Feature Selection for Classification Using an Ant System Approach

Nadia Abd-Alsabour

HAL Id: hal-01054496

https://inria.hal.science/hal-01054496

Submitted on 7 Aug 2014

Feature Selection for Classification Using an Ant System Approach

Nadia Abd-Alsabour[1]

[1] School of Computer Science, Engineering and mathematics, Flinders University of South Australia
nadia.abdalsabour@ieee.org

Abstract. Many applications such as pattern recognition and data mining require selecting a subset of the input features in order to represent the whole set of features. The aim of feature selection is to remove irrelevant, redundant or noisy features while keeping the most informative ones. In this paper, an ant system approach for solving feature selection for classification is presented. The results we got are promising in terms of the accuracy of the classifier and the number of selected features in all the used datasets.

Keywords: Ant colony optimization, pattern recognition, support vector machine and feature selection.

1 Introduction

Pattern recognition is the assignment of an input pattern to one of several predefined categories/classes [1]. The basic component of any pattern recognition system is the classifier whose task is to partition the feature space into class-labeled decision regions. The performance of the classifiers is sensitive to the choice of the features that are used for constructing those classifiers. The choice of the features that are presented to the classifiers affects the following important things:

- The accuracy of the classifiers,
- The time needed for learning the classification function, and
- The number of examples needed for learning the classification function [2].

Some research suggests increasing the examples amount of training data, but this affects the time needed for the learning. It is here that feature selection becomes important. The assumption that more features can offer more information about the inputs is not always valid in practice. It has been found that including more features can be time consuming and may lead to finding a less optimal solution. This makes feature selection from the original set of features is highly desirable in many situations [1], [3].

Feature selection (FS) is the problem of selecting a subset of features without reducing the accuracy of representing the original set of features [3]. Feature selection (the most general term is variable selection) is used in many applications to remove irrelevant and redundant features where there are high dimensional datasets. These

datasets can contain high degree of irrelevant and redundant features that may decrease the performance of learning algorithms.

The main approaches that are used for solving feature selection problem can be classified into filter or wrapper approach depending on whether or not feature selection is done independently of the learning algorithm. Some researchers use a hyprid approach to take the advantage of these 2 approaches and to handle large datasets.

Feature selection can be seen as an optimization problem that involves searching the space of possible feature subsets to identify the optimal one. Many optimization techniques such as genetic algorithms (GA) [2], tabu search (TS), simulated annealing (SA) and ant colony optimization algorithms (ACO) have been used for solving feature selection.

Real ants are able to find the shortest path between their nest and food sources because of the chemical substance (pheromone) that they deposit on their way. The pheromone evaporates over time so the shortest paths will contain much pheromone and subsequently will attract more ants in future.

Ant colony optimization algorithms simulate the foraging behavior of some ant species [5]. ACO algorithms are guided search algorithms that use 2 factors for guiding the search process. These factors are: 1) the pheromone values (numerical values as a simulation for the pheromone that real ants deposit on their way from/to their nest). 2) Heuristic information that is crucial for good performance of the system when we can not use local search. There are 2 types of heuristic information used by ACO algorithms; static heuristic information (that is computed at the initialization time and then remains unchanged throughout the whole algorithm's run such as the distances between cities in traveling salesman problem) and dynamic heuristic information (that depends on the partial solution constructed so far and therefore it is computed at each step of an ant's walk).

One of the recent trends in ACO is to apply them in solving new optimization problems such as applying them in solving many industrial problems proving that these algorithms are useful in real-world applications [6]-[8]. Recently, many researchers adopted some ACO algorithms for the solutions of feature selection problem such as in [9]-[12].

The traditional ACO algorithms were designed for solving ordering problems such as traveling salesman problem and quadratic assignment problem [11]. Feature selection problem is different from these optimization problems in terms of there is no prior information known about the features such as in TSP where the distances between the cities are known in advance for guiding the search process besides the pheromone values.

In order to solve an optimization problem using an ACO algorithm, the problem should be represented as a fully connected construction graph and 2 factors are used for guiding the search process. These factors are the heuristic information (known in advance about the given problem) and the pheromone values.

In the traditional ACO algorithms, the pheromone values are associated with the nodes or the edges of the construction graph representing the problem (depending on the chosen problem representation), which may also contain heuristic information representing prior information about the given problem [6]-[7].

In the proposed algorithm, we did not use the graphical representation hence; there is no concept of path. We associate the pheromone with each feature. Although there is no heuristic information known in advance in this type of problems, we used heuristic information in computing the moving probability that is used by each ant to select a particular feature in each construction step. We used the proportional of the ants chose a particular feature as heuristic information so the proposed algorithm does not need prior knowledge of features. We also used a new equation for the pheromone update as we will explain in the following sections.

