Knowledge-Based Systems 75 (2015) 141–151

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

# Multiple Boosting in the Ant Colony Decision Forest meta-classifier

Jan Kozak *, Urszula Boryczka

*Institute of Computer Science, University of Silesia, Będzińska 39, 41–200 Sosnowiec, Poland*

ARTICLE INFO

*Article history:*
Received 5 June 2014
Received in revised form 22 November 2014
Accepted 23 November 2014
Available online 29 November 2014

*Keywords:*
Meta-ensemble
Boosting
Bagging
Random Forest
Ant Colony Optimization
Ant Colony Decision Forest

ABSTRACT

The idea of ensemble methodology is to combine multiple predictive models in order to achieve a better prediction performance. In this task we analyze the self-adaptive methods for improving the performance of Ant Colony Decision Tree and Forest algorithms. Our goal is to present and compare new meta-ensemble approaches based on Ant Colony Optimization. The proposed meta-classifiers (consisting of homogeneous classifiers) can be characterized by the self-adaptability or the good accommodation with the analyzed data sets and offer appropriate classification accuracy.

In this article we provide an overview of ensemble methods in classification tasks and concentrate on the different methodologies, such as Bagging, Boosting and Random Forest. We present all important types of ensemble methods including Boosting and Bagging in context of distributed approach, where agent-ants create better solutions employing adaptive mechanisms. Self adaptive, combining methods and modeling appropriate issues, such as ensembles presented here are discussed in context of the quality of the results. Smaller trees in decision forest without loss of accuracy are achieved during the analysis of different data sets.



## 1. Introduction

Many recent approaches in machine learning have benefited from the idea that the predictions of a committee or ensemble of models will usually be better than the predictions of one single model. The errors of one model can be counteracted by the hits (in context of incorrect predictions) of the other models. This collaborative view on learning can take place without interaction between the learning agents, which is known as ensemble learning, or with interaction during the learning stage, known as co-learning.

It is intuitively clear that an ensemble of identical classifiers will be no better than a single member. Classifier ensembles that enforce diversity fare better than ones that do not. The classical example is Boosting versus Bagging, currently two of the most successful ensemble strategies. Both approaches build ensembles by training each classifier on a bespoke data set. Boosting [1] promotes diversity actively, whereas Bagging [2] relies on independent re-sampling from the training set. Boosting has been crowned as the "best off-the-shelf classifier" by Leo Breiman himself, the creator of Bagging [2].

Bootstrap aggregating (Bagging) and Boosting are useful techniques that can be used to improve the predictive performance of tree models. Boosting may also be useful in connection with many other models, e.g. for additive models with high-dimensional predictors; whereas Bagging is most prominent for improving tree algorithms. Boosting and Bagging are two approaches used to combine "weak" models in order to build prediction models that are significantly better. The general theoretical and practical consensus, however, is that the weak learners for Boosting should truly be weak, while the "weak learners" for Bagging should actually be strong. In tree terminology, one should use small trees when Boosting and big trees for Bagging. In this article a size of the tree is defined as the number of nodes, often is treated as a depth (in other research work). In intuitive "bias-variance" terms, we can say that Bagging is mainly a variance reduction (or stabilization) operation, while Boosting, in the way that it flexibly combines models, is also a bias reduction operation, i.e., it adds diversity to the representation beyond that of a single learner.

Boosting is a bias reduction technique in contrast to Bagging. Boosting typically improves the performance of a single tree model. The reason for this is that we often cannot construct trees which are sufficiently large due to the thinning out of objects in the terminal nodes. Boosting is then a device used to come up with a more complex solution by taking a linear combination of trees. In the presence of high-dimensional predictors, Boosting is also very

* Corresponding author.
*E-mail addresses:* jan.kozak@us.edu.pl (J. Kozak), urszula.boryczka@us.edu.pl (U. Boryczka).

http://dx.doi.org/10.1016/j.knosys.2014.11.027
0950-7051/

useful as a regularization technique for additive or interaction modeling.

AdaBoost [3] gains its strength from using ensembles of weak learners (i.e., Boosting the performance of the weak learner) and from adaptively choosing objects' weights based on a classification of the previous iteration. One adaptively boosts the weak learners – hence the name of the algorithm. In each step, the weights of correctly classified objects are decreased, while the weights for misclassified objects are increased. AdaBoost, like Bagging, forms a collection of classifiers by applying a single base learning algorithm to successive derived training sets formed by sampling from a base training set. However, the main difference is that AdaBoost associates a weight with each objects in the training set. AdaBoost appears to have a greater average effect, thus leading to, on average, substantially larger error reductions than Bagging.

In this article we propose an approach to construct classifiers based on computational intelligence methods. Similar attempts have been undertaken by Liu et al. [4], where genetic algorithm is used in ensemble approach based on single classifiers (decision rules). Another approach, based on genetic algorithm is proposed in [5], where the problem of optimization the number of classifiers in ensemble is discussed. Due to genetic approach the number of week classifiers is diminished as well as good accuracy of classification is maintained.

