Hybrid ACO and TOFA Feature Selection Approach for Text Classification

Hanan S. Alghamdi
Information Systems Department, King Abdul-Aziz University, Jeddah, Saudi Arabia
hsaalghamdi@kau.edu.sa
Computing Department, School of Electronics and Physical Sciences, University of Surrey, Guildford, UK
h.alghamdi@surrey.ac.uk

H. Lilian Tang
Computing Department, School of Electronics and Physical Sciences
University of Surrey, Guildford, UK
h.tang@surrey.ac.uk

Abstract—With the highly increasing availability of text data on the Internet, selecting an appropriate set of features for classification becomes more important, for not only reducing the dimensionality, but also improving the classification performance. This paper proposes a novel feature selection approach to improve the performance of text classifier based on an integration of Ant Colony Optimization (ACO) and Trace Oriented Feature Analysis (TOFA). ACO is a metaheuristic algorithm derived by the study of foraging behavior of ant, whilst TOFA is a unified optimization framework to integrate and unify several state-of-the-art dimension reduction algorithms. It has been shown in previous research that ACO is one of the promising approaches for feature selection and TOFA is capable of dealing with large scale text data. For scalable yet effective text classification, the proposed approach make use of TOFA and classifier performance as heuristic information for ACO. The results on Reuters and Brown public datasets demonstrate the effectiveness of the proposed approach.

Keywords—Ant Colony Optimization; Trace Oriented Feature Analysis; feature selection; text classification

I. INTRODUCTION

The amount of data available on the Internet has made the task of data analysis more and more challenging. Text classification (TC) is to assign a given data instance into a predefined set of categories [1][2] and it has become one of the central theme in analyzing complex data. Applications include spam filtering, automated indexing of scientific articles, personalized information dissemination, among many others. For a TC task, a text document is typically represented as a feature vector of term weights in a bag of words feature space[2][3]. Such representation poses the problem of very high dimensionality of feature space and data sparseness which not only cause the computational cost but also significantly degrade the classification performance [4] [5]. Therefore it is highly desirable to reduce the dimensionality of feature space. Two commonly used dimension reduction techniques are: Feature Extraction (FE) and Feature Selection (FS) [4]. FE such as Latent Semantic Indexing (LSI) and Principle Components Analysis (PCA) generates a much smaller dimension space from the original one through algebraic transformation [6][2][7]. However, the associated complexity with FE algorithms is often too high to be applied on real-world text data. On the other hand, FS searches for the most representative features according to some criterion. Information Gain (IG), CHI, Document Frequency (DF), Entropy-based Ranking (En) and some population based optimization algorithms such as Genetic Algorithm (GA) and Ant Colony Optimization (ACO) are among several algorithms used for FS [8][9][10][11][12]. However, FS, in contrast to FE, is usually treated as NP-hard problem in which the size of the feature space is 2^n, where n is the number of the features. Thus, for the TC, searching through the whole feature space to find the optimal solution is not practical. Therefore, FS usually involve heuristic or random search and suboptimal solutions are obtained.

Trace Oriented Feature Analysis (TOFA) is a novel dimension reduction algorithm developed by Jun Yan and colleagues in order to unify FE and FS algorithms through optimization framework to find optimal feature subset[13]. TOFA optimizes the feature extraction objective function in the solution space of the feature selection algorithms which makes TOFA feasible to analyze large scale text data. One of the interesting property of TOFA is that some state-of-the-art dimension reduction algorithms have been proven to be special cases of TOFA and it is applicable to supervised, semi-supervised and unsupervised learning [14]. TOFA basically computes scores of original features and selects features with the highest scored, that is, according to TOFA, they are the optimal feature set for text analysis. TOFA has been proven mathematically and experimentally to be more effective than other previous techniques[13] [14].

In this paper, TOFA has been tested for TC task. The initial results show that the optimal feature set generated by TOFA has improved the classifier performance. However, through experiments, we find that the classifier performance can be further improved by using some other features and it seems that
there is no heuristic that can guide the search to find the optimal subset of features every time. However, ACO is particularly attractive for such type of problems, the optimal feature set can be discovered by ants as they proceed throughout the search space [8][3]. In this paper a new optimization-based FS algorithm is proposed based on ACO metaheuristic algorithm and TOFA. The results show the effectiveness of the proposed algorithm.

The rest of the paper is organized as follows. Section 2 gives a brief overview of ACO. TOFA is reviewed in Section 3. Section 4 explains the proposed feature selection approach. The computational experiments and the obtained results are discussed in Section 5. Section 6 concludes the paper and states the future work.

