AIML 05 Conference, 19-21 December 2005, CICC, Cairo, Egypt

ICGST www.icgst.com AIML

An Ant Colony Optimization Based Approach for Feature Selection

Ahmed Al-Ani
Faculty of Engineering, University of Technology, Sydney
PO Box 123, Broadway, NSW 2007, Australia
ahmed@end.uts.edu.au

Abstract

This paper presents a new feature subset selection algorithm based on the Ant Colony Optimization (ACO). ACO is a metaheuristic inspired by the behaviour of real ants in their search for the shortest paths to food sources. It looks for optimal solutions by utilizing distributed computing, local heuristics and previous knowledge. We modified the ACO algorithm so that it can be used to search for the best subsets of features. A new pheromone trail update formula is presented, and the various parameters that lead to better convergence are tested. Results on speech classification problem show that the proposed algorithm achieves better performance than both greedy and genetic algorithm based feature selection methods.

***Keywords:** Feature selection, Ant colony optimization, Ant system, Pattern recognition.*

1. Introduction

Pattern recognition is an important and multi-disciplinary field of research with wide range of applications that include handwriting recognition, speech recognition, medical diagnosis, fingerprint verification and face recognition. Among the several parameters that affect the performance of patter recognition systems, feature representation of patterns can be considered the most important. In some applications, it might be sufficient to use simple features that are previously known. However, in other applications, unique feature sets that are necessary and sufficient to the classification task do not exist. Moreover, the assumption that more features can offer complementary information about the patterns to be classified is not always valid. It has been found that including more features will make the classification and analysis more difficult, time consuming and may even lead to poorer generalization on unseen data. This makes feature selection and reduction in feature set dimensionality very desirable.

Ideally, the best subset of features can be found by evaluating all the possible subsets, which is known as exhaustive search. If we have a set of N features, then there will be $2^N - 2$ candidate subsets. This procedure may be practically impossible even for a moderate-size feature set, e.g. for $N = 20$, there will be 1,048,574 subsets. On the other hand, examining features individually may not be sufficient, as it is important to take into consideration the interaction between features [1]. A more practical approach would be computationally feasible and aims at achieving optimal or "semi-optimal" solutions. Hence, several search procedure methods have been developed, which basically differ in their computational cost and the optimality of the subsets they find. In addition to the search procedure, a feature subset evaluation measure is needed to evaluate the importance of subsets. The existing evaluation measures can be broadly divided into two main groups: filters and wrappers. Filters operate independently of any learning algorithm, where undesirable features are filtered out of the data before learning begins. On the other hand, performance of classification algorithms is used to select features for wrapper methods [2].

In recent years, population-based optimization algorithms have attracted a lot of attention. Such methods attempt to achieve better solutions by utilizing knowledge from previous iterations. One of the promising population-based algorithms is the Ant Colony Optimization (ACO) [3]. The ant algorithm was inspired by the real ants' behaviour in their search of food, and targets discrete optimization problems. The coordination of a population of ants takes place through indirect communication, which is mediated by laying an odorous substance on food paths. This will increase the probability that other ants will follow those specific marked paths.

In this paper, we propose a new feature selection algorithm that searches the feature space using a modified ACO algorithm. In the next section, we give a brief description of some of the well-known feature subset search algorithms. Section three explains the ACO metaheuristic. The proposed feature subset search algorithm is presented in section four. Experimental results are presented in section five and a conclusion is given in section six.

2. Available Feature Subset Search Algorithms

Due to the importance of feature selection, many feature subset search algorithms have been proposed in the literature. Two of the most famous approaches are stepwise search and Genetic Algorithms (GA).

The stepwise (or greedy) search adds/removes a single feature to/from the current subset [4]. It considers local changes to the current feature subset. Often, a local change is simply the addition or deletion of a single feature from the subset. The stepwise, which is also called the Sequential Forward Selection (SFS)/ Sequential Backward Selection (SBS) is probably the simplest search procedure and is generally sub-optimal and suffers from the so-called "nesting effect". It means that the features that were once selected/deleted cannot be later discarded/re-selected. To overcome this problem, Pudil et al. [5] proposed a method to flexibly add and remove features, which they called "floating search".

