithms suffer from several drawbacks [8,9], and the three notable ones, which causes a great impact on the successful application of the algorithm, are random initial prototype selection, search sequence and consistent subset of the training set.

Edition [10] is another popular strategy in pattern reduction, and can also be carried out through some sampling schemes of dataset. The main goal of these schemes is to remove or modify patterns located at class boundaries. To this end, they act over patterns that are noise or do not agree with their nearest neighbors, thus obtain smoother class decision boundaries. However, these algorithms based on the edition strategy do not remove internal points that do not necessarily contribute to the decision boundaries. The classification effect obtained is related to the improvement of generalization accuracy in test dataset at the cost of the lower reduction ratio. So some improved versions have been occurred.

As well known, patterns nearby class boundaries have higher contribution for pattern classification, while most of patterns that contain the vast number of inner patterns and all of outlier patterns have little contribution for pattern classification. Moreover, for many classification algorithms, the computation and storage requirements are very high when dealing with the large datasets. To solve this problem, an efficient algorithm, which utilizes some partitioning strategies to divide the whole space into several smaller spaces, will be essential. So how to obtain the boundary area and design the partitioning strategies have been studied in some literatures [11–14]. Cluster-based learning (*CBL*) algorithms are proposed in [12–14], in which prototypes are not only patterns per se, but also the weighted averages of patterns. In particular, following the same scheme, the Generalized-Modified Chang algorithm is proposed in [12], which merges the same-class nearest clusters and selects the centers from the new merged clusters as prototypes. In [13], after splitting the training set into c clusters, the selected prototypes are the centers of the c clusters of *CBL*. Based on the clustering idea, a new improved clustering algorithm that names as Prototype Selection by Clustering (*PSC*) [14] selects border prototypes and some interior prototypes. In recently, the k-means clustering algorithm, the fuzzy c-means algorithm, and some hierarchical clustering algorithms are widely used in some prototype reduction application. Divide-and-conquer approach has attracted many attentions in saving the runtime consumption. In 2005, Raicharoen [15] proposed an algorithm which can realize the prototypes' correct selection insensitive to the scan sequence of patterns by building some separating hyperplanes located among the *POC−NN* essential prototypes. In 2009, Haro-García and García-Pedrajas [16] divided the original training set into some smaller subsets where the instance selection algorithm is applied. Then the divide-and-conquer operator is repeated in those smaller subsets that include the selected instances and the previous subsets. In 2009, Fayed and Atiya [17] proposed a prototype reduction algorithm, namely, the template reduction for *KNN*(*TRKNN*). The basic idea of *TRKNN* is to define a "chain" of the nearest neighbors and set a cutoff value for the distances among alternating adjacent nodes of the chain to select the "condensed set".

In addition, some new techniques have been emerged for solving prototype reduction problem. For instance, Kim [18] obtained the prototype set using a neural network learning algorithm. Devi and Murty [19] introduced some evolutionary methods to solve the optimizing problem of prototype reduction. Evolutionary algorithms, such as genetic algorithm, Michigan Approach, *PSO*, etc, have been well applied to prototype reduction problems [20,21]. Hybrid model [22,23] is formed by combining the condensing and editing techniques or other techniques, and is widely used in the field of prototype reduction problems.

The dimension reduction algorithms, which can reduce the dimension of the original dataset and speed up the classification running, have been studied for many years. They can be divided into filter, wrapper and embedded approaches [24]. For the filter based methods, the process of selecting features is not guided by the performance of a classification method. On the contrary, it is guided by the accuracy of a classification algorithm to wrapper based methods. Although a large number of strategies have been proposed to improve these two traditional approaches, the embedded approach has been developed rapidly. For instance, merged with Neural Network, Cheung and Huang [25] adopted divide-and-conquer strategy to divide the conventional *RBF* network into several sub-networks and it can deal with a high-dimensional modeling problem via several low-dimensional ones. Furthermore, Cheung and Law [26] proposed a novel rival-model penalized self-organizing map learning algorithm that can adaptively chooses several rivals of the best-matching unit and penalizes their associated models. Wu and Li [27] merged a variable weighting method into the self-organizing-feature-maps algorithm to find the winning feature neuron. In addition, many other technologies are also adopted in the dimension reduction algorithms, such as clustering [28] and cooperative game theory [29], etc.

In this paper, our work is focused on using a new algorithm for obtaining a prototype set that consists of some generated and selected prototypes. Unlike all of these above mentioned algorithms, our proposed algorithm, named as the binary nearest neighbor tree algorithm (*BNNT*), obtains the prototype set through selecting and generating prototypes from the binary nearest neighbor trees. In our algorithm, a "binary nearest neighbor tree" is first defined, and then two strategies of the selection and generation are proposed. By building a binary nearest neighbor tree, the relationships among the tree nodes can be effectively obtained and used as follows: when the tree locates in a class interior, we generate the centroid pattern to replace these tree nodes; when the tree locates in the boundary of several classes, we select those patterns that have different class labels and are directly connected in the binary tree. In order to show the performance of our algorithm, we compare it with several prototype algorithms using UCI benchmark datasets [30] based on theory and experiment analysis.

To be specific, the main contributions in this paper are as follows:

(1) The fast distinguishing rule to the pattern location is introduced. Using the distinguishing rule, we can quickly distinguish the location of patterns whether they locate at nearby class boundaries or internal class regions;
(2) A new criterion for prototype reduction is proposed, which can select the prototypes nearby class boundaries and generate a few internal prototypes based on binary nearest neighbor tree. To each binary nearest neighbor tree, we adopt some different control strategies of the tree building, the growth control, prototype selection and prototype generation, etc;
(3) The proposed algorithm is insensitive to the growth control parameter. Although the parameter is predefined, the proposed algorithm can generate some new internal patterns when trees are homogeneous and do not reach the growth control threshold. The proposed algorithm can reduce the sensitivity of the parameter due to the generation mechanism of new internal pattern;
(4) The proposed algorithm can reduce the influence of noise and can be used as a preprocessor of some hybrid algorithms.