Knowledge-Based Systems 39 (2013) 240–247

Contents lists available at SciVerse ScienceDirect

Knowledge-Based Systems

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

Genetic algorithms in feature and instance selection

Chih-Fong Tsai [a,*], William Eberle [b], Chi-Yuan Chu [a]

[a] Department of Information Management, National Central University, Taiwan
[b] Department of Computer Science, Tennessee Technological University, USA

ARTICLE INFO

Article history:
Received 9 May 2012
Received in revised form 12 November 2012
Accepted 18 November 2012
Available online 28 November 2012

Keywords:
Genetic algorithms
Feature selection
Instance selection
Data mining
Data preprocessing

ABSTRACT

Feature selection and instance selection are two important data preprocessing steps in data mining, where the former is aimed at removing some irrelevant and/or redundant features from a given dataset and the latter at discarding the faulty data. Genetic algorithms have been widely used for these tasks in related studies. However, these two data preprocessing tasks are generally considered separately in literature. It is unknown what the performance differences would be when feature and instance selection and feature or instance selection are performed individually. Therefore, the aim of this study is to perform feature selection and instance selection based on genetic algorithms using different priorities to examine the classification performances over different domain datasets. The experimental results obtained from four small and large scale datasets containing various numbers of features and data samples show that performing both feature and instance selection usually make the classifiers (i.e., support vector machines and *k*-nearest neighbor) perform slightly poorer than feature selection or instance selection individually. However, while there is not a significant difference in classification accuracy between these different data preprocessing methods, the combination of feature and instance selection largely reduces the computational effort of training the classifiers, as opposed to performing feature and instance selection individually. Considering both classification effectiveness and efficiency, we demonstrate that performing feature selection first and instance selection second is the optimal solution for data preprocessing in data mining. Both SVM and *k*-NN classifiers provide similar classification accuracy to the baselines (i.e., those without data preprocessing). The decisions regarding which data preprocessing task to perform for different dataset scales are also discussed.

1. Introduction

The process of knowledge discovery in databases (KDD), or data mining, generally involves a number of steps, such as dataset selection, data preprocessing, data analysis, and result interpretation and evaluation [5,16]. Data preprocessing is one of the most important steps with the aim of making the chosen dataset as 'clean' as possible for eventual analysis and evaluation. In other words, quality mining results cannot be obtained if the data quality is low [27,8].

Feature selection (or dimensionality reduction) and *instance selection* (or record reduction) are two of the more active preprocessing problems in data mining. This is because the number of features and data samples selected is usually very large in most real-world data mining problems.

If too many instances are considered, it can result in large memory requirements, high disk access, slow execution speed, and a possible over-sensitivity to noise [55]. In addition, it is often the fact that data are not all equally informative and some data points will be further away from the sample mean than what is deemed reasonable. Similarly record reduction is aimed at discarding faulty data (or outliers), which could be considered as noisy points lying outside a set of defined clusters and could lead to significant performance degradation [1,4]. Data mining tasks such as classification or prediction performance that is carried out without considering the instance selection step will very likely lead to poorer results [47,56].

On the other hand, if too many features are used for data analysis, it can cause the curse of dimensionality problems [36]. Since not all of the pre-chosen features are informative, the objective of feature selection is to select more representative features which have more discriminative power over a given dataset. This is also called dimensionality reduction [26].

In the literature, many related studies have shown promising results for feature selection and instance selection approaches [25,35,42,50,53]. However, up until now, the focus has been on either selecting more representative features or reducing faulty data, as it relates to effective classification or prediction. This leads to the important research question about which step (i.e., feature

* Corresponding author. Tel.: +886 3 422 7151; fax: +886 3 4254604.
E-mail address: cftsai@mgt.ncu.edu.tw (C.-F. Tsai).

0950-7051/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.knosys.2012.11.005

selection or instance selection) should be performed first when both steps are critical to improving the mining performance. For many relevant and large scale datasets, both data preprocessing steps need to be performed. This is because in many domain problems there is usually no exact agreed upon number of variables, and all of those collected for a specific domain may not be informative. Furthermore, some data samples in a given large dataset may be regarded as noisy. Therefore, feature selection and instance selection should both be considered in order to develop a more effective model [17,11].

Genetic algorithms (GAs) comprise one of the most widely used techniques for feature and instance selection, and can improve the performance of data mining algorithms [12,14,39,37,46,51,52]. In particular, Cano et al. [7] have shown that better results can be obtained with GAs than with many traditional and non-evolutionary instance selection methods in terms of better instance selection rates and higher classification accuracy. Moreover, GAs have been shown to be suitable for large-scale feature selection problems [33].

