International Journal of Advanced Computer Research (ISSN (print): 2249-7277 ISSN (online): 2277-7970)
Volume-3 Number-3 Issue-11 September-2013

Hybrid Meta-heuristic Pattern Classification

Rashmi G. Dukhi[1], Pratibha Mishra[2]

Abstract

Feature selection is vital in the field of pattern classification due to accuracy and processing time considerations.The selection of proper features is of greater importance when the initial feature set is considerably large. Text classification is a typical example of this situation, where the size of the initial feature set may reach to hundreds or even thousands. There are numerous research studies in the literature offering different feature selection strategies for text classification, mostly focused on filters. In spite of the extensive number of these studies, there is no significant work investigating the efficacy of a combination of features, which are selected by different selection methods, under different conditions. Proposed algorithm a new hybrid meta-heuristic approach for feature selection (ACOFS) has been presented that utilizes ant colony optimization. The main focus of this algorithm is to generate subsets of salient features of reduced size. ACOFS utilizes a hybrid search technique that combines the wrapper and filter approaches. In this regard, ACOFS modifies the standard pheromone update and heuristic information measurement rules based on the above two approaches.

Keywords

pherome; heuristic; autocatalytic (positive) feedback process; constraint-satisfaction method

1. Introduction

FEATURE selection (FS) is a commonly used step in machine learning, especially when dealing with a high dimensional space of features. The main objective of FS is to choose a subset of features from the original set of features forming patterns in a given dataset . Feature selection is extensive and it spreads throughout many fields, including text categorization, data mining , machine learning, pattern recognition, and signal processing .Recently, text categorization has become a key technology to deal with and organize a large number of documents. A major problem of text categorization is the high dimensionality of the feature space. Most of these dimensions are not relative to text categorization; even some noise data hurt the performance of the classifier. Hence, we need to select some representative features from the original feature space to reduce the dimensionality of feature space and improve the efficiency and performance of classifier . Feature selection algorithms designed with different evaluation criteria broadly fall into three categories: the filter model , the wrapper model and the hybrid model. The filter model relies on general characteristics of the data to evaluate and select feature subsets without involving any mining algorithm. The wrapper model requires one predetermined mining algorithm and uses its performance as the evaluation criterion. It searches for features better suited to the mining algorithm aiming to improve mining performance, but it also tends to be more computationally expensive than the filter model . The hybrid model attempts to take advantage of the two models by exploiting their different evaluation criteria in different search stages. The two methods are as follows:

Filter

The filter methods, or actually the scoring schemes, utilized in this study are document frequency, mutual information, chi-square, and information gain.

Document frequency

Document frequency (DF) is one of the simplest approaches to assess feature relevance in text classification problems. The DF of a specific term simply corresponds to the number of documents in a class containing that term . Hence, the DF of each term constitutes the relevancy score of the term.

Mutual information

The mutual information (MI) of 2 random variables indicates the mutual dependence of the variables. Therefore, the MI related to term *t* and class *c* describes the amount of information the presence of that term carries about the relevant class [25]. Therefore, MI can be formulated as:

MI(t,c)=log P(t|c)/P(t),

where $P(t)$ is the probability of term *t* and $P(t|c)$ is the probability of term *t* given class *c*.

Chi-square

Another popular selection approach is chi-square (CHI2). In statistics, the CHI2 test is applied to examine the independence of 2 events. The events, *X* and *Y* , are assumed to be independent if:

$$p(XY) = p(X)p(Y).$$

Information gain (IG)
It measures how much information the presence or absence of a term contributes to making the correct classification decision for any class [25]. IG reaches its maximum value if a term is an ideal indicator for class association, that is, if the term is present in a document, if and only if the document belongs to the respective class.

