Integrating Swarm Intelligence and Statistical Data for Feature Selection in Text Categorization

M. Janaki Meena
Department of CSE
PSG College of Technology
Coimbatore, Tamilnadu

K.R. Chandran
Department of IT
PSG College of Technology
Coimbatore, Tamilnadu

J. Mary Brinda
Student, Department of CSE
PSG College of Technology
Coimbatore, Tamilnadu

ABSTRACT

Feature selection is the principal step in classification problems with attributes of high dimension. It may also be considered as a problem to determine the subset of terms in training corpus, which maximizes the classifier's performance. Most of the machine learning algorithms has tainted performance in high dimensional feature space. In this paper, a novel feature selection method based on Ant Colony Optimization, a swarm intelligence algorithm is proposed. Ant Colony Optimization is a metaheuristic algorithm used to increase the ability of finding high quality solutions to NP-hard problems. The heuristic information required for the optimization process is obtained through a chi-square based statistical method, CHIR which results in fast convergence. Performance of the classifier with features selected by proposed method is compared to the feature selected by conventional chi-square and CHIR methods. It is found that the proposed algorithm identifies better feature set than the conventional methods.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *Information Filtering;* 1.5.4 [Pattern Recognition]: Applications – Text processing.

General Terms

Algorithms, Design, Experimentation and Performance

Keywords

Machine learning, feature selection, Ant colony optimization, chi-square method.

1. INTRODUCTION

Text Categorization (TC) is the task of assigning a given text document to one or more predefined categories based upon the contents of the document. It is one of the most important ways to organize and manage information. It is enduring to be one of the most researched problems due to continuously-increasing amount of electronic documents and digital data. This task requires knowledge about the contents of the documents and the role of the categories. When the corpus contains a large collection of documents, it is infeasible to perform the classification manually. For the past two decades machine learning has become the primary approach and many numbers of machine learning, knowledge engineering, associative rule learning and probabilistic-based methods have been proposed for text classification [16], [6], [13].

Many features in the corpus could be irrelevant and noisy, hence feature selection is performed to reduce the number of features and avoid overfitting [3], [4], [11]. Feature selection is the process of identifying an optimal feature subset measured by an evaluation criterion. As the dimensionality of the feature space expands, the complexity of finding optimal subset becomes very high [8], [10], [12]. Many problems related to feature selection is usually intractable and shown to be NP-hard. Traditional feature selection algorithms for feature selection are either supervised or unsupervised, depending on whether the category labels are available for the documents in the training corpus. Unsupervised selection methods include document frequency and term strength and can be easily applied to clustering and classification tasks whereas supervised feature selection methods using information gain and χ^2 statistics improve the classifiers performance [3], [4], [5], [8].

Ant Colony Optimization is a new metaheuristic approach to stochastic combinatorial optimization. Characteristics of this model include positive feedback, distributed computation and constructive greedy heuristic. In this model, subset selection problem is reformulated as a search problem in a connected graph. Ant Colony Optimization is inspired by the observation of real ants in their search for shortest paths to food source. The ants communicate through a chemical aromatic substance called as pheromones [1], [15].

The rest of this paper is organized as follows: Section 2, present a brief overview of the methodologies used in the proposed algorithm Section 3 describes the proposed algorithm and Section 4 discuss about the experimentations carried out and results.

2. BACKGROUND KNOWLEDGE

In this section, the general procedure of feature selection, the chi-square modified CHIR algorithm for feature selection and Ant colony optimization algorithm are explained.

2.1 General Procedure for Feature Selection

Terms in the training corpus are the nodes in the search space and search is made to determine the optimal subset. The procedure of feature selection consists of four steps subset generation, subset evaluation, stopping criteria and result validation [3], [8].

