Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 111 (2017) 252–259

Procedia Computer Science

www.elsevier.com/locate/procedia

8th International Conference on Advances in Information Technology, IAIT2016, 19-22 December 2016, Macau, China

RMHC-MR: Instance selection by random mutation hill climbing algorithm with MapReduce in big data

Lu Si*, Jie Yu, Wuyang Wu, Jun Ma, Qingbo Wu, Shasha Li

College of Computer, National University of Defense Technology, Changsha 410073, China

Abstract

Instance selection is used to reduce the size of training set by removing redundant, erroneous and noisy instances and is an important pre-processing step in KDD (knowledge discovery in databases). Recently, to process very large data set, several methods divide the training set into disjoint subsets and apply instance selection algorithms to each subset independently. In this paper, we analyze the limitation of these methods and give our viewpoint about how to "divide and conquer" in instance selection procedure. Furthermore, we propose an instance selection method based on random mutation hill climbing (RMHC) algorithm with MapReduce framework, called RMHC-MR. The experimental result shows that RMHC-MR has a good performance in terms of classification accuracy and reduction rate.

Peer-review under responsibility of the organizing committee of the 8th International Conference on Advances in Information Technology

Keywords: Instance selection; MapReduce; big data; nearest neighbor; classification

1. Introduction

Data have quality if they satisfy the requirements of the intended use[1]. In machine learning algorithms, a training set is a collection of training examples called instances. In practice, training set of raw data may collect a lot of redundant, erroneous, and noisy instances that can result in large memory occupation, slow execution speed, and

* Corresponding author. Tel.:+86-15116421601
*E-mail address:*lusi@ubuntukylin.com

1877-0509
Peer-review under responsibility of the organizing committee of the 8th International Conference on Advances in Information Technology
10.1016/j.procs.2017.06.061

over-sensitivity to noise. Especially for some instance-based methods, such as k-NN[2] and DDC[3], the effectiveness are related to the size of the training set.

Instance selection techniques aim to reduce the size of training sets by removing instances that are noisy, redundant or both, and so likely to degrade the mining performance. A successful algorithm can significantly reduce the size of training set without a significant reduction of generalization accuracy[4]. Instance selection is usually divided into three approaches: *selection*, *abstraction* and *hybrid*. *Selection* algorithm maintains a subset of the original instances and *abstraction* modifies the instances using a new representation. The *hybrid* algorithm is based on the combination of *selection* and *abstraction* methods.

In this paper, we use T as original instances in the training set and S represents the subset of T. That is to say, instance selection is searching for a subset S of instances to keep from training set T.

The condensed nearest neighbor (CNN)[5] is the first and probably the simplest instance selection strategy. It begins with an empty subset S, and adds each instance from training set T to S if the instance is misclassified using only the instances in S. However, noisy instance will usually be misclassified by their neighbors, and thus will be retained in S. Generalized condensed nearest neighbor (GCNN)[6] is a recent extension of CNN. GCNN assigns an instance to S if it satisfies an absorption criterion according to a threshold. It can reduce the size of S compared to CNN. Reduced nearest neighbor (RNN)[7] starts with S = T and removes each instance from S if the removal does not cause other instances in T to be misclassified by the remaining instances in S. It is able to remove noisy instances, but is more expensive in terms of learning time compared to CNN. Edited nearest neighbor rule (ENN)[8] proposed by Wilson also starts with S = T, and then removes any instance that would be misclassified by its k nearest neighbors (with k = 3, typically). This algorithm removes noisy and close border instances. Repeated ENN (RENN) applies the ENN algorithm repeatedly until all instances in S have a majority of their k nearest neighbors with the same class. Five decremental reduction optimization procedure algorithms (DROP1 – DROP5)[9] were proposed by Wilson and Martinez. Among them, DROP3 is the most successful method. In DROP3, each instance has k nearest neighbors called associates. The algorithm removes it if at least as many of its associates in T would be classified correctly without it (where S = T originally). Furthermore, DROP3 uses ENN to remove noisy instances.

Recently, some new instance selection methods were proposed by different researchers. Based on outlier pattern analysis and prediction. Lin et al.[10] proposed an approach to detect the representation instances from large data sets. IRAHC[4] maintains a hyperrectangle and removes interior instances and keeps border and near border ones. Ref.[11] utilizes the fuzzy-rough instance selection method based on weak gamma evaluator to remove redundant and erroneous instances. Ref.[12] proposed a fast instance selection method for large data sets by clustering. This algorithm selects border instances and some non-border instances. Evolutionary algorithm (EA)[13] is one type of instance selection. In this method, some initial sets of instances are represented by chromosome strings. According to a fitness function, the individuals are evaluated and the best chromosomes are selected after a specific number of iterations.

However, with the exponential growth of data in many application domains such as industry, medicine and financial businesses, processing very large scale data sets for instance selection is becoming a major limitation. The traditional methods lack enough scalability to cope with data sets of millions of instances even though they have already been performed over the previous smaller data set. Recent improvements in this field cover the stratification of data and redesign the algorithms and their inclusion in parallel environments. Based on the divide and conquer principle, Ref.[14] divides the original training set into small subsets where the instance selection algorithms are applied and rejoined in a new training set. Ref.[15] proposed a MapReduce-based framework for nearest neighbor classifier to deal with the classification problems of large data sets. MRIVS[16] is another MapReduce-based distributed instance selection algorithm. It partitions the large data sets into some small subsets and selects informative instances in Map phase with instance selection algorithms. In Reduce phase, it collects the selected instances from different nodes and obtains a subset. These processes are repeated p times (p is a parameter defined by the user).

All these distributed algorithms[14, 15, 16] just simply divide the training set into small subsets and apply traditional algorithms independently. In this paper, we analyze the limitation of these method, redesign the random mutation hill climbing (RMHC)[17] algorithm and implements it with MapReduce[18] framework.

The remainder of the paper is organized as follows: In Section 2, we briefly introduce the MapReduce paradigm and RMHC algorithm. In Section 3, we give out our viewpoint about how to divide and conquer for instance