II. ANT COLONY OPTIMIZATION (ACO)

Since the first development of Ant Colony Optimization (ACO) by M. Dorigo in 1991-1992 and its successful application to the Travelling Salesman Problem (TSP), ACO has attracted the attention of many researchers and has been successfully applied to several real-world optimization problems such as course timetabling, project scheduling, classification rules, DNA sequencing, etc. [15]. ACO belongs to the field of Swarm Intelligence, meta heuristics, nature-inspired algorithms [10]. ACO takes inspiration from the foraging behavior of real ants, specifically the pheromone communication between ants to find the shortest path to the food source. Once food is encountered, an ant will deposit a chemical substance called pheromone on that path; a pheromone trail. Other ants while moving at random, if any ant finds a previously laid pheromone trail, it can decide with high probability to follow it and thus laying down its own pheromone which makes the path more attractive to other ants. However, pheromone trail evaporates over time. Given that, over time, shortest path will be reinforced by more ants while others will be diminished until all ants follow the same shortest path. ACO exploits similar mechanism for solving optimization problems. Artificial ants move randomly in the solution space and build partial solution incrementally. Ants deposit a certain amount of pheromone on the solution components depending on the quality of the solution. Subsequent ants use the pheromone information and problem specific heuristics as guide toward promising solution components[16]. ACO is applicable to any optimization problem as long as the following aspects can be defined:

- **Appropriate problem representation.** The problem can be described as a graph with a set of vertices and edges.

- **Heuristic desirability.** A suitable heuristic measure of the quality of the solution components.

- **Solution Construction Mechanism.** To ensure that only feasible solutions are constructed.

- **Pheromone updating rule.** A suitable method to update the pheromone concentrations and the corresponding evaporation rule.

- **Probabilistic transition rule.** The rule that determines the probability of an ant to use one node in the graph in its path.

ACO is an iterative algorithm consists of three main steps: a number of solutions are constructed by ants at each iteration; solutions could be enhanced through local search and finally the pheromone is updated based on the quality of these. However, several variants of ACO have been proposed in the literature such as Ant System (AS), Ant-Q, Ant Colony System, Max-Min AS and Hyper-Cube AS [16].

III. TRACE ORIENTED FEATURE ANALYSIS (TOFA)

TOFA as presented in [13] and [14] is a matrix trace oriented optimization framework aims to unify both feature extraction and feature selection algorithms through optimization framework. Optimal feature subset is produced and used to project the original high dimension matrix, representing the data, into much lower dimension matrix used to train the classifier[13]. TOFA optimizes the feature extraction objective function in the solution space of the feature selection algorithms which makes TOFA feasible to analyze large scale text data. TOFA is attractive since it is feasible for application on large scale text data and can handle supervised, unsupervised and semi-supervised text analysis [14]. The objective function of TOFA is to maximize:

$$J_{\text{TOFA}}(W) = \text{tr}(W^T (\lambda S_b + (1 - \lambda) C) W)$$  \hspace{1cm} (1)

Where, $W$ is the matrix of bag of words (BOW) in which each text document is represented as a numerical vector of weights. $S_b$ is the inter-class scatter matrix, $C$ is the covariance matrix of the text documents and $\lambda$ is a positive real parameter.

To maximize the objective function of TOFA, each feature is scored and the highest scored features are selected for dimension reduction. The experiments show the superiority of TOFA over other previous techniques[13] [14].

IV. PROPOSED ACO-TOFA FEATURE SELECTION ALGORITHM

For a given TC task, the problem of feature selection can be formulated as follows: given the original data samples, $X \in R^{d \times n}$, of $d$ features and $n$ documents, find $X' \in R^{p \times n}$ of $p$ features and $n$ documents (where $p$ is much less than $d$) such that the classification accuracy is maximized. The FS problem can be represented as ACO optimization problem as follows:

- **The problem can be described as a graph with a set of vertices representing $d$ features and edges representing the next feature to be selected.**
• Heuristic desirability $h_i$ and pheromone trail intensity $\tau_i$ are associated with each feature $f_i \in F = (f_1, ..., f_d)$. Where $F$ is the original feature set.

• A set of $m$ ants search through the feature space to construct a subset $S = (f_1, ..., f_p)$ where $p$ is much less than $d$.

• Each ant $k$, utilizes both the heuristic desirability $h_i$ and pheromone trial $\tau_i$ as a probabilistic transition rule to a possible feature.

• The classification accuracy $C_k$ is used to evaluate the constructed solution of $k$ ant.

• Updating pheromones phase is achieved by decreasing all the pheromone values associated with all features through pheromone evaporation and by increasing the pheromone values associated with best so far solutions.