Another famous search approach is based on the Genetic Algorithm (GA). The GA is a combinatorial search technique based on both random and probabilistic measures. Subsets of features are evaluated using a fitness function and then combined via cross-over and mutation operators to produce the next generation of subsets [6]. The GA employ a population of competing solutions, evolved over time, to converge to an optimal solution. Effectively, the solution space is searched in parallel, which helps in avoiding local optima. A GA-based feature selection solution would typically be a fixed length binary string representing a feature subset, where the value of each position in the string represents the presence or absence of a particular feature. According to [7, 8], the GA was able to achieve better performance than other conventional methods.

We propose in this paper an ant system approach for feature subset selection that aims at achieving similar or better results than GA-based feature selection.

3. Ant Colony Optimization

There has been an increasing interest in studying the behaviour of animals and insects, and in particular how they interact to achieve their goal. Among the various animals and insects, ants have attracted a lot of attention.

Scientists have found that an odorous substance, known as pheromone, is used as an indirect communication medium between ants. When a source of food is found, ants lay some pheromone to mark the path between the nest and the food source. The quantity of the laid pheromone depends upon the distance, quantity and quality of the food source. While an isolated ant that moves at random detects a laid pheromone, it is very likely that it will decide to follow its path. This ant will itself lay a certain amount of pheromone, and hence enforce the pheromone trail of that specific path. Accordingly, the path that has been used by more ants will be more attractive to follow. This process is hence characterized by a positive feedback loop [3].

A number of experiments have been conducted by Deneubourg et al. [9] to study the behaviour of ants when they are forced to choose between paths that vary in their lengths. In one experiment a bridge of two branches was placed between the ants' nest and the food source, where one branch was twice as long as the other branch (see Figure. 1.a). Initially, when ants arrived at the bridge, they randomly chose between the two branches. However, it is obvious that ants that chose the short branch would reach the food and start their journey back to the nest faster than those that chose the long branch. Accordingly, pheromone would start to accumulate faster on the short branch and hence influence more ants to choose the short branch. After a certain period of time, the vast majority of the ants chose the short branch.

In another experiment, only the long branch was offered and after 30 minuets, the short branch was added (see Figure 1.b). Because ants were only using the long branch for the first 30 minuets, a large quantity of pheromone was laid on this branch. Hence, even after adding the short branch, ants were still choosing the long branch, because they were influenced by the high pheromone concentration.

It is important to mention that if ants stop laying pheromone on a certain path, then the pheromone intensity on that path will decrease over time. This process favours exploration of new paths, and is known as pheromone evaporation. Note that in the second experiment, the slow pheromone evaporation rate could not allow ants to disregard the long path and search for a more optimal and shorter path.

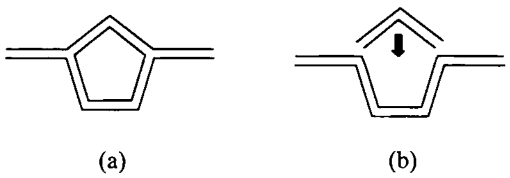

Figure 1. Two bridge experiments aim at measuring the usage of short and long paths

Dorigo et. al. [10] adopted the concept of ants' foraging behaviour and proposed an artificial colony of ants algorithm. The algorithm was called the Ant Colony Optimization (ACO) metaheuristic, and aimed at solving hard combinatorial optimization problems. The ACO was originally applied to solve the classical traveling salesman problem [3], where it was shown to be an effective tool in finding good solutions. The ACO has also been successfully applied to other optimization problems including data mining and telecommunications networks [11, 12].

In order to solve an optimization problem, a number of artificial ants are used to iteratively construct solutions. In each iteration, an ant would deposit a certain amount of pheromone proportional to the quality of the solution. At each step, every ant computes a set of feasible expansions to its current partial solution and selects one of these depending on two factors: local heuristics and prior knowledge.