However, very few consider feature selection and instance selection *together* using a GA over a given dataset. For example, given a dataset D containing m dimensional features and i data samples, using feature selection and instance selection as the first and second preprocessing steps respectively, will lead to D_1 containing n dimensional features and j data samples (where $0 < n < m$ and $0 < j < i$). On the other hand, if the operations are performed in reverse order, different results can be obtained.

The aim of this study is to perform feature selection and instance selection based on genetic algorithms using different priorities and to examine the classification performances over different domain datasets. In addition, the results will be compared, where a dataset is created without considering both data preprocessing steps, by feature selection only, and a dataset by instance selection only.

The rest of this paper is organized as follows. Section 2 describes the concept of feature selection and instance selection. In addition, genetic algorithms are overviewed in terms of data preprocessing. Section 3 presents the research design and experimental results. Finally, some conclusions are offered in Section 4.

2. Literature review

2.1. Feature selection

The number of features (or variables) collected in a dataset is usually relatively large (i.e., the curse of dimensionality) and not all of these features are informative or can provide high discriminative power [43]. The aim of feature selection is to remove the irrelevant and/or redundant features from the chosen dataset, thereby improving the performance of the classification and/or clustering algorithms. In addition, for a specific a dataset, feature selection can help analysts understand which features are important as well as how they are related.

Feature selection can be defined as the process of choosing a minimum subset of m features from the original dataset of n features ($m < n$), so that the feature space (i.e. the dimensionality) is optimally reduced according to the following evaluation criteria [10]:

- the classification accuracy does not significantly decrease; and
- the resulting class distribution, given only the values for the selected features, is as close as possible to the original class distribution, given all features.

A feature selection algorithm usually consists of four steps: subset generation, subset evaluation, stopping criterion, and result validation [10]. Subset generation is a search procedure which generates subsets of features for evaluation. Each subset generated is evaluated by some specific evaluation criterion and compared with the previous best one with respect to this criterion. If a new subset is found to be better, then the previous best subset is replaced by the new subset.

The interested reader can refer to Kudo and Sklansky [33] and Guyon and Elisseeff [26] for more information.

2.2. Instance selection

Wilson and Martinez [55] found that one problem with using the original data points is that there may not be any located at the precise points that would make for the most accurate and concise concept description. Therefore, the aim of instance selection, or record reduction, is to reduce the size of a dataset while still maintaining the integrity of the original dataset. In some cases, generalization accuracy can increase when noisy instances are removed and when decision boundaries are smoothed to more closely match the true underlying function.

These instances can also be regarded as outliers (or bad data). Specifically, outliers are those data points which are highly unlikely to occur given a model of the data. One approach to performing this task is to calculate the distances to neighboring data points by implementing a clustering algorithm [21].

Instance selection can be defined as follows. Let X_i be an instance where $X_i = (X_{i1}, X_{i2}, \ldots, X_{im}, X_{ic})$ meaning that X_i is represented by m-dimensional features and X_i belongs to class c given by X_{ic}. Then, assume that there is a training set TR which consists of M instances and a testing set TS composed of N instances. If $S \subseteq TR$ is the subset of selected samples that are produced by some instance selection algorithm, then we can classify a new pattern T from TS over the instances of S.

The interested reader can refer to Reinartz [47], Liu and Motoda [40], Jankowski and Grochowski [30] and Grochowski and Jankowski [24] for more information.

2.3. Genetic algorithms

The main idea behind the evolutionary algorithms (EAs) is derived from Darwin's theory of evolution arising from natural selection, of which genetic algorithms (GA) are one example. The basic idea of a GA is that you have a population of strings (called chromosomes), which encode candidate solutions (called individuals) to an optimization problem. In general, the genetic information (i.e., chromosome) is represented by a bit string (such as binary strings of 0s and 1s), and sets of bits encode the solution. Genetic operators are then applied to the individuals of the population for the next generation (i.e., a new population of individuals). There are two main genetic operators: crossover and mutation. Crossover creates two offspring strings from two parent strings by copying selected bits from each parent, whereas mutation randomly changes the value of a single bit (with small probability). In addition, a fitness function is used to measure the quality of an individual in order to increase the probability that the single bit can survive throughout the evolutionary process. Moreover, a GA can deal with large search spaces efficiently, and hence has less chance to arrive at a local optimal solution than other algorithms [22,15].

GAs have been tested on a number of domains for solving the feature and instance selection problems individually, such as Aydogan et al. [3], Das et al. [9], Pedrycz and Syed Ahmad [41], and Ratta et al. [45] for feature selection and Garcia et al. [18,19], Garcia-Pedrajas and Perez-Rodriguez [20], and Triguero et al. [49] for instance selection.