We propose to use a hybrid selection approach that is able to evaluate the overall quality of the constructed features subsets as well as the individual importance of features. A classification algorithm is used to evaluate the performance of the feature subsets. On the other hand, the individual importance of a given feature is measured using TOFA. After initialization, $m$ ants would construct different possible subsets. Each ant $k$ will select randomly the first feature to construct its feature subset $S_k$. Then each ant will choose independently the feature to be selected next until the full subset is constructed. The probability of ant $k$ to include feature $f_i$ in its solution is given by:

$$ p_k^i = \begin{cases} \frac{\tau_i^\alpha \eta_i^\beta}{\sum_{i \in \text{not included}_k} \tau_i^\alpha \eta_i^\beta} & \text{if } i \in \text{not included}_k \\ 0 & \text{otherwise} \end{cases} $$

Where $\tau_i$ and $\eta_i$ are the pheromone value and TOFA score of feature $f_i$ respectively. The parameters $\alpha$ and $\beta$ determines the relative importance of pheromone versus the heuristic information and $\sum_{i \in \text{not included}_k} \tau_i^\alpha \eta_i^\beta$ is the summation of the product $\tau_i^\alpha \eta_i^\beta$ of all features that are still not included in the partial solution of ant $k$.

After every ant completes its solution, pheromone is updated to increase the pheromone of good features and decrease those that are associated with bad ones as follows:

$$ \tau_i \leftarrow (1 - \rho) \cdot \tau_i + \Delta \tau_i $$

Where $\rho$ a user is defined parameter and is called evaporation rate, and $\Delta \tau_i$ represents the sum of the pheromone contributions of all ants that used feature $f_i$ to construct their solutions. The pheromone contributions are proportional to the quality of the solutions, i.e. the better the solution is, and the more pheromone quantity will be added to features used in that solution. In this work, the quality of the constructed solution is determined by the classifier accuracy, $C_k$, and the quantity of the pheromone laid on feature $f_i$ is:

$$ \Delta \tau_i = \sum_{k=1}^{m} \Delta \tau_i^k $$

Where $m$ is the number of ants and $\Delta \tau_i^k$ is the amount of pheromone laid on feature $f_i$ by ant $k$, which computed as follows:

$$ \Delta \tau_i^k = \begin{cases} Q \times C_k & \text{if ant } k \text{ uses feature } f_i \\ 0 & \text{otherwise} \end{cases} $$

Where $Q$ is a constant parameter. $C_k$ is the classifier accuracy when using the feature subset constructed by the $k$th ant. ACO-TOFA Feature Selection Algorithm is shown in Algorithm 1.

Algorithm 1 : ACO-TOFA Feature Selection Algorithm

**Input:** data samples for training $X = (x_1, x_2, ..., x_n)$

**Output:** The optimal feature subset for classification

1. Initialize TOFA Parameters
2. Generate Heuristic Information
3. Initialize ACO Parameters
4. Initialize Pheromone;
5. Determine the population of ants $(m)$;
6. For each ant $k$ do
   \[ \text{Repeat} \]
   \[ \text{Choose in probability the feature to include;} \]
   \[ \text{Use TOFA scores to adjust probability selection;} \]
   \[ \text{Append the partial solution with the candidate feature;} \]
   \[ \text{Until ant } k \text{ has chosen } p \text{ features} \]
   \[ \text{End for} \]
   \[ \text{For each feature } f_i \text{ do} \]
   \[ \text{Evaluate the constructed subset } S_k; \]
   \[ \text{Use classifier accuracy } C_k \text{ to evaluate the candidate feature subset;} \]
   \[ \text{If (termination condition not met) do} \]
   \[ \text{For each feature } f_i \text{ used in } S_k \]
   \[ \text{Update Pheromone } p_i \text{ based on the solution quality, } C_k \]
   \[ \text{End for} \]
   \[ \text{For each feature } f_j \text{ do} \]
   \[ \text{Evaporate Pheromone } p_j \]
   \[ \text{End for} \]
V. EXPERIMENTS

A. Experimental Setup

To demonstrate the effectiveness of the proposed approach, two public text datasets are used. Experiments were carried out on samples from Brown Corpus, which was the first million-word electronic corpus of English, created in 1961 at Brown university, and The Reuters Corpus contains 10,788 news documents. The two datasets have been categorized by genre such as news, religion, reviews and so on. Table 1 shows the number of categories and the number of documents in each dataset.