The rest of this paper is organized as follows. Section 2 addresses the fundamentals of 4 ACO algorithms based on which we developed the proposed algorithm. The third section explains the proposed algorithm. Section 4 details the experiments carried out and presents the obtained results. The discussion of the results is presented in section 5. Section 6 highlights future work in this area. And finally, section 7 concludes this paper.

2   Related Work

In this section, we explain ant system (AS) and ant colony system (ACS) in solving traveling salesman problem since we used many aspects of these approaches. We will then explain briefly 2 ACO algorithms for solving feature selection.

The first ant colony optimization algorithm is AS as proposed in the early nineties [13]. Since its appearance, it became the basis for many successive ACO algorithms and it is known as the original ACO algorithm. AS is an iterative algorithm where in each iteration each ant selects the next city to be visited using the following equation:

$$\rho_{ij}^{k} = \frac{\tau_{ij}^{\alpha} . \eta_{ij}^{\beta}}{\sum_{c_{ij} \in N(s^{\rho})} \tau_{il}^{\alpha} . \eta_{il}^{\beta}} \qquad \text{if } c_{ij} \in N(s^{\rho}) \text{ and } 0 \text{ otherwise} \tag{1}$$

where $N(s^{\rho})$ is the set of feasible components. The parameters α and β control the relative importance of the pheromone versus the heuristic information η_{ij} that is equal to $\frac{1}{d_{ij}}$ where d_{ij} is the distance between city i and city j.

At the end of each iteration, the pheromone values (numerical values associated with each solution components- here are the edges) are updated by all ants that have built solution according to the following equation:

$$\tau_{ij} = (1 - \rho) . \tau_{ij} + \sum_{k=1}^{m} \Delta \tau_{ij}^{k} \tag{2}$$

Where ρ is the evaporation rate, m is the number of ants and $\Delta\tau_{ij}^k = Q/L_k$ where Q is a constant and L_k is the length of the tour constructed by ant k.

Ant colony system (ACS) is considered one of the most successful ACO algorithms [14]. Since its appearance, it has been using for solving many optimization problems. ACS is an iterative algorithm where at each iteration, each ant chooses the next city to be visited (j) using pseudorandom proportional rule that is computed according to the following equation:

$$\arg\max_{C_{il}\in N(s^P)} \{\tau.\eta_{il}^\beta\} \tag{3}$$

This equation is used by each ant at each construction step to choose the next city depending on a random variable q and a parameter q_0 and it is used if q<= q_0.

After each construction step, the local pheromone update is performed by all ants to the last edge traversed according to the following equation:

$$\tau_{ij} = (1-\varphi).\tau_{ij} + \varphi.\tau_0 \tag{4}$$

Where $\varphi \in (0,1]$ and τ_0 is the initial pheromone.

Local pheromone update leads to decreasing the pheromone values on the edges that encourages subsequent ants to choose other edges and subsequently produce different solutions.

At the end of each iteration, the pheromone values are updated by only the best ant according to the following equation:

$$\tau_{ij} = \begin{cases} (1-\rho).\tau_{ij} \\ \tau_{ij} \quad otherwise \end{cases} + \rho.\Delta\tau_{ij} \quad if \ (i,j) \ belongs \ to \ best \ tour, \tag{5}$$

Where $\Delta\tau_{ij} = 1/L_{best}$ and L_{best} is the length of the tour constructed by the best ant.

An ACO approach called antselect for variable selection in quantitative structure-activity relationship (QSAR) has been developed in [9]. In antselect, a weight is associated with each feature and used for calculating the probability with which the feature is randomly selected by an ant. Initially, the weights and the probabilities are equal for all variables. The moving probability is calculated according to the following equation:

$$\rho_k = \frac{\omega_k}{\sum_k \omega_k} \tag{6}$$

where ω_k is the weight associated with feature k.

The weights are updated according to the following equation:

$$\omega_k = (1-\rho).\omega_k + \frac{\Delta\omega}{L} \qquad (7)$$

where ρ is the evaporation rate, $\Delta\omega$ is a constant factor, and L is the length of ant k's path. In [10], antselect was used with artificial neural networks for variable selection with different dataset.

A modified ant colony optimization algorithm for solving feature selection in QSAR [11] expresses the feature selection problem in a binary notation where an ant moves in an N-dimensional search space of N variables. The pheromone levels on each feature are divided into two kinds, τ_{i0} and τ_{i1}.