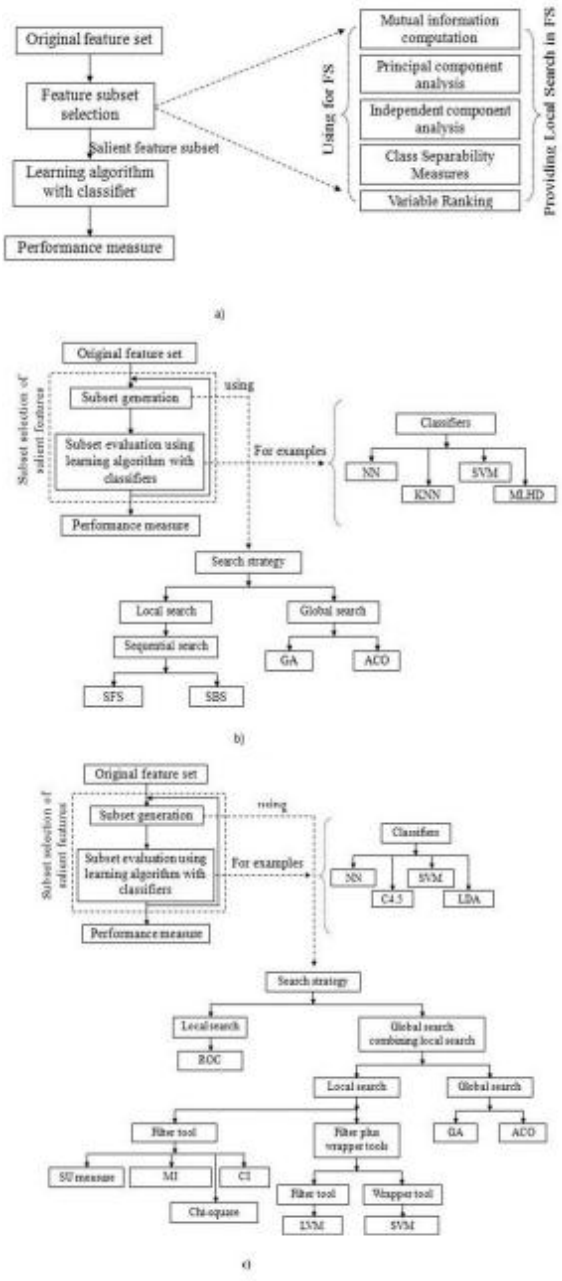

Figure 1 : a)Schematic diagram of filter approach. Each approach incorporates the specific search strategies. (b)Schematic diagram of wrapper approach. Each approach incorporates the specific search strategies and classifiers. Here, NN, KNN, SVM, and MLHD refer to the neural network, K-nearest neighbour, support vector machine, and maximum likelihood classifier, respectively. (c)Schematic diagram of hybrid approach. Each approach incorporates the specific search strategies and classifiers. Here, LDA, ROC, SU, MI, CI, and LVM, refer to the linear discriminant analysis classifier, receiver operating characteristic method, symmetrical uncertainty, mutual information, correlation information, and latent variable model, respectively

2. Ant Colony Optimization (ACO)

In the early 1990s, ant colony optimization (ACO) was introduced by M. Dorigo and colleagues as a novel nature inspired meta-heuristic for the solution of hard combinatorial optimization (CO) problems. ACO belongs to the class of meta-heuristics, which includes approximate algorithms used to obtain good enough solutions to hard CO problems in a reasonable amount of computation time. The inspiring source of ACO is the foraging behavior of real ants [24].

The first ACO algorithm developed was the ant system (AS) [25] and since then several improvement of the AS have been devised . The ACO algorithm is based on a computational paradigm inspired by real ant colonies and the way they function. The underlying idea was to use several constructive computational agents (simulating real ants). A dynamic memory structure incorporating information on the effectiveness of previous choices based on the obtained results, guides the construction process of each agent. The behavior of each single agent is therefore inspired by the behavior of real ants [24].

The paradigm is based on the observation made by ethologists about the medium used by ants to communicate information regarding shortest paths to food by means of pheromone trails. A moving ant lays some pheromone on the ground, thus making a path by a trail of this substance.While an isolated ant moves practically at random,exploration, an ant encountering a previously laid trail can detect it and decide with high probability to follow it, exploitation, and consequently reinforces the trail with its own

pheromone. What emerges is a form of autocatalytic process through which the more the ants follow a trail, the more attractive that trail becomes to be followed. The process is thus characterized by a positive feedback loop, during which the probability of choosing a path increases with the number of ants that previously chose the same path.The mechanism above is the inspiration for the algorithms of the ACO family [24].