Subset Generation

Subset generation is a process of heuristic search in the search space, to identify a subset S. The disposition of this process is determined by two issues: the starting point of the search procedure and search strategy. Search procedure may start with an empty set and add features consecutively (forward approach) or start with all the features in the subset and eliminate one at a time in each step (backward approach) or start with an empty set and a full set, add and remove feature simultaneously in each step. Search may also begin with a randomly selected subset in order to avoid getting trapped in the local optima. Existing search strategies include complete search, sequential search and random search [3]. Generally exhaustive search is a complete search, in which every possible subset is analyzed, but the complexity of the search is exponential to the cardinality of feature space. Sequential search algorithms include pied greedy search algorithm, such as sequential forward selection, sequential backward selection, and bidirectional selection. Sequential search gives up completeness, thus work at risk to lose the optimal solutions. Random search starts with randomly selected subsets and may be proceed in a sequential way as random-restart hill climbing algorithm or generate next subset in a completely random manner. Randomness helps these methods to escape the local optima [3].

Subset Evaluation

Each newly generated subset is evaluated by a classifier dependent or independent evaluation criterion. Classifier independent algorithms are used in filter model and the goodness of the selected subset is exploited by the intrinsic characteristics of the training data. Some popular independent evaluation criteria are distance measures, dependency measures, and consistency measures. A dependant criterion used in the wrapper model requires a predetermined mining algorithm, the performance of the algorithm is considered as the evaluation criteria. Performance of wrapper model is relatively high when compared to filter model, but the complexity of the model is very high [3], [8].

Stopping Criteria

Stopping criteria of the algorithm determines when the algorithm should stop. The conditions may be given as number of iteration, subsequent addition or deletion of features does not improve the performance or a minimum required performance is reached.

Result Validation

In real world applications, the selected features are evaluated by monitoring the changes in the mining algorithms performance then the features change.

2.2 CHIR Algorithm

CHIR is a supervised learning algorithm based on χ^2 statistics, which determines the dependency between a term and a category and also the type of dependency. Type of dependency indicates whether the feature is a positive or negative feature for the category. A feature f is positive for a category c, when its occurrence in a document d, increases the probability of the document to be in c. To evaluate the dependency of a term w, to category c, $R_{w,c}$ is defined in CHIR as [1]:

$$R_{w,c} = \frac{O(w,c)}{E(w,c)} \tag{1}$$

If there is no dependency between the term w and the category c, then the value of $R_{w,c}$ is close to 1. If there is a positive dependency then the observed frequency is larger than the expected frequency, hence value of $R_{w,c}$ is larger than 1 and when there is a negative dependency $R_{w,c}$ is smaller than 1.

Based on χ^2 statistics and $R_{w,c}$ a new definition for term-goodness for a corpus with m classes is given in CHIR algorithm as [1]:

$$r\chi^2(w) = \sum_{j=1}^{m} p(R_{w,cj})\chi^2_{w,cj} \text{ with } R_{w,c_j} > 1 \tag{2}$$

Where $p(R_{w,c_j})$ is the weight of $\chi^2_{w,cj}$ in the corpus. In terms of R_{w,c_j}, $p(R_{w,c_j})$ is defined as:

$$p(R_{w,c_j}) = \frac{R_{w,c_j}}{\sum_{j=1}^{m} R_{w,c_j}} \text{ with } R_{w,c_j} > 1 \tag{3}$$

Larger value of $r\chi^2(w)$ indicates that the term w is more relevant to the category.

2.3 Ant Colony Optimization

Ant Colony Optimization is a metaheuristic algorithm that guides and modifies other heuristics to produce solutions beyond those that are normally generated in a quest for local optimality [1]. Identifying the optimal feature subset is a NP-hard problem; in worst case exact algorithms need exponential time to find the optimal solution, hence approximate algorithms also called as heuristic methods obtain good, which are near-optimal solutions at relatively low computational cost.

Ant Colony Optimization algorithm begins and proceeds randomly, at each point of choice it randomly choose ‘n’ points and analyze their probabilistic value. Node with the highest probability value is chosen as next point. The probabilistic value for each point in the search space is determined, based on its heuristic information and pheromone value [1],[14].

