Applied Soft Computing 47 (2016) 179–190

Contents lists available at ScienceDirect

Applied Soft Computing

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

Evolving meta-ensemble of classifiers for handling incomplete and unbalanced datasets in the cyber security domain

G. Folino*, F.S. Pisani

Institute of High Performance Computing and Networking (ICAR-CNR), Via P. Bucci, 87036 Rende (CS), Italy

ARTICLE INFO

Article history:
Received 15 July 2015
Received in revised form 26 April 2016
Accepted 30 May 2016
Available online 8 June 2016

Keywords:
Ensemble
Data mining
Cyber security
Missing features

ABSTRACT

Cyber security classification algorithms usually operate with datasets presenting many missing features and strongly unbalanced classes. In order to cope with these issues, we designed a distributed genetic programming (GP) framework, named CAGE-MetaCombiner, which adopts a meta-ensemble model to operate efficiently with missing data. Each ensemble evolves a function for combining the classifiers, which does not need of any extra phase of training on the original data. Therefore, in the case of changes in the data, the function can be recomputed in an incremental way, with a moderate computational effort; this aspect together with the advantages of running on parallel/distributed architectures makes the algorithm suitable to operate with the real time constraints typical of a cyber security problem. In addition, an important cyber security problem that concerns the classification of the users or the employers of an e-payment system is illustrated, in order to show the relevance of the case in which entire sources of data or groups of features are missing. Finally, the capacity of approach in handling groups of missing features and unbalanced datasets is validated on many artificial datasets and on two real datasets and it is compared with some similar approaches.

1. Introduction

In the last few years, as a consequence of our interconnected society, the interest in cyber security problems has really been increasing and cyber crime seriously threatens national governments and the economy of many industries [1]. Indeed, computer and network technologies have intrinsic security vulnerabilities, i.e., protocol, operating system weaknesses, etc. Therefore, potential threats and the related vulnerabilities need to be identified and addressed to minimize the risks. In addition, computer network activities, human actions, etc. generate large amounts of data and this aspect must be seriously taken into account.

Data mining techniques could be used to fight efficiently, to alleviate the effect or to prevent the action of cybercriminals, especially in the presence of large datasets. In particular, classification can be used efficiently for many cyber security applications, i.e., classification of the user behavior, risk and attack analysis, intrusion detection systems, etc. However, in this particular domain, datasets often have different number of features and each attribute could have different importance and cost. Furthermore, the entire system must also work if some features are missing and/or the classes are unbalanced. Therefore, a single classification algorithm performing well for all the datasets would be really unlikely, especially in the presence of changes and with constraints of real time and scalability.

In the ensemble learning paradigm [2,3], multiple classification models are trained by a predictive algorithm, and then their predictions are combined to classify new tuples. This paradigm presents a number of advantages with regard to using a single model, i.e., it reduces the variance of the error, the bias, and the dependence on a single dataset and works well in the case of unbalanced classes; furthermore, the ensemble can be build in an incremental way and can be easily implemented on a distributed environment. If we consider a stream of data, the ensemble needs to be re-trained to take into account changes in the data. This process could be computationally expensive, especially if it is necessary to retrain the models or to regenerate new models on the new data.

Therefore, in order to classify large datasets in the field of cyber security, usually having the above-cited issues of unbalanced classes and missing features, a new framework, named CAGE-MetaCombiner, is proposed. The framework extends a well-known implementation of distributed GP (CellulAr GEnetic programming (CAGE) environment) and adopts a meta-ensemble model in order to cope with missing data, while the GP system, which evolves the

* Corresponding author at: ICAR-CNR Istituto di Calcolo e Reti ad Alte Prestazioni, Via P. Bucci 41/C, c/o DIMES, University of Calabria, 87036 Rende, Italy.
Tel.: +39 0984831731; fax: +39 0984839054.
E-mail address: folino@icar.cnr.it (G. Folino).

http://dx.doi.org/10.1016/j.asoc.2016.05.044
1568-4946/

combiner function of the ensemble, permits to handle unbalanced classes thanks to a weighted fitness function. In practice, an ensemble is built for each group of likely missing features, as explained in the following, and the different ensembles perform a weighted vote in order to decide the correct class. Each ensemble evolves a function for combining the classifiers, which can be trained only on a portion of the training set and does not need any extra phase of training on the original data. In fact, in the case of changes in the data, the function can be recomputed in an incremental way, with a moderate computational effort. In addition, all the phases of the algorithm are distributed and can exploit the advantages of running on parallel/distributed architectures to cope with real time constraints.

The rest of the paper is structured as follows: in Section 2 presents some related works; in Section 3, a real scenario in the field of cyber security is illustrated; Section 4 is devoted to some background information concerning the problem of missing data and incomplete datasets and the ensemble of classifiers; in Section 5, the framework and its software architecture is illustrated; Section 6 shows a number of experiments conducted to verify the effectiveness of the approach and to compare it with other similar approaches; finally, Section 7 concludes the work.