ACO algorithms can be applied to optimization problems, for which the following problem-dependent aspects can be defined [1], [2]:

1. An appropriate graph representation to represent the discrete search space. The graph should accurately represent all states and transitions between states. A solution representation scheme also has to be defined.
2. Heuristic desirability of links the representation graph.
3. An autocatalytic (positive) feedback process; that is, a mechanism to update pheromone concentrations such that current successes positively influence feature solution construction.
4. A constraint-satisfaction method to ensure that only feasible solutions are constructed.
5. A solution construction method which defines the way in which solutions are built and a state transition probability.

3. Ant Colony Optimization For Feature Selection

Feature selection is one of the applications of subset problems (SSP). Given a feature set of size n, the FS problem is to find a minimal feature subset of size s ($s < n$) while retaining a suitably high accuracy in representing the original features. Therefore, there is no concept of path. A partial solution does not define any ordering among the components of the solution, and the next component to be selected is not necessarily influenced by the last component added to the partial solution [25]. Furthermore, solutions to an FS problem are not necessarily of the same size. To apply an ACO algorithm to solve a feature selection problem, these aspects need to be addressed. The first problem is addressed by redefining the way that the representation graph is used.

Graph Representation

The feature selection problem may be reformulated into an ACO-suitable problem. ACO requires a problem to be represented as a graph. Here nodes represent features, with the edges between them denoting the choice of the next feature. The search for the optimal feature subset is then an ant traversal through the graph where a minimum number of nodes are visited that satisfies the traversal stopping criterion. Figure 1 illustrates this setup. Nodes are fully connected to allow any feature to be selected next. The ant is currently at node *f*1 and has a choice of which feature to add next to its path (dotted lines). It chooses feature *f*2 next based on the transition rule, then *f*3 and then *f*4. Upon arrival at *f*4, the current subset {*f*1, *f*2, *f*3, *f*4} is determined to satisfy the traversal-stopping criterion (e.g. suitably high classification accuracy has been achieved with this subset). The ant terminates its traversal and outputs this feature subset as a candidate for data reduction [1].

Based on this reformulation of the graph representation, the transition rules and pheromone update rules of standard ACO algorithms can be applied. In this case, pheromone and heuristic value are not associated with links. Instead, each feature has its own pheromone value and heuristic value.

Heuristic Desirability

The basic ingredient of any ACO algorithm is a constructive heuristic for probabilistically constructing solutions [24]. A constructive heuristic assembles solutions as sequences of elements from the finite set of solution components. A solution construction starts with an empty partial solution. Then, at each construction step, the current partial solution is extended by adding a feasible solution component from the set of solution components. A suitable heuristic desirability of traversing between features could be any subset evaluation function for example, an entropybased measure or rough set dependency measure [13]. In proposed algorithm classifier, performance is mentioned as heuristic desirability for feature selection. The heuristic desirability of traversal and node pheromone levels are combined to form the so-called probabilistic transition rule, denoting the probability that ant *k* will include feature *i* in its solution at time step *t*:

$$P_i^k(t) = \{ [\tau_i(t)]^{\alpha} . [\eta_i]^{\beta} / \Sigma_{u \in J^k} [\tau_u(t)]^{\alpha} . [\eta_u]^{\beta} \text{ if } i \in J^k$$
$$0 \text{ otherwise}$$

Where J^k is the set of feasible features that can be added to the partial solution; τ_i and η_i are respectively the pheromone value and heuristic desirability associated with feature *i*. α and β are two parameters that determine the relative importance of the pheromone value and heuristic information. The transition probability used by ACO is a balance between pheromone intensity (i.e. history of previous successful moves), τ_i, and heuristic information (expressing desirability of the move), η_i. This effectively balances the exploitation–exploration trade-off. The search process favors actions that it has found in the past and which proved to be effective, thereby exploiting knowledge obtained about the search space. On the other hand, in order to discover such actions, the search has to investigate previously unseen actions, thereby exploring the search space. The best balance between exploitation and exploration is achieved through proper selection of the parameters α and β. If α = 0, no pheromone information is used, i.e. previous search experience is neglected. The search then degrades to a stochastic greedy search.

4. Experimental Results

Tables 1 shows the results of ACOFS over 20 independent runs on nine real-world benchmark classification datasets. The classification accuracy (CA) refers to the percentage of exact classifications produced by trained NNs on the testing set of a classification dataset. In addition, the weights of features for the above nine datasets over 20 independent runs are exhibited. On the other hand, shows how the best solution was selected in ACOFS for the glass dataset. In order to observe whether the internal process of FS in ACOFS is appropriately being performed.