The pheromone levels are updated according to the appropriate one of the following two updating rules:

$$\tau_{i0}(new) = \rho\tau_{i0}(old) + \Delta\tau_{i0} \qquad (8)$$

or

$$\tau_{i1}(new) = \rho\tau_{i1}(old) + \Delta\tau_{i1} \qquad (9)$$

$$\text{where} \quad \Delta\tau_{i0} = \sum_{k=1}^{m} \Delta\tau_{i0}^{k} \qquad (10)$$

$$\text{And} \quad \Delta\tau_{i1} = \sum_{k=1}^{m} \Delta\tau_{i1}^{k} \qquad (11)$$

The moving probability for any feature is 0 or 1, where 1 means that this feature will be selected and 0 means the inverse. The moving probability is calculated according to the following equation:

$$\rho_i^k = \frac{\tau_{i1}}{\tau_{i1} + \tau_{i0}} \qquad (12)$$

This algorithm has also been used for solving feature selection problem in different dataset in [12].

3 The Proposed Algorithm

The proposed algorithm is a wrapper-based system that deals with the problem of feature selection as a binary problem where a set of binary bits (of a length equivalent to the number of the features in a given dataset) is associated with each ant. If the nth bit is a 1 this means that feature number n in the dataset is selected, otherwise this feature is not selected. Thus, the concept of path in the traditional ACO algorithms does not pertain here. At the start of the algorithm, the bits are randomly initialized to zeros and ones.

The pheromone values are associated with the features. At each construction step, each ant selects a feature out of all the features with the probability computed according to the following equation:

$$\rho_i = \tau_i . \Delta \tau_i \tag{13}$$

where τ_i is the pheromone value associated with feature i and $\Delta \tau_i =$ proportion of the ants that selected this feature and acts as heuristic information that represents the desirability of feature i. In ACO algorithms, the design of the moving probability is critical. Here, we used both the pheromone values and heuristic information to compute it.

At the end of each iteration, the pheromone values are updated according to the following equation:

$$\tau_i = ((1-\rho) . \tau_i) \quad + \quad \left(\rho . \sum_{k=1}^{m} \frac{Q}{L_k} \right)^{\beta} \tag{14}$$

where $L_k =$ no. of features selected by ant k and p is the evaporation rate. This equation is used by all ants at the end of each iteration to all features that have been chosen.

The updating of the pheromone values is important to reinforce those features that lead to high quality feature subsets. Features that belong to good solutions will contain larger pheromone. Consequently, these features tend to be selected more often.

The pseudo code of the proposed algorithm is as follows:

```
Procedure of the proposed algorithm

Initialization

While (not terminate) do
   construct ant solution
   build SVM
   update statistics
   update pheromone values by all ants
 end while
end.
```

4 Tests and results

In order to test the proposed algorithm, we used it with a Support Vector Machine (SVM) learning algorithm, which is considered one of the most popular, powerful and efficient classification and regression methods. Although not all machine learning algorithms require the phase of feature selection, feature selection is important in building SVM-based regression and classification as well [1], [3]. We used the number of correct classification / the whole number of observations as a fitness function where each ant evaluates its solution based on its ratio of correct classifications.

4.1 Datasets

In order to test the proposed algorithm, we did several experiments using several datasets. In our experiments, we did not do any modifications to these datasets before using them rather than converting them into a suitable format for our systems. In our experiments, we used 5-fold cross validation (CV). The used datasets here are: backache, prnn_virus3, prnn_viruses, analcatdata_authorship, and analcatdata_marketing from statistical datasets available in .arff format from the Website of Waikato University [15].

4.2 Methods

In our experiments, we developed the following 2 systems:
- SVM: that uses the entire set of features (without the phase of feature selection), and
- SVM-FS: that uses subset of features selected by the proposed algorithm (with the phase of feature selection).

In this paper, we focus on testing the effect of feature selection on the performance of the classifier. So we did not optimize the performance of SVM although further investigation is highly required since it affects the performance of the whole system. We used the default values to its parameters in both cases, with and without the use of feature selection. We used C-classification SVM of package (e1071) of R language with the default values to its parameters (cost, gamma, and epsilon).

In these 2 systems, we used 5-fold CV. The number of ants was set to the number of the features in any dataset. The initial pheromone was set at 1. The number of iterations is 10 iterations. P was set at 0.3. β was set at 0.3. Q was set at 4.

4.3 Results

Table 1 shows the results of these 2 systems using the above mentioned datasets. The results for the proposed algorithm represent the average of 5 independent runs. These systems are implemented using R language [16]-[17] and WEKA machine learning tool [18]-[19]. All the experiments were run on a personal PC with 2 GHz CPU and 2 GB RAM.

Table 1: The accuracy of SVM with and without the use of feature selection

	Dataset Name	No. of original features	Avg. no. of selected features	SVM (without FS)	SVM (with FS)
1	backache	32	20.2	0.9	0.9222
2	prnn_virus3	17	13.2	0.9474	0.9789
3	prnn_viruses	17	10.8	0.0164	0.9016
4	analcatdata_authorship	70	33	1	1
5	analcatdata_marketing	32	27.2	0.6401	0.6561

5 Discussion

The proposed algorithm deals with feature selection problem as a binary one but uses many concepts from ant system although it does not have the concept of path.