Ant colony optimization for feature selection regard, terms in the training corpus as nodes in the feature space. Each of them posseses an initial pheromone value and a heuristic information according to the importance of the term to the category in which it has occurred. The algorithm randomly starts with a feature, and then chooses ‘b’, (b is the branching factor at each point) features randomly and determines their probabilistic value using (4)

$$P_i^k(t) = \begin{cases} \dfrac{[\tau_i(t)]^\alpha [\eta_i]^\beta}{\sum_{u \in S} [\tau_u(t)]^\alpha [\eta_u]^\beta} & \text{if i} \in \text{S} \\ 0 & \text{Otherwise} \end{cases} \quad (4)$$

Where τ_i denotes the pheromone value and η_i denotes heuristic information for the feature i, S is the set of features chosen randomly. α and β determines the importance of the pheromone value and heuristic information respectively. The feature with the highest probabilistic value is included in the constructed feature subset. The probabilistic rule for the problem should balance the exploration and exploitation of the search. Exploration refers to the tendency of the algorithm to analyze the unvisited nodes and exploitation refers to the utilization of the search experience gathered from previous iterations. Using heuristic information is important since it gives the possibility of exploiting, problem specific knowledge [1]. Generally, two types of heuristic information static and dynamic are used. In static method, heuristic information is computed once at initialization and remains unchanged throughout the whole algorithms run. Dynamic method, changes the heuristic information according to the partial solution constructed and computed at each step of the ant's walk [1].

Pheromone contents of all the paths decrease with time, to chance on the evaporation rate and the pheromone contents of the paths taken by best performing ants are increased. Hence the pheromone update rule is as in (3)

$$\tau_i(t+1) = (1-\rho)\tau_i(t) + \Delta\tau_i^{best}(t) \quad (5)$$

Where $\tau_i(t)$ denotes the pheromone value at the i[th] iteration, $\rho \in (0,1)$ denotes the evaporation rate. The role evaporation is to avoid stagnation that is the situation in which all the ants constructs the same solution. The quantity of pheromone that each ant has to deposit is determined, by the quality of the chosen features. The quality of the selected features may be determined using the performance of the classifier.

$$\Delta\tau_i^k(t) = \begin{cases} \phi.\left(\gamma(S^k(t)) + \dfrac{\varphi(n - |S^k|)}{n}\right) & \text{if i} \in S^k(t) \\ 0 & \text{Otherwise} \end{cases} \quad (6)$$

The equation to determine the deposit of pheromone is as in (6), Where $\gamma(S^k(t))$ is the performance of the classifier. ϕ and φ are the two parameters that control the relative weight of the classifier performance and length of the feature subset.

3. Proposed ACO for Feature Selection in TC

In the feature selection algorithm, all the terms in the training corpus are treated as nodes; each node is associated with an initial heuristic information and pheromone value. Static heuristic values are assigned to the terms; the heuristic information reveals the importance of the term to the category in which it occurred in the training documents. Each ant starts with an empty subset and start at a random point.

The proposed algorithm involves design of heuristic function for the terms in the training corpus, update function for pheromone value of the terms, and the list of features whose pheromone value is updated. After every iteration local search may be applied to some of the features selected by the iteration best ant.

3.1 Heuristic Function

Heuristic information for each term w in the feature space gives the approximate dependency of w to the category c of the document d in which w occurred. The new measure $R_{w,c}$, defined in the CHIR algorithm is used to determine the dependency between the word or term and the category. $R_{w,c}$ defined in (1) is rewritten by expanding E(w,c), the expected frequency of the term w in the category c in (7):