Table 1: Performance of ACOFS

Dataset	Avg. result with all features				Avg. result with selected features			
	n	SD	CA (%)	SD	n_s	SD	CA(%)	SD
Cancer	9.00	0.00	97.97	0.42	3.50	1.36	98.91	0.40
Glass	9.00	0.00	76.60	2.55	3.30	1.14	82.54	1.44
Vehicle	18.00	0.00	60.71	11.76	2.90	1.37	75.90	0.64
Thyroid	21.0	0.00	98.04	0.58	3.00	1.34	99.08	0.11
Ionosphere	34.0	0.00	97.67	1.04	4.15	2.53	99.88	0.34
Credit card	51.0	0.00	85.23	0.67	5.85	1.76	87.99	0.38
Sonar	60.0	0.00	76.82	6.97	6.25	3.03	86.05	2.26
Gene	120.0	0.00	78.97	5.51	7.25	2.53	89.20	2.46
Colon cancer	100.0	0.00	59.06	5.75	5.25	2.48	84.06	3.68

As can be seen from Table 3, ACOFS was able to select a smaller number of features for solving different datasets. For example, ACOFS selected, on average, 3.00 features from a set of 21 features in solving the thyroid dataset. It also selected, on average, 7.25 genes (features) from a set of 120 genes in solving the gene dataset.

On the other hand, a very large-dimensional dataset, that of colon cancer, was preprocessed from the original one to be utilized in ACOFS. In this manner, the original 2000 features of colon cancer were reduced to within 100 features. ACOFS then obtained a small number of salient genes, 5.25 on average, from the set of 100 genes for solving the colon cancer dataset. In fact, ACOFS selected a small number of features for all other datasets having more features. Feature reduction in such datasets was several orders of magnitude (see Table 3).

The positive effect of a small number of selected features (ns) is clearly visible when we observe the CA. For example, for the vehicle dataset, the average CA of all features was 60.71%, whereas it had been 75.90% with 2.90 features. Similarly, ACOFS an average CA of 86.05% with the average number of features of 6.25 substantially reduced for the sonar dataset, while the average CA had been 76.82% with all 60 features. Other similar types of scenarios can also be seen for all remaining datasets in ACOFS. Thus, it can be said that ACOFS has a powerful searching capability for providing high-quality solutions. CA improvement for such datasets was several orders of magnitude . Furthermore, the use of ns caused a relatively small standard deviation (SD), as presented in Table 3 for each entry. The low SDs imply robustness of ACOFS. Robustness is represented by consistency of an algorithm under different initial conditions.

References

[1] M.H. Aghdam, N. Ghasem-Aghaee, and M.E. Basiri, "Application of ant colony optimization for feature selection in text categorization" IEEE Congress on Evolutionary Computation (CEC 2008), Hong Kong, 2008, pp. 2872-2878.

[2] M. H. Aghdam, N. Gsem-Aghaee, M. E. Basiri, "Text feature selection using ant colony optimization". Expert Systems with Applications, Expert Systems with Applications 36, 2009, pp. 6843–6853.

[3] M.E. Basiri, N. Ghasem-Aghaee, and M.H. Aghdam, Using Ant Colony Optimization-Based Selected Features for Predicting Post-Synaptic

