2009 6th International Multi-Conference on Systems, Signals and Devices

Feature Selection using Ant Colony Optimization

Mohamed Deriche
Department of Electrical Engineering
King Fahd University of Petroleum and Minerals
Dhahran, 31261, Saudi Arabia
mderiche@kfupm.edu.sa

***Abstract*—The ant feature selection algorithm has recently been proposed as a new method for feature subset selection. It uses measures of both local feature importance and overall performance of subsets to search the feature space for optimal solutions. In this paper, we evaluate the effect of different local importance measures; namely the Fisher Criterion, the Mutual Information based Feature Selection, and the Mutual Information Evaluation Function.**

Keywords-Feature selection; ant systems; ant colony optimization; local measure.

I. INTRODUCTION

Feature subset selection aims at reducing the feature set dimensionality through selecting a subset of features that performs the best under certain classification systems. This is essential to reduce computational cost and improve classification performance, especially when dealing with finite sample size. One of the important aspects in designing feature selection methods is identifying proper evaluation criterion. The existing feature selection criteria can be 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 [1]. On the other hand, performance of classification algorithms is used to select features for wrapper methods [2].

Searching the feature subset space is another important aspect that has been widely investigated. This is reflected by the different search procedure methods proposed in the literature. Among those, 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 population-based algorithms is the Ant Colony Optimization algorithm (ACO) [3].

The ant algorithm was inspired by the natural ants' behaviour in their search for 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. In order to solve an optimization problem, a number of artificial ants are used to iteratively construct solutions. At every 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 upon two factors: local heuristics and previous knowledge.

The ant algorithm has recently been applied to the problem of feature selection [13]. The author chose to use a filter criterion to estimate the local importance of features, and a wrapper criterion to measure the overall performance, which provides "previous knowledge" for the future iterations.

In this paper, we evaluate the performance of a number of filter methods, which would then be used to estimate the local importance of features.

II. ANT COLONY OPTIMIZATION

Dorigo et. al. [3] adopted the concept of ants' foraging behaviour and proposed an artificial colony of ants algorithm. The algorithm was called the Ant Colony Optimization (ACO), and aimed at solving difficult combinatorial optimization problems. The ACO was originally applied to solve the classical travelling salesman problem [4], 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 [5, 6].

For the classical Travelling 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 neighbouring 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 [3], pheromone evaporation helps in avoiding rapid convergence towards a sub-optimal region.

Note that searching the feature space in the problem of feature selection is quite different from other optimization problems that researchers attempted to solve using ACO. We have recently applied ACO to the problem of feature selection with promising results [7, 8]. In the next section, we investigate a number of filter methods that can be used to estimate the local importance of features.

III. FEATURE EVALUATION MEASURES

Let the original feature set be denoted by $\mathcal{F} = \{f_1, \ldots, f_n\}$, where n is the number of features or the dimension of the feature set, and let $C = \{c_1, \ldots, c_l\}$ represents the class labels, where l is the number of classes. The objective of feature selection is to find subset $\mathcal{S} = \{f_1', \ldots, f_m'\}$, such that $m < n$, and $\mathcal{S}$ has a lower

978-1-4244-4346-8/09/$25.00 ©2009 IEEE

probability of classification error than any other subset of the same size.

The filter evaluation measures that we will be studying are the Fisher criterion, mutual information-based feature selection, and the mutual information evaluation function.

A. Fisher Criterion

The Fisher Criterion (FC) determines the correlation of individual features with the class labels [9]. The importance of each feature is measured using Eq. (1)

$$FC(f_i) = \sum_{a=1}^{l-1} \sum_{b=a+1}^{l} \pi_a \pi_b \frac{(\mu_{i,a} - \mu_{i,b})^2}{\sum_{c=1}^{l} \sigma_{i,c}^2} \quad (1)$$

- π_a and π_b are the relative mass of classes a and b
- $\mu_{i,a}$ and $\mu_{i,b}$ are the mean levels of feature f_i in classes a and b
- $\sigma_{i,c}$ is the standard deviation of feature f_i in class c

It is important to emphasize that this measure evaluates features individually and does not take into consideration the degree of correlation and interaction between different features.

B. Mutual Information-based Feature Selection

The Mutual Information (MI) measures arbitrary dependencies between random variables and thus is suitable for assessing the "information content". It has been found that maximizing the MI between transformed data and the desired target achieves a lower bound to the probability of error [10]. This was the main motivation behind developing the Mutual Information based Feature Selection (MIFS) algorithm.