The previous results show that SVM-FS with the proposed algorithm for performing feature selection outperforms SVM that uses all the features in all the used datasets (the accuracy of SVM is larger than that of SVM with all features). The number of features selected by the proposed algorithm is significantly smaller than the total number of the features in the original datasets in all of the used datasets.

In the proposed algorithm, we used:

- Heuristic information in calculating the moving probability. This heuristic information indicates to how often a particular feature has been chosen by different ants. There are many ideas that could be used as heuristic information for guiding the search process besides the pheromone values but we see this equation is very useful based on many experiments we have done.
- Different pheromones update equation rather than the usual ones used in ACO for feature selection.
- Pheromone evaporation as in ant colony system.

6 Future Work

Trying other classifiers and additional experiments with other datasets are currently in progress.

Although the initial results are promising, further investigation is required particularly in solving the problem of the big number of parameters that the proposed algorithm suffers from (its parameters besides the parameters of SVM-all of these parameters need to be adjusted especially that the performance of ACO algorithms in general is sensitive to its parameters) in order to enhance its performance.

Another direction for future work is conducting comparisons of the performance of the proposed algorithm with that of other stochastic algorithms used for solving feature selection.

7 Conclusion

In this paper, we solved feature selection problem for classification using an ant system approach on an SVM classifier with several datasets. We used heuristic information in order to guide the search process besides the pheromone values as in most of conventional ACO algorithms. The results we got are promising in terms of the solution quality and the number of selected features.

References

1. Duda, R.O., Hart, P.E., and Stork, D.G.: Pattern classification and scene analysis 2nd edition. Wiley, (2001)
2. Yang, J., Honavar, V.: Feature subset selection using a genetic algorithm. IEEE Intelligent Systems, Vol. 13, No. 2, PP. 44--49, (1998)
3. Liu, H., Motoda, H.: Feature Selection for Knowledge Discovery and Data Mining. Kluwer Academic, Norwell, MA, (1998)
4. Guyon, I., and Elisseeff, A.: An introduction to variable and feature selection. Journal of Machine Learning Research, Vol. 3, PP. 1157--1182, (2003)
5. Dorigo, M., Bonabeou, E., and Theraulaz, G.: Inspiration for optimization from social insect behavior. Nature, Vol. 406, PP. 39--42, (2000)
6. Dorigo, M., and Stutzle, T.: Ant Colony Optimization. MIT Press, Cambridge, Massachusetts, (2004)
7. Dorigo, M., Birattari, M., and Stutzle, T.: Ant colony optimization- artificial ants as a computational intelligence technique. IEEE computational intelligence magazine, (2006)
8. Dorigo, M., Caro, G.D., and Sampels, M.: Ant Algorithms. Springer-Verlag (2002)
9. Izrailev, S., and Agrafiotis, D., A novel method for building regression tree models for QSAR based on artificial ant colony systems. J. Chem. Inf. Computer Science, Vol. 41, PP. 176--180, (2001)
10. Izrailev, S., Agrafiotis, D.: Variable selection for QSAR by artificial ant colony systems. SAR QSAR in environmental research, Vol. 13, PP. 417--423, (2002)
11. Shen, Q., Jiang, J.H., Tao, J.C., Shen, G.L., and Yu, R.Q.: Modified Ant Colony Optimization Algorithm for Variable Selection in QSAR Modeling: QSAR Studies of Cyclooxygenase Inhibitors. J Chem. Inf. Model, Vol. 45, PP. 1024--1029, (2005)
12. Shi, W., Shen, Q., Kong, W., and Ye, B.: QSAR analysis of tyrosine kinase inhibitor using modified ant colony optimization and multiple linear regression. European Journal of Medicinal Chemistry. Vol. 42, PP. 81--86, (2007)
13. Dorigo, M., Maniezzo, V., Colorni, A.: Ant System: optimization by a colony of cooperating agents. IEEE Transactions on System, Man, and Cybernetics Vol. 26, No. 1, PP. 1--13, (1996)
14. Dorigo, M., and Gambardella, L.M.: Ant colony system: A cooperating leaning approach to the traveling salesman problem. IEEE Transactions on evolutionary computation, Vol.1, No. 1, PP. 53--66, (1997)
15. http://www.cs.waikato.ac.nz/ml, viewed 25th January 2010.
16. R: A Language and Environment for Statistical Computing 2006 [http://www.R-project.org]. R Foundation for Statistical Computing, Vienna, Austria.
17. Dalgaard, P.: Introductory statistics with R. Springer, (2008)
18. http://www.cs.waikato.ac.nz/ml/weka, viewed 25th January 2010.

19. Witten, I., and Frank, E., Data Mining: Practical machine learning tools and techniques, 2nd Edition, Morgan Kaufmann, San Francisco, (2005)