Activity in Proteins, EvoBIO 2008, LNCS 4973, 2008, pp.12–23.
[4] M.E. Basiri, N. Ghasem-Aghaee, Protein Function Prediction Using ACO And Bayesian Networks, Proceedings of BIOMA 2008, The 3rd International Conference on Bioinspired Optimization Methods and their Applications, Ljubljana, Slovenia, 2008, pp. 147-157.
[5] S. Nemati, R. Boostani, and M. D. Jazi, "A Novel Text-Independent Speaker Verification System using Ant Colony Optimization Algorithm". ICISP2008, LNCS 5099, Springer-Verlag, Berlin Heidelberg, France, 2008, pp. 421-429.
[6] W. Shang, H. Huang, H. Zhu, Y. Lin, Y. Qu, and Z. Wang, "A novel feature selection algorithm for text categorization," Expert Systems with Applications, vol. 33, no. 1, July 2007, pp. 1–5.
[7] E. Bonabeau, M. Dorigo, and G. Theraulaz, Swarm Intelligence: FromNatural to Artificial Systems. Oxford University Press, New York,1999.
[8] B. Liu, H.A. Abbass, and B. McKay, "Classification Rule Discovery with Ant Colony Optimization," IEEE Computational Intelligence Bulletin, vol.3, no. 1, February 2004, pp. 31–35.
[9] M. Dorigo and G.D. Caro, "Ant Colony Optimization: A New Metaheuristic," in Proc. of the Congress on Evolutionary Computing, 1999.
[10] M. Srinivas, and L. M. Patnik, "Genetic Algorithms: A Survey", IEEE Computer Society Press, Los lamitos, 1994.
[11] W. Siedlecki, and J. Sklansky, "On automatic feature selection",International Journal of Pattern Recognition and Artificial Intelligence, 2(2), 1998, pp. 197–220.
[12] M.F. Caropreso and S. Matwin, Beyond the Bag of Words: A Text Representation for Sentence Selection. Berlin: Springer-Verlag, 2006, pp. 324–335.
[13] R. Jensen, "Combining rough and fuzzy sets for feature selection,"Ph.D. dissertation, School of Information, Edinburgh Univ., 2005.
[14] R.O. Duda and P.E. Hart, Pattern Classification and Scene Analysis. John Wiley & Sons, Chichester, 1973.
[15] G. Forman, "An extensive empirical study of feature selection metrics for text classification," Journal of Machine Learning Research 3, pp. 1289–1305, 2003.
[16] D. Mladeni, Feature Selection for Dimensionality Reduction. Berlin: Springer-Verlag, 2006, ch.5.
[17] W. Siedlecki and J. Sklansky, A note on genetic algorithms for largescale feature selection. Pattern Recognition Letters, vol. 10, no. 5,335–347, November 1989.
[18] A.A. Ani, "Ant Colony Optimization for Feature Subset Selection," Trans. Engineering, Computing and Technology, vol. 4, pp. 35–38, February 2005.
[19] C.K. Zhang and H. Hu, "Feature Selection Using the Hybrid of Ant Colony Optimization and Mutual Information for the Forecaster," in Proc. 4th International Conf. Machine Learning and Cybernetics, vol. 3, pp. 1728–1732, Aug 2005.
[20] H.K. Rashidy, K. Faez, and S.M. Taheri, Feature Selection Using Ant Colony Optimization (ACO): A New Method and Comparative Study in the Application of Face Recognition System, ICDM, Berlin: Springer-Verlag, pp. 63–73, 2007.
[21] L. Kml and J. Kittler, "Feature set search algorithms," in Proc. Chen, C.H. (ed.) Pattern Recognition and Signal Processing, Sijhoff and Noordhoff, the Netherlands, 1978.
[22] J. Bins, "Feature Selection from Huge Feature Sets in the Context of Computer Vision", Ph.D. dissertation, Dept. Computer Science, Colorado State Univ., 2000.
[23] A. Sheta, and H. Turabieh, "A comparison between genetic algorithms and sequential quadratic programming in solving constrained optimization problems", ICGST Internatioanl Journal on Artificial Intelligence and Machine Learning (AIML), 6(1), 2006, pp.67–74.
[24] M. Dorigo and C. Blum, "Ant colony optimization theory: A survey," Theoretical Computer Science 344, pp. 243–278, 2005.
[25] C. Blum, A. Roli, and M. Dorigo, "HC-ACO: The Hyper-Cube Framework for Ant Colony Optimization," in Proc. 4th Metaheuristic International Conference, pp. 399–403, 2001.
[26] G. Leguizamon and Z. Michalewicz, "A New Version of Ant System for Subset Problems," in Proc. IEEE Congress on Evolutionary Computation, vol. 2, pp. 1465, July 1999.

Rashmi Dukhi received the M.C.A degree from Nagpur University,India in 2002. She is currently persuing Mtech(CSE)-III Semester from Nagpur University.She is a lecturer in the Department of Computer Application,GHRIIT,Nagpur. Her research interests include database, data mining, algorithms, text mining and genetic algorithms