The MIFS estimates the importance of features by calculating their individual MI with the class labels. Hence, the importance of feature f_i is determined by $I(C; f_i)$. In addition, the mutual information between pairs of features, $I(f_i; f_j)$, is used to avoid selecting highly dependent features. Thus, Eq. (2) is used to measure the importance of features:

$$MIFS(f_i) = I(C; f_i) - \frac{\beta}{|S|} \sum_{f_s \in S} I(f_i; f_s) \quad (2)$$

where S is the subsets of already selected features, $|S|$ is the cardinal of S, and β *is a* parameter[1] that regulates the relative importance of MI between feature f_i and each of the already selected features, s, with respect to MI between f_i and C.

C. Mutual Information Evaluation Function

Despite the fact that MIFS takes into consideration the degree of dependence between features, it does not consider the interaction between features, or how the different features work together. An experiment has been conducted, using a limited number of features and class labels, to compare the performance of all subsets of features ranked according to the values of $I(C; S_i)$ and $MIFS(S_i)$. It has been found that in the vast majority of the cases, $I(C; S_i)$, gives a very good indication about the importance of S_i, while the highest value of $MIFS(S_i)$ does not necessarily mean that S_i is better than all other subsets of the same size. This experiment reflects the importance of interaction between features. However, because in most cases the distribution of data is unknown, and using the histogram does not give a very good estimate of the distribution (especially for limited amount of data), the authors proposed the Mutual Information Evaluation Function (MIEF) [11], which is a computationally feasible function that aims at considering the interaction between features.

$MIEF(S)$ satisfies certain properties of $I(C; S)$, namely; the upper bound, lower bound and monotonicity. Feature f_i is evaluated according to the following formula:

$$MIEF(f_i) = I(C; f_i) \times \left[\frac{2}{1 + \exp(-\alpha D_i^S)} - 1 \right] \quad (3)$$

where

$$D_i^S = \min_{f_s \in S} \left[\frac{H(f_i) - I(f_i, f_s)}{H(f_i)} \right] \times \frac{1}{|S|} \sum_{f_s \in S} \left[\beta \left(\frac{I(C; \{f_i, f_s\})}{I(C; f_i) + I(C; f_s)} \right)^{\gamma} \right] \quad (4)$$

The parameters α, β, and γ are constants[2] and $H(f_i)$ is the entropy of f_i. The first term of Eq. (4) reflects the degree of dependency between f_i and the already chosen features f_s. On the other hand, the second term estimates the interaction between f_i and the already chosen features f_s. For detailed explanation of the MIEF measure, the reader is referred to [11]. It is worth mentioning that the MIEF requires only a slight increase in the computational cost compared to the traditional MIFS, due to the estimation of $I(C; \{f_i, f_s\})$.

The next section explains the ant feature selection algorithm and the use of each of the above three evaluation criteria to estimate the local importance of features.

IV. The Ant Feature Selection Algorithm

Derived concepts from ants' foraging and Dorigo's ACO algorithm have been used in the ant feature selection algorithm [8]. 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. 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. A hybrid evaluation measure is used to 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 calculated using one of the three filter measures described in the previous section (FC, MIFS or MIEF).

The following parameters are used in the algorithm:

- n: number of features that constitute the original set, $F = \{f_1, \ldots, f_n\}$.
- na: number of artificial ants to search through the feature

[1] A reasonable choice for $\beta = 0.75$

[2] $\alpha = 0.3$, $\beta = 1.65$ and $\gamma = 3$, are found to be an appropriate choice for this and other classification tasks

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) using Eq (1), (2) or (3). The Selection Measure (SM) is used for this purpose and is defined as:

$$SM_i^{S_j} = \begin{cases} \dfrac{(\mathcal{T}_i)^{\eta}\left(LI_i^{S_j}\right)^{\kappa}}{\sum_{g \notin S_j}(\mathcal{T}_g)^{\eta}\left(LI_g^{S_j}\right)^{\kappa}} & \text{if } i \notin S_j \\ 0 & \text{Otherwise} \end{cases} \quad (5)$$

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.

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 $\mathcal{S}_j$.
   - 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 $\mathcal{S}_j$, Choose feature f_i that maximizes $SM_i^{S_j}$.
       - $\mathcal{S}_j = \mathcal{S}_j \cup \{f_i\}$.
   - 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 $\mathcal{S}_j$.
   - 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. $\mathcal{PL} = [\mathcal{PL}\,;\, \mathcal{S}_j]$, 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:

$$\mathcal{T}_i = a_1 R_{1i} + a_2 R_{2i} + a_3(1 - R_{3i}) + a_4 \quad (6)$$

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.