$$R_{w,c} = \frac{O(w,c)n}{((O(w,c)+O(\neg w,c))(O(w,c)+O(w,\neg c)} \quad (7)$$

Where $O(w,c)$, denotes the observed frequency of the term w in the category c, $O(w,\neg c)$ denotes the observed frequency of the terms occurred in documents that do not belong to the category c, $O(\neg w,c)$ denotes the number of documents in the category c without the word w and 'n' is the total number of documents in the training corpus. It may be observed that 'n', the total number of documents in the training corpus and $O(w,c)+O(\neg w,c)$, which is the number of documents in the category c is constant for all the terms. Hence they may be replaced with a constant in (7) and rewritten as (8):

$$R_{w,c} = \frac{O(w,c)}{O(w,c)+O(w,\neg c)} \times C_0 \quad (8)$$

Where $C_0 = \dfrac{n}{|d_c|}$ is a constant for all the terms. (8) is rewritten as in (9) for better understanding.

$$R_{w,c} = \frac{1}{1+\dfrac{O(w,\neg c)}{O(w,c)}} \times C_0 \quad (9)$$

From (9), it may be observed that $R_{w,c}$ reveals the type and strength of dependency between the term and the category. When a word is a positive feature for a category c, it occurs more frequently in category c than in other categories, that is the value of $O(w,c)$ is greater than $O(w,\neg c)$ and ultimately $R_{w,c}$ is greater than 1. The strength of dependency of a positive term w to category c is proportional to the distance of $R_{w,c}$ from 1. When a word is a negative feature, it occurs rarely in the category c and the value of $O(w,c)$ is lesser than $O(w,\neg c)$. The strength of dependency of a negative term w to category c is inversely proportional to the distance of $R_{w,c}$ from 1. Hence when the value of $R_{w,c}$ is greater than 1, it is directly used as the heuristic value and inverse of $R_{w,c}$ is used otherwise.

3.2 Update Pheromone

Pheromones values of the selected terms are updated after estimating the performance of the classifier with the selected features. The features selected by the ants are given as input to the classifier and the root mean square value of the error is calculated.

Update function deposits pheromone inversely proportional to the mean square error of the classifier.

Pheromone update is done for the features selected by the iteration best ant global best ant, when high exploration of search space is required and the convergence time is more for this model. Pheromone update is done only for features selected by the global best ant while fast convergence is required.

Table I

Proposed ACO Algorithm for Feature Selection in TC

Input: D – training set of documents of size N_d;
W – Set of distinct words in D
C – Set of document categories in D of size N_c
τ_0 - Initial pheromone value
b – branching factor
n_c - number of features required in each category
N_{ants} – number of ants
N_i – number of Iterations

Output: Selected Features

Procedure ACO_for_Feature_Selection

1: For all $w_i \in$ W do
2: Initialize W_i.pheromone = τ_0
3: $R_{w,c} \leftarrow$ Estimate_$R_{w,c}$( )
4: If $R_{w,c} > 1$, then
5: W_i.heuristic_Value $\leftarrow R_{w,c}$
6: Else
7: W_i.heuristic_Value $\leftarrow \frac{1}{R_{w,c}}$
8: End If
9: End For
10: For iter = 1 to N_i
11: Generate N_{ant} ants
12: For ant_1 to ant_{Nant}
13: $L_{cp} \leftarrow$ List of randomly generated b features
14: For all $L_{cpi} \in L_{cp}$
15: p $\leftarrow$ Apply probabilistic rule
16: End For
17: $F_k \leftarrow$ Feature with highest p value in L_{cp}
18: While F_k is not added to ant_j.F
19: If C_a is the category of F_k then
20: If $ant_j.|LF_{ca}| < n_c$ then // number of features chosen in // category C_a by j^{th} ant is less than n_c
21: $ant_j.F \leftarrow ant_j.F \cup F_k$
22: End If
23: Else
24: $F_k \leftarrow$ Feature with next highest p value in L_{cp}
25: End If
26: End While
27: End For
28: Identify_Iteration_Best_Ant()
29: Iteration_Best_Ant.Local Search() // Optional
30: Identify_Global_Best_Ant()
31: Iteration_Best_Ant.Update_Pheromone() //Optional
32: Global_Best_Ant.Update_Pheromone()
33: End For
34: Return feature set built by global best ant

3.3 Local Search

Local search is an optional way to optimize the iteration best ant. In local search, top feature of the selected feature set is replaced by features with nearly equivalent heuristic value of the feature's category.