For the classical Traveling Salesman Problem (TSP) [3], each artificial ant represents a simple "agent". Each agent explores the surrounding space and builds a partial solution based on local heuristics, i.e., distances to neighboring cities, and on information from previous attempts of other agents, i.e., pheromone trail or the usage of paths from previous attempts by the rest of the agents. In the first iteration, solutions of the various agents are only based on local heuristics. At the end of the iteration, "artificial pheromone" will be laid. The pheromone intensity on the various paths will be proportional to the optimality of the solutions. As the number of iterations increases, the pheromone trail will have a greater effect on the agents' solutions. The ACO makes probabilistic decision in terms of the artificial pheromone trails and the local heuristic information. This allows ACO to explore larger number of solutions than greedy heuristics. Another characteristic of the ACO algorithm is the pheromone trail evaporation, where according to [10], pheromone evaporation helps in avoiding rapid convergence of the algorithm towards a sub-optimal region.

Note that searching the feature space in the problem of feature selection is quite different from the other optimization problems that researchers attempted to solve using ACO. We have recently applied ACO to the problem of feature selection with initial encouraging results [13]. In the next section, we present an expansion to our previous work and a detailed description of the new feature subset search approach.

4. The Proposed Feature Selection Algorithm

The object of feature selection is to find a smaller subset of features that minimizes the probability of error. Thus, given the original feature set, $\mathcal{F}$, of n features, we need to find subset $\mathcal{S}$, which consists of m features ($m < n$, $\mathcal{S} \subset \mathcal{F}$), such that the Mean Square Error (MSE) between the classification result and the target output is minimized.

Concepts from ants' foraging and Dorigo's ACO algorithm are used in our proposed feature subset search procedure. Similar to the original ACO algorithm, a number of artificial ants are used to iteratively construct solutions in the proposed algorithm. However, instead of accumulating pheromones, as the original ACO algorithm does, the proposed algorithm estimates the pheromone intensities at each iteration. This will favour exploration and reduce the possibility of being trapped in local minima (as in the case of Figure 1.b). In addition, unlike the original ACO that builds sequential solutions at each iteration, the proposed algorithm only changes a small number of features in subsets that are selected by the best ants. This will reduce the computational complexity as the size of the selected feature set gets larger. We propose to use a hybrid evaluation measure that estimates the overall performance of subsets as well as the local importance of features. A classification algorithm is used to estimate the performance of subsets (i.e., wrapper evaluation function). On the other hand, the local importance of a given feature is measured using the Mutual Information Evaluation Function (MIEF) [14], which is a filter evaluation function.

The following parameters are used in the algorithm:

- n: number of features that constitute the original set, $\mathcal{F} = \{f_1, \ldots, f_n\}$.
- na: number of artificial ants to search through the feature space.
- $\mathcal{T}_i$: intensity of pheromone trail associated with feature f_i.
- $\mathcal{S}_j = \{s_1, \ldots, s_m\}$: a list that contains the selected feature subset for ant j.
- $\mathcal{PL}$: list of the previously tested subsets
- k, where the best k subsets ($k < na$) will be used to influence the feature subsets of the next iteration.
- $\mathcal{BL}$: list of the best k subsets.

In the first iteration, each ant will randomly choose a subset of m features. In the second and following iterations, each ant will start with $m - p$ features that are randomly chosen from the previously selected k-best subsets, where p is an integer that ranges between 1 and $m - 1$. In this way, the features that constitute the best k subsets will have more chance to be present in the subsets of the next iteration. Nevertheless, it will still be possible for each ant to consider other features as well. For instance, ant j will consider those features that achieve the best compromise between previous knowledge, i.e., pheromone trails, and local importance. The local importance of feature f_i is measured with respect to the features of $\mathcal{S}_j$ (features that have already been selected by ant j). The Selection Measure (SM) is used for this purpose and is defined as:

$$SM_i^{S_j} = \begin{cases} \dfrac{(\mathcal{T}_i)^{\eta}\left(LI_i^{S_j}\right)^{\kappa}}{\sum\limits_{g \notin S_j} (\mathcal{T}_g)^{\eta}\left(LI_g^{S_j}\right)^{\kappa}} & \text{if } i \notin S_j \\ 0 & \text{Otherwise} \end{cases} \qquad (1)$$

where $LI_i^{S_j}$ is the local importance of feature f_i given the subset $\mathcal{S}_j$. The parameters η and κ control the effect of trail intensity and local feature importance respectively. $LI_i^{S_j}$ is defined as:

$$LI_i^{S_j} = I(C; f_i) \times \left[\frac{2}{1 + \exp(-\alpha D_i^{S_j})} - 1\right] \qquad (2)$$

where

$$D_i^{S_j} = \min_{f_s \in S_j}\left[\frac{H(f_i) - I(f_i, f_s)}{H(f_i)}\right] \times \frac{1}{|S_j|}\sum_{f_s \in S_j}\left[\beta\left(\frac{I(C;\{f_i, f_s\})}{I(C;f_i) + I(C;f_s)}\right)^{\gamma}\right] \tag{3}$$

The parameters α, β, and γ are constants, H(f_i) is the entropy of f_i, I(f_i; f_s) is the mutual information between f_i and f_s, I(C; f_i) is the mutual information between the "class labels" and f_i, and $|S_j|$ is the cardinal of S_j. For detailed explanation of the MIEF measure, the reader is referred to [14].

Below are the steps of the algorithm:

1. Initialization:
   - Set $\mathcal{T}_i = cc$, where cc is a constant.
   - Define the maximum number of iterations.
   - Define k, where the k-best subsets will influence the subsets of next iteration.
   - Define p, where $m - p$ is the number of features each ant will start with in the second and following iterations.
2. If in the first iteration,
   - For $j = 1$ to na,
     - Randomly assign a subset of m features to S_j.
   - Goto step 4.
3. Select the remaining p features for each ant:
   - For $mm = m - p + 1$ to m,
     - For $j = 1$ to na,
       - Given subset S_j, Choose feature f_i that maximizes $SM_i^{S_j}$.
       - $S_j = S_j \cup \{f_i\}$.
   - Replace the duplicated subsets, if any, with randomly chosen subsets.
4. Evaluate the selected subset of each ant using a chosen classification algorithm:
   - For $j = 1$ to na,
     - Estimate the Mean Square Error (MSE_j) of the classification results obtained by classifying the features of S_j.
   - Sort the subsets according to their MSE. Update the minimum MSE (if achieved by any ant), and store the corresponding subset of features.
   - Update the list of the previously tested subsets. $\mathcal{PL} = [\mathcal{PL}\,;\, S_j]$, where ($j$=1:$na$).
5. Update $\mathcal{BL}$ (the list of the k best subsets).
6. For each feature f_i, update the pheromone trail according to the following formula:

$$\mathcal{T}_i = a_1 R_{1i} + a_2 R_{2i} + a_3(1 - R_{3i}) + a_4 \tag{4}$$

where

- a_1, a_2, a_3, and a_4 are constants.
- R_{1i}: ratio indicating the occurrence of f_i in $\mathcal{BL}$.
- R_{2i}: ratio between the occurrence of f_i in the best half subsets and the overall occurrence of f_i.
- R_{3i}: ratio indicating the overall occurrence of f_i.

7. Using the feature subsets of the best k ant:
   - For $j = 1$ to na,
     - Randomly produce $m - p$ feature subset for ant j, to be used in the next iteration, and store it in S_j.
8. If the number of iterations is less than the maximum number of iterations, goto step 3.

The rationale behind Eq. 4 is to update the pheromone intensities instead of accumulating pheromones. R_{1i} shows the contribution of f_i towards the best k subsets. R_{2i} indicates the degree that f_i contributes toward forming good subsets. Hence, a new subset formed by combining f_i with the other "right" features might become the best subset. The term (1 - R_{i3}) aims at favouring explor , where this term will be close to 1 if the overall usage of f_i is very low.