Table 1: Brown and Reuters datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Number of Documents</th>
<th>Number of categories</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brown</td>
<td>500</td>
<td>15</td>
</tr>
<tr>
<td>Reuters</td>
<td>10788</td>
<td>90</td>
</tr>
</tbody>
</table>

The TOFA and ACO-TOFA algorithms were implemented using Python coding and tested on a computer with 2.80GHz processor and 8GB of RAM. Each text document was represented as a vector in the bag of words model using TFIDF indexing:

$$\text{tfidf}(t_i, d_j) = \frac{[(t_i, d_j)]}{|d_j|} \cdot \log \frac{n}{|\{d|t_i \in d\}|}$$ (6)

Where \(\text{tfidf}(t_i, d_j)\) is the weight of term \(t_i\) in \(d_j\) document, \([t_i, d_j]\) stands for the number of appearance of \(t_i\) in \(d_j\), \(|x_j|\) is the total number of terms in \(d_j\) document \(n\) is the total number of documents and \(|\{d|t_i \in d\}|\) is the numbers of documents contain \(t_i\) term. The parameters were set as follows: \(\alpha = 1\), \(\beta = 0.1\), \(m = 10\) and \(\lambda = 1\) in TOFA and ACO-TOFA algorithms.

B. Key steps:

The experiments consist of three main steps. First, all features extracted and indexed are used to train two text classifiers, Naïve Bayesian and Decision Tree. Then TOFA and TOFA-ACO are tested with different reduction percentage ranging from 0.05 up to 0.07 of the original feature sets. The reduced features are used to train the text classifiers.

C. Evaluation Metric:

In this work, the classifier accuracy is used to evaluate the performance of the TC algorithms. The accuracy is measured as the percentage of correctly classified data in the testing dataset and is given by the following:

$$\frac{\text{number of correctly classified test samples}}{\text{total number of test samples}}$$ (7)

D. Results

Table 2, shows the results of TC for the two datasets, Brown and Reuters. First, all extracted and indexed features are used to train the Naïve Bayesian and Decision Tree classifiers. Then, the features sets are reduced to 5% of the original feature space by using ACO-TOFA and TOFA reductions.

From table 2 we can see that the reduction of the feature space by using the proposed approach has improved the accuracy of the Naïve Bayesian classifier while giving comparable or even better results with the Decision Tree classifier.

For demonstration of the proposed algorithm when reducing the original feature space to different levels, the accuracy of the Naïve Bayesian Classifier on a discrete set of reduction percentage is plotted in Figure 1 and Figure 2 corresponding to Reuters and Brown datasets respectively.

By integrating the results of Figures 1 and 2, we can see that the proposed approach can significantly improve the classification performance when reducing the feature space to much smaller dimension such as 5% of the original features.

Table 2: Results of Classification Accuracy on Brown and Reuters datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Algorithm</th>
<th>Selected Features</th>
<th>Naïve Bayesian</th>
<th>Decision Tree</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brown</td>
<td>ACO-TOFA</td>
<td>5%</td>
<td>0.71</td>
<td>1.00</td>
</tr>
<tr>
<td></td>
<td>TOFA</td>
<td>5%</td>
<td>0.14</td>
<td>0.29</td>
</tr>
<tr>
<td></td>
<td>All Features used - no selection performed</td>
<td></td>
<td>0.21</td>
<td>1.00</td>
</tr>
<tr>
<td>Reuters</td>
<td>ACO-TOFA</td>
<td>5%</td>
<td>0.90</td>
<td>0.57</td>
</tr>
<tr>
<td></td>
<td>TOFA</td>
<td>5%</td>
<td>0.87</td>
<td>0.57</td>
</tr>
<tr>
<td></td>
<td>All Features used - no selection performed</td>
<td></td>
<td>0.41</td>
<td>0.41</td>
</tr>
</tbody>
</table>

Figure 1: Classification Accuracy of Naïve Bayesian on Brown dataset

Figure 2: Classification performance of Naïve Bayesian on Reuters dataset
VI. CONCLUSION AND FUTURE WORK

This paper presents a novel feature selection procedure based on the Ant Colony Optimization metaheuristic and Trace Oriented Feature Analysis. The proposed algorithm utilizes both the individual importance of features measured by TOFA and the performance of subsets determined by classifier accuracy to search through the feature space. The proposed approach is tested on Brown and Reuters’ datasets using different reduction levels. Naïve Bayesian and Decision Tree are used as text classifiers in this work. The performance of the proposed approach is compared with that of TOFA. The experimental results show that the proposed approach can obtain better classification accuracy when reducing the feature space to much smaller dimension.

In the future, we will further study and analyze the utilization of the proposed approach for other text mining tasks such as text retrieval and document clustering.

ACKNOWLEDGMENT

Hanan Alghamdi thanks the Joint Supervision Program at King Abdul-Aziz University for its financial support for her PhD work and acknowledges the support from the Information Systems Department at the same University.

REFERENCES