2. Related works

Evolutionary algorithms have been used mainly to evolve and select the base classifiers composing the ensemble [5,6] or adopting some time-expensive algorithms to combine the ensemble [7]; however, a limited number of papers concerns the evolution of the combining function of the ensemble by using GP.

In the following, we analyze two groups of approaches. The first group comprises GP-based ensembles used to evolve the combination function. Most of the analyzed approaches employ a high number of resources to generate the function and therefore, their usage is not particularly recommended for large real datasets. The approaches of the second group adopt the ensemble paradigm to cope with incomplete and/or unbalanced data, but differently from our work, require to use the training set also in the phase of generating the combiner function, with a considerable overhead in this phase. The only exception is the last analyzed paper, which does not use the training set in this phase; however, it builds a number of random subsets of the features of the original dataset and therefore, its application is problematic when the number of features is high.

Chawla et al. [8] propose an evolutionary algorithm to combine the ensemble, based on a weighted linear combination of classifiers predictions, using many well-known data mining algorithm as base classifiers, i.e., J48, NBTree, JRip, etc. In [9], the authors extend their work in order to cope with unbalanced datasets. In practice, they increase the total number of base classifiers and adopt an oversampling technique. In [10], the authors consider also the case of a homogenous ensemble and show the effect of a cut-off level on the total number of classifiers used in the generated model. In [11], the authors develop a GP-based framework to evolve the fusion function of the ensemble both for heterogeneous and homogeneous ensemble. The approach is compared with other ensemble-based algorithms and the generalization properties of the approach are analyzed together with the frequency and the type of the classifiers presents in the solutions. These works can also operate on incomplete datasets, but differently from our approach, use an oversampling technique. In addition, they do not take into account the problems concerning the unbalanced datasets, while our technique permits to efficiently handle them by considering different weights derived from the performance of the classifiers on the training sets.

In [12], Brameier and Banzhaf use linear genetic programming to evolve teams of ensembles. A team consists of a predefined number of heterogeneous classifiers. The aim of a genetic algorithm is to find the best team, i.e., the team obtaining the highest accuracy on the given datasets. The prediction of the team is the combination of individual predictions and it is based on the average or the majority voting strategy, also considering predefined weights. The errors of the individual members of the team are incorporated into the fitness function, so that the evolution process can find the team with the best combination of classifiers. Differently from our approach, the recombination of the team members is not completely free, but only a maximum pre-defined percentage of the models can be changed. In our approach, GP generates tree-based models and the number of base classifiers in the tree is not predefined; therefore, the evolution process can freely select the best combination of the base classifiers.

Chen et al. [14] use multiple ensembles to classify incomplete datasets. Their strategy consists in partitioning the incomplete datasets in multiple complete sets and in training the different classifiers on each sample. Then, the predictions of all the classifiers could be combined according to the ratio between the number of features in this subsample and the total features of the original dataset. This approach is orthogonal to our and therefore, it could be included in our system.

Another approach to cope with incomplete datasets can be found in [13]. The authors build all the possible LCP (Local Complete Pattern), i.e., a partition of the original datasets into complete datasets, without any missing features; a different classifier is built on each LCP, and then they are combined to predict the class label, basing on a voting matrix. The experiments compared the proposed approach with two techniques to cope with missing data, i.e., deletion and imputation, on small datasets and show how the approach outperforms the other two techniques. However, the phase of building the LCP could be really expensive.

Learn++.MF [15] is an ensemble-based algorithm with base classifiers trained on a random subset of the features of the original dataset. The approach generates a large number of classifiers, each trained on a different feature subset. In practice, the instances with missing attributes are classified by the models generated on the subsets of the remaining features. Then, the algorithm uses a majority voting strategy in order to assign the correct class under the condition that at least one classifier must cover the instance. When the number of attributes is high, it is unfeasible to build classifiers with all the possible sets of features; therefore, the subset of the features is iteratively updated to favor the selection of those features that were previously undersampled. However, this limits the real applicability of the approach to datasets with a low number of attributes.

3. A real scenario: classification of user profiles in e-payment systems

The inspiration of the approach taken in this paper comes from a project on cyber security for e-payment systems, in which one of the main tasks consists in dividing the users of an e-payments systems into homogenous groups on the basis of their weakness or vulnerabilities from the cyber security point of view. In this way, the provider of an e-payment system can conduct a different information and prevention campaign for each class of users, with obvious advantages in terms of time and cost savings.

This technique is usually named segmentation, i.e., to the process of classifying customers into homogenous groups (segments), so that each group of customers shares enough characteristics in common to make it viable for a company to design specific offerings or products for it. It is based on a preliminary investigation in