It is worth mentioning that there is little difference between the computational cost of the proposed algorithm and the GA-based search procedure. This is due to the fact that both of them evaluate the selected subsets using a "wrapper approach", which requires far more computational cost than evaluating the local importance of features using the "filter approach" adopted in the proposed algorithm.

5. Experimental Results

In order to evaluate the performance of the proposed algorithm, we carried out an experiment to classify speech segments according to their manner of articulation. Six classes were considered: vowel, nasal, fricative, stop, glide, and silence. We used speech signals from the TIMIT database, where segment boundaries were identified.

Three different sets of features were extracted from each speech frame: 16 log mel-filter bank (MFB), 12 linear predictive reflection coefficients (LPR), and 10 wavelet energy bands (WVT). A context dependent approach was adopted to perform the classification. So, the features used to represent each speech segment Seg_n were the average frame features over the first and second halves of segment Seg_n and the average frame features of the previous and following segments (Seg_{n-1} and Seg_{n+1} respectively). Hence, the baseline feature sets based on MFB, LPR, and WVT consist of 64, 48 and 40 features respectively.

An Artificial Neural Network (ANN) was used to classify the features of each baseline set into one of the six manner-of-articulation classes. 2000 segments were used to estimate the MSE between the classification results and the target output. The obtained MSE values for MFB, LPR and WVT were 0.1410, 0.1941 and 0.1551 respectively. It is clear that MFB achieved the lowest MSE among the three baseline sets; however, it used more features. The LPR on the other hand was outperformed by WVT despite the fact that it used more features.

The three baseline feature sets were concatenated to form a new set of 152 features. The SFS, GA and proposed ACO algorithms are used to select from these features. For the SFS method, the algorithm starts with no features and then adds one feature at a time, such that the MIEF measure (Eq. 2) is maximized. The GA-based selection is performed using the following parameter settings: population size = 30, number of generations = 25, probability of crossover = 0.8, and probability of mutation = 0.05. The obtained binary strings are constrained to have the number of '1's matching a predefined number of desired features. The MSE of an ANN trained with the 2000 speech segments is used as the fitness function. The parameters of the ACO algorithms described in the previous section are assigned the following values:

- $\eta = \kappa = 1$, which basically makes the trail intensity and local measure equally important.
- $\alpha = 0.3$, $\beta = 1.65$ and $\gamma = 3$, are found to be an appropriate choice for this and other classification tasks.
- The number of ants, *na* = 30, and the maximum number of iterations is 25, are chosen to justify the comparison with GA.
- $k = 6$. Thus, only the best *na*/5 ants are used to update the pheromone trails and affect the feature subsets of the next iteration.
- $m - p = \max(m - 5, \text{round}(0.65 \times m))$, where p is the number of the remaining features that need to be selected in each iteration. It can be seen that p will be equal to 5 if $m \geq 13$. The rational behind this is that evaluating the importance of features locally becomes less reliable as the number of selected features increases. In addition, this will reduce the computational cost especially for large values of m.
- The initial value of trail intensity $cc = 1$.
- Similar to the GA-based feature selection, the *MSE* of an ANN trained with 2000 speech segments is used to evaluate the performance of the selected subsets in each iteration.

The selected features of each method are classified using ANNs, and the obtained MSE are shown in Fig. 2.

It can be seen that the three feature selection methods were able to achieve MSE values similar to that of the baseline sets with smaller number of features, which shows the advantage of feature selection. The figure also shows that SFS achieved reasonable performance when selecting small number of features, but its performance starts to worsen as the desired number of features increases. For example, the SFS did not do much better than WVT when selecting 40 features. This is expected, as the selection process of SFS is performed by evaluating small number of subsets and selects features sequentially, which does not always lead to optimal solutions.

On the other hand, the performance of both GA and the proposed ACO was found to be much better than that of SFS. The proposed ACO algorithm was able to achieve similar or slightly better performance than GA in most of the cases. This indicates that the proposed ACO algorithm is a powerful and reliable method to search the feature subset space.