1. 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 $\mathcal{S}_j$.
2. If the number of iterations is less than the maximum number of iterations, go to step 3.

The rationale behind Eq. (6) is to update the pheromone intensities instead of accumulating pheromones. R_{1i} represents 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 "good" features might produce the best subset. The term (1 - R_{i3}) aims at favouring exploration, where this term will be close to 1 if the overall usage of f_i is very low.

V. Experimental Results

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 feature sets based on MFB, LPR, and WVT consist of 64, 48 and 40 features respectively. Those feature sets were concatenated to form a new set of 152 features.

The local importance measures described in section III are used to implement three versions of the ant feature selection algorithm. In addition, a random selection procedure has been

adopted as a forth local importance criterion, which would provide a useful comparison basis with the other three measures. The four versions of the algorithm are used to select featured from the generated feature set. The parameters chosen for the algorithms above are:

- $\eta = \kappa = 1$, which basically makes the trail intensity and local measure equally important.
- The number of ants, $na = 30$, and the maximum number of iterations is 25.
- k=6, only the best na/5 ants are used to update pheromone trails and affect feature subsets at next iteration.
- m–p=max(m – 5, round(0.65 × 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 reduces computational cost especially for large values of m.
- The initial value of trail intensity $cc = 1$.
- The Mean Square Error (*MSE*) of an ANN trained with 2000 speech segments was used to evaluate the performance of the selected subsets in each iteration.

The selected features are classified using ANNs, and the obtained MSE values are shown in Fig. 1.

It can be seen that the three selection measures outperform the random selection, especially when the number of selected features is less than 50. When selecting more than 50 features (more than 1/3 of the whole number of features), the performance of the random selection gets closer to the other measures. Accordingly, further study on the effect of local importance when selecting large number of features needs to be conducted.

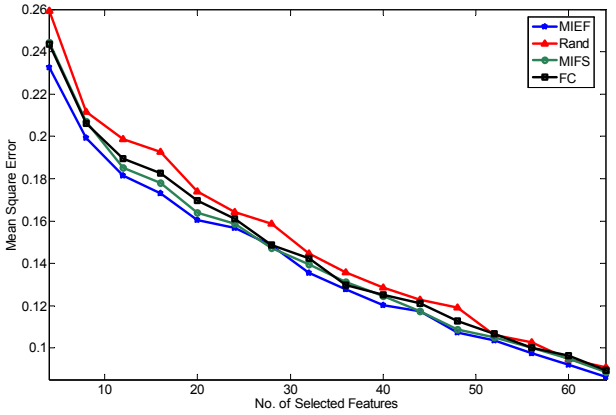

Fig. 1 MSE versus No. of selected features for 6-class problem

The MIFS is found to be better than the FC. This is expected, as the FC treats features individually. On the other hand, the figure shows that the MIEF outperforms the other two measures in almost all the cases considered. This reflects the importance of considering the interaction between features.

VI. Conclusion

We presented in this paper a study of the effect of local measure on performance of the ant feature selection algorithm. The results indicate that the local importance plays an important role, especially when selecting small number of features. The study also confirms that the MIEF evaluation measure outperforms other feature selection measures. A future study to evaluate the performance of the presented local measures when selecting large number of features will be conducted. We will also evaluate other feature selection measures, which we expect to lead to better performance.

VII. Acknowledgments

The authors wish to thank University of Technology Sydney, King Abdulaziz City for Science & Technology, and King Fahd University or Petroleum & Minerals, for the support provided to carry this research.

References

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

[2] R. Kohavi. *Wrappers for performance enhancement and oblivious decision graphs*. PhD thesis, Stanford University, 1995.

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

[4] 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.

[5] 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.

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

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

[8] A. Al-Ani. An Ant Colony Optimization Based Approach for Feature Selection. AIML'05 (to appear).

[9] J. Xuan; Y. Dong; J. Khan, E. Hoffman, R. Clarke, Y. Wang. Robust feature selection by weighted Fisher criterion for multiclass prediction in gene expression profiling. ICPR 2004, pp 291 – 294.

[10] R. Battiti, Using mutual information for selecting features in supervised neural net learning, IEEE Transactions on Neural Networks 5 (1994), 537–550.

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