The next application of computation intelligence is discussed in the work of Hidaka and Kurita [6]. Authors presented the new application of Particle Swarm Optimization (PSO) in ensemble classifiers. The proposed of this work is to sequentially apply the classification algorithm to repeatedly modified the weights of objects. Whereas, the optimization of the depth of the tree has been studied by Moshkov et al. using this i.a. dynamic programming [7,8].

The aim of proposed method in contrary to the described approaches, is to construct the homogeneous classifiers using the Ant Colony Decision Forest (ACDF) algorithm. The homogeneous classifiers are ensemble methods that use the same algorithm (over diversified data sets) for the construction of individual classifiers. In contrast, heterogeneous use different learning algorithms over the same data. An effect of our previous works, concerning Ant Colony Decision Trees (ACDT) and ACDF approaches, described in [9–11] shows us a purpose of the current works. The application of pheromone updatings as well as a treatment of the decision tree as an a-cyclic directed graphs, opposite to another approaches – resulted in better ensemble classifiers performance.

The Ant Colony Optimization allows to find multiple local optima in a single algorithm run. One classifier may correspond to one optimum. The Ant Colony Optimization may be used for building the meta ensemble – it allows to search the solution space and select only the proper local optima. Particular classifiers create the meta ensemble.

Our proposition – an ensemble method with virtual ants leads to some effects called: autocatalysis, positive feedback method in pheromone representations and finally selforganisation in our ant colony behavior. We can noticed some correlation between these mechanisms and innovation in the area of data mining. This new form of innovative methodology leads to collective intelligence spreading within this artificial organism. This innovation should be summarized as a new form of simulation, concentrating on the "life" of our system. The wisdom of artificial ants in ACDF depends on the number of attributes and its values and consequently on the number of nodes included in decision trees. The number of nodes and the strength of the connections expressing by pheromone values arises from the aggregation of the ACO approach as well as the CART algorithm. There is an outbreak of ant-based initiative and a new approach for decision tree construction paradigm may be called "embedded innovation". But surely it requires some deeper understanding of the information-knowledge transition and answering questions. First of all we must realize that such decentralized system of virtual ants convert pieces of information gathered in attribute–values pairs into collective intelligence. Secondly we need to understand the power of exchanging information via pheromone-collaborative efforts and using different channels called in this structure as a heterarchy [12,13]. Making such innovation in data mining tasks for the reinforcement in learning by ants in ACDF (collective intelligence) is intended to provide a comprehensive approach to difficult, challenging and significant classification problems.

This article is organized as follows: Section 1 comprises an introduction to the subject matter of this article and related work. Section 2 reviews Ant Colony Optimization in Data Mining. Decision Trees and Ant Colony Decision Trees are presented in Sections 3 and 4. Section 5 describes the ensemble methods (Bagging, Random Forests and Boosting). Section 6 focuses on the ACDF approach, especially on the self-adaptive ACDF approach and Boosting ideas in the ACDF approach. Section 7 presents the experimental study that was conducted to evaluate the performance of multiple Boosting in the ACDF meta-classifier by taking into consideration twelve data sets. Finally, we conclude with general remarks on this work, and some directions for future research are pointed out.

## 2. Ant Colony Optimization in data mining

Machine learning techniques are being utilized to learn models over increasingly large feature and example spaces. An attractive option for learning from large datasets is distributed learning. The approach discussed here is to learn an ensemble of individual classifiers inspired by ant colonies, with each learner creating its own classifier from a subset of the total data set.

Ant Colony Optimization is a branch of a newly developed form of artificial intelligence called swarm intelligence. Swarm intelligence is a form of emergent collective intelligence of groups of simple individuals, e.g. ants, termites or bees, in which a form of indirect communication via pheromones was observed. Pheromone values encourage ants to follow a path in order to build good solutions for the analyzed problem, and the learning process occurring in this situation is called positive feedback or autocatalysis.

In this paper we defined an ant algorithm that would be a multi-agent system as we were inspired by observations of real ant colony behavior exploiting the *stigmergic* communication paradigm. The optimization algorithm in this paper was inspired by previous works on Ant Systems (AS) and, in general, by the term *stigmergy*. This phenomenon was first introduced by Grasse [14,15].

An essential step in this direction was development of the Ant System by Dorigo and Stützle [16]. It was a new type of heuristic inspired by analogies to the foraging behavior of real ant colonies, which has proven to work successfully in a series of experimental studies. Diverse modifications of AS have been applied to many different types of discrete optimization problems and have produced very satisfactory results [17]. Recently, the approach has been extended by Dorigo et al. [18–21] to a full discrete optimization metaheuristic, called the Ant Colony Optimization (ACO) metaheuristic (Algorithm 1).