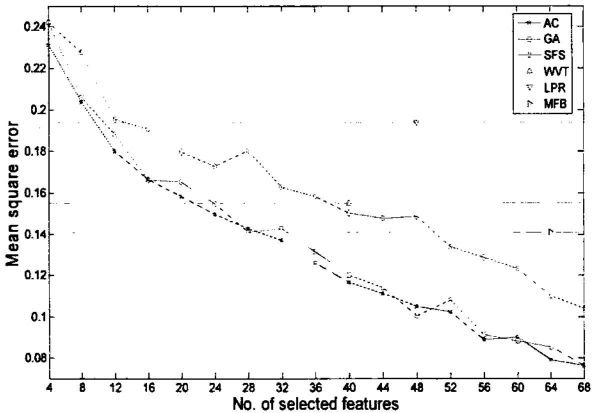

Figure 2. Mean square error of feature subsets obtained using SFS, GA and AC.

6. Conclusion

A novel feature subset search algorithm is presented. The algorithm utilizes concepts from ants' foraging and original ACO algorithm. Both local importance of features and overall performance of subsets are used to search the feature space for optimal solutions. In addition, a pheromone intensity formula is designed to reduce the chance of being trapped in local minima. When used to select features for a speech segment classification problem, the proposed algorithm outperformed both stepwise- and GA-based feature selection methods.

7. References

[1] B.D. Ripley. Pattern recognition and neural networks. Cambridge university press, 1996.

[2] M.A. Hall. Correlation-based feature selection for machine learning. PhD thesis, The University of Waikato, 1999.

[3] M. Dorigo, V. Maniezzo, and A. Colorni. "Ant System: Optimization by a colony of cooperating agents". IEEE Transactions on Systems, Man, and Cybernetics – Part B, 26:29–41, 1996.

[4] J. Kittler. "Feature set search algorithms". In C. H. Chen, editor, Pattern Recognition and Signal Processing. Sijhoff and Noordhoff, the Netherlands, 1978.

[5] P. Pudil, J. Novovicova, and J. Kittler. "Floating search methods in feature selection". Pattern Recognition Letters, 15:1119-1125, 1994.

[6] J. Yang and V. Honavar, "Feature subset selection using a genetic algorithm," IEEE Transactions on Intelligent Systems, 13: 44–49, 1998.

[7] M. Gletsos, S.G. Mougiakakou, G.K. Matsopoulos, K.S. Nikita, A.S. Nikita, and D. Kelekis. "A Computer-Aided Diagnostic System to Characterize CT Focal Liver Lesions: Design and Optimization of a Neural Network Classifier" IEEE Transactions on Information Technology in Biomedicine, 7: 153-162, 2003.

[8] I.-S. Oh, J.-S. Lee, and B.-R. Moon, "Hybrid Genetic Algorithms for Feature Selection" IEEE Transactions on Pattern Analysis and Machine Intelligence, 26: 1424-1437, 2004.

[9] M. Dorigo and T. Sttzle. Ant colony optimization. MIT press, 2004.

[10] T. Stützle and M. Dorigo. "The Ant Colony Optimization Metaheuristic: Algorithms, Applications, and Advances". In F. Glover and G. Kochenberger, editors, Handbook of Metaheuristics, Kluwer Academic Publishers, Norwell, MA, 2002.

[11] R.S. Parpinelli; H.S. Lopes; A.A. Freitas, "Data mining with an ant colony optimization algorithm", IEEE Transactions on Evolutionary Computation, 6: 321 - 332 2002.

[12] G. Di Caro and M. Dorigo. "AntNet: Distributed stigmergetic control for communications networks". Journal of Artificial Intelligence Research, 9:317-365, 1998.

[13] A. Al-Ani. Feature Subset Selection Using Ant Colony Optimization. International Journal of Computational Intelligence. 2(1), pp 53 - 58, 2005.

[14] A. Al-Ani, M. Deriche and J. Chebil. "A new mutual information based measure for feature selection", Intelligent Data Analysis, 7: 43-57, 2003.