Table II

Algorithm for Local Search

Input: D – training set of documents of size N_d;
C – Set of document categories in D of size N_c
$T[C_i]$ – Set of terms in the category C_i sorted in ascending order of heuristic information
F – Features selected by Iteration_Best_Ant
F_r - Root Mean Square Error of F

Output: Modified Feature Set

Procedure Local_Search

1: For $F_{max} \in$ F do // Feature with maximum heuristic information
2: $C_j \leftarrow$ Identify_Category(F_{max})
3: p $\leftarrow$ position of F_{max} in $T[C_j]$
4: $F_{c1} \leftarrow F - F_{max} \cup T[C_j][p-1]$
5: $F_{c2} \leftarrow F - F_{max} \cup T[C_j][p+1]$
6: Run_Classifier(F_{c1})
7: $F_{r1} \leftarrow$ Root Mean Square Error of F_{c1}
8: Run_Classifier(F_{c2})
9: $F_{r2} \leftarrow$ Root Mean Square Error of F_{c2}
10: Return feature set with minimum_RMSE
11: End

4. Experimental Results and Discussion

The text categorization approach proposed in this paper has been implemented and evaluated with extensive experimentations on the six categories alt.atheism, comp.graphics, comp.os.ms-windows.misc, comp.sys.ibm.pc.hardware, comp.sys.mac.hardware and comp.windows.x of 20 newsgroup benchmarks. Out of the six categories, five are related to computer

science and other one is entirely different topic. It is difficult to determine the features for these types of data sets.

4.1 Experimental Setup

The issues to be taken care in the experimental setup of ACO algorithm are value of α, β, ρ, ϕ and φ. In our experiments, equal importance was given for the heuristic information as well as the pheromone concentration on nodes hence both α and β are set to 0.5. Evaporation rate ρ is set to 0.8 and ϕ =0.8 and φ =0.2. Experiments were conducted with 10 trials of randomly chosen 10% of the documents; this is the case when there are fewer documents as example. The algorithm was tested with 500 and 1000 ants. The algorithm was tested by running for 50 iterations with three different cases: pheromone contents updated for features selected by iteration and global best ants, pheromone contents updated for features selected by global best ants, and pheromone contents updated for features selected by iteration and global best ants local search applied after each iteration.

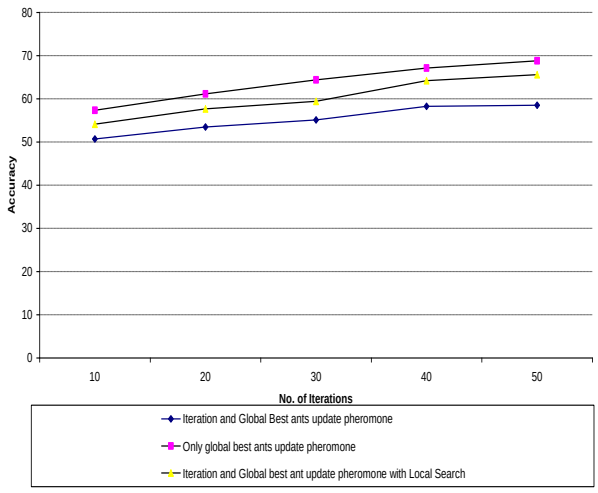

Fig 1: Comparsion of algorithms

4.2 Performance of the proposed algorithm

Performance of the classifier with features selected by the proposed algorithm is compared to the features selected by conventional CHI and CHIR algorithms. The feature subset selected by the proposed algorithm in 10 iterations has classification accuracy equivalent to the features selected by CHI algorithm. As the number of iterations increase the classification accuracy also increases and in 30 iterations it could find a feature subset with classification accuracy equivalent to the one selected by CHIR algorithm. After 50 iterations the algorithm could find a better solution than both of the compared algorithms.

The algorithm has a fast convergence when pheromone of terms selected by only global ants were updated. And local search also improves the performance significantly.

5. Conclusion

Identifying optimal feature subset for text categorization is an NP-hard problem, and becomes more difficult when the number of training examples is less. In this paper, a hybrid metaheuristic algorithm is proposed and its performance is compared with the existing feature selection algorithms. Two variations are also made in the proposed algorithm and compared. In future, analysis could be made to further improve classification accuracy with fewer training documents.

6. REFERENCES

[1] Marico Dorgio and Thomas Stutzle 2005. Ant Colony Optimization. MIT Press.

[2] Yanjun Li, Congnan Luo, and Soon M. Chung Text Clustering with Feature Selection by using Statistical Data, IEEE Transactions on Knowledge and Data Engineering, Vol., XX, May 2008, 641-652.

[3] Huan Liu and Lei Yu. Towards Integrating Feature Selection Algorithms for Classification and Clustering, IEEE Transactions on Knowledge and Data Engineering, Vol., 17, No. 4, April 2005, 491-502.

[4] Elena Montanes, Irene Diaz, Jose Ranilla, Elias F. Combarro, and Javier Fernandez. Scoring and Selecting Terms for Text Categorization, IEEE Intelligent Systems, 2005.

[5] Elias F. Combarro, Elena Montanes, Irene Diaz, Jose Ranilla, and Ricardo Mones. Introducing A Family Of Linear Measures For Feature Selection In Text Categorization, IEEE Transactions on Knowledge and Data Engineering, Vol., 17, No. 9, 2005, 1223-1232.

[6] Baoli Li, Neha Sugandh, Ernest V. Garcia, Ashwin Ram. Adapting Associative Classification to Text Categorization, ACM Symposiun on Document Engineering, Winnipeg, Canada, August 28-31, 2007.

[7] Xiao-Bing Xue and Zhi-Hua Zhou. Distributional Features for Text Categorization, IEEE Transactions on Knowledge and Data Engineering, Vol., 21, No. 3 2009, 428-442.

[8] M. Dash, H. Liu. Feature Selection for Classification, Intelligent Data Analysis 1, 1997, 131-156.

[9] Tao Liu, Shengping Liu, Zheng Chen, and Wei Ying Ma. An Evaluation On Feature Selection For Text Clustering, Proceedings of the twentieth International Conference on Machine Learning, Washington DC 2003.

[10] Ciya Liao, Shamim Alpha and Paul Dixon. Feature Preparation in Text categorization, Oracle Corporation.

[11] Yiming Yang, Jan O. Pedersen. A Comparative Study On Feature Selection In Text Categorization., in Proc. 1997. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.32.9956.

[12] George Formann. Feature Selection we've Barely scratched the surface, Hewlett Packard Laboratories, Palo Alto, 2007.

[13] Forman, G. 2003. An extensive empirical study of feature selection metrics for text classification. J. Mach. Learn. Res. 3 (Mar. 2003), 1289-1305.

[14] Ahmed Al-Ani. Ant Colony Optimization for Feature Subset Selection, Proceedings of World Academy of Science, Engineering and Technology, Vol. 4, February 2005, 35-38.

[15] Thomas Stutzle and Holgar Hoos. Max-Min Ant System And Local Search For The Traveling Salesman Problem, IEEE Conference 1997

[16] Hisham Al-Mubaid and Syed A. Umair. A New Text Categorization Technique using Distributional Clustering and Learning Logic, IEEE Transactions on Knowledge and Data Engineering, Volume 18, No. 9, pp 1156 – 1165, September, 2006.