Multi-label Feature Selection using Ant Colony Optimisation with Local Search

Team Members:

- Pratham Nayak - 191IT241
- Naveen Shenoy - 191IT134
- Sarthak Jain - 191IT145

Introduction

- Ant colony optimisation is a heuristic swarm intelligence algorithm which aims to exploit the foraging behavior of ants.
- Ants deploy pheromone on the traversed path which evaporate over time.
- Next node is selected on the basis of pheromone deposition and a probabilistic function.
- It is based on the logic that paths that are a part of the optimal solutions will be traversed more frequently and hence will have a higher pheromone deposition.
- Feature selection is the task of selecting an optimal set of features of a given size with minimal information loss.
- Since feature selection is a combinatorial NP hard problem, heuristic algorithms like ACO can be used to approximate a good enough solution.

Problem Statement & Objectives

To develop an Ant Colony Optimisation(ACO) algorithm with local search to perform multi-label feature selection.

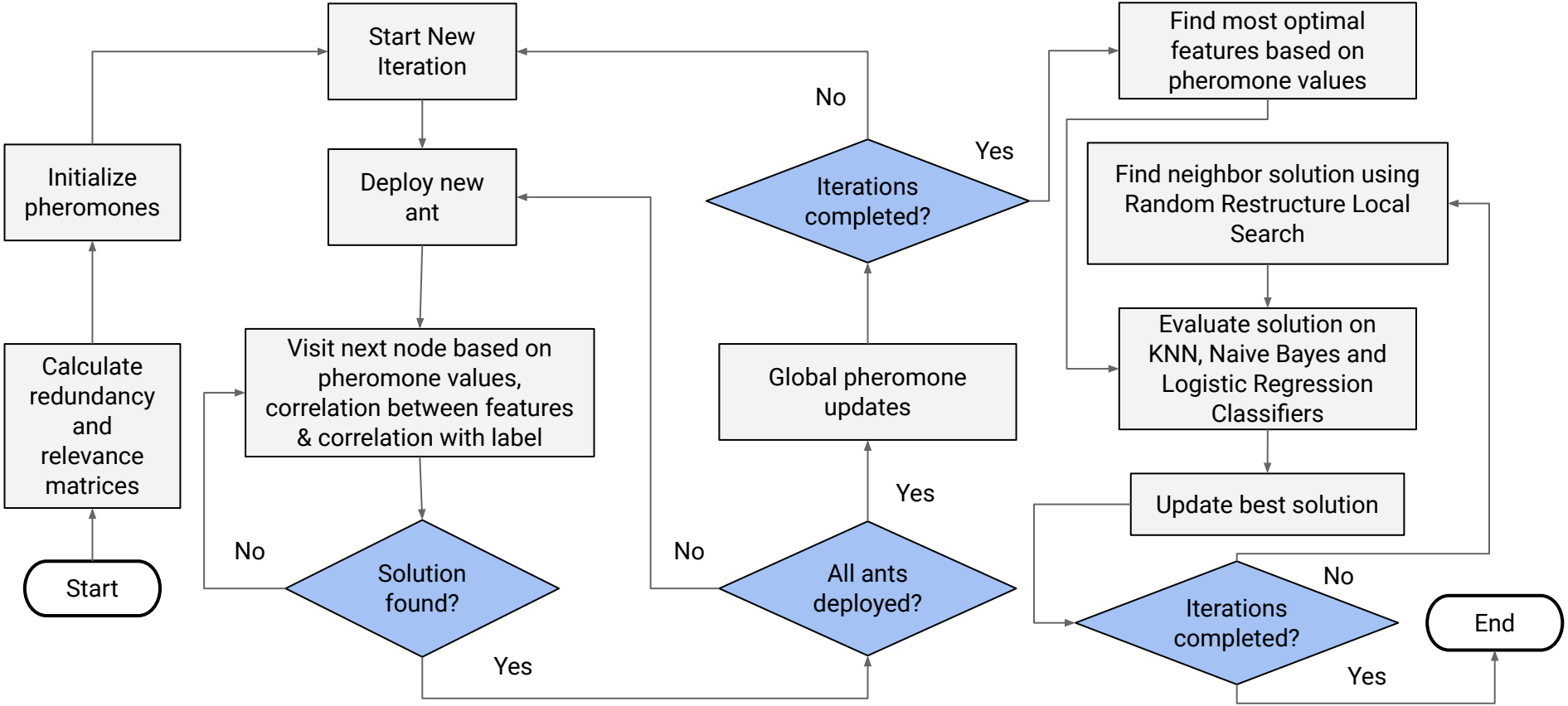
Objectives:

- To convert to wrapper based method to improve performance compared to filter methods.
- To integrate local search to avoid local optimal convergence.
- To evaluate using multiple learning algorithm to avoid inherent bias.

Literature Survey

Authors	Methodology	Merits	Limitations
Paniri et al.	Used ACO for feature selection with feaures as nodes in the graph.	Cosine similarity based initialization along with redundancy and relevance concepts used to get similarity between features and labels.	Used filter based feature selection methods. Evaluated solutions on KNN classifier only.
Jiam et al.	Technique for multi-label feature selection by first lowering the output space's dimension before using feature selection.	Helps to remove the negative effect of the noisy labels in feature selection.	Does not work well when dataset is high dimensional.
Jović et al.	Identified hybrid feature selection methods which are a mixture of evolutionary computation based heuristic algorithms and genetic algorithm.	Found that evaluation of proposed methodologies of feature selection should be made on larger datasets.	Checked feature selection application in only few popular domains.
Kashef and Nezamabadi-pour	First removing the irrelevant features in order to remove number of existing features.	Feature selection process is fast.	Used filter method for algorithm evaluation.

Workflow


Start
Calculate redundancy and relevance matrices
Initialize pheromones
Start New Iteration
Deploy new ant
Visit next node based on pheromone values, correlation between features & correlation with label
Solution found?
No
Yes
All ants deployed?
No
Yes
Global pheromone updates
Iterations completed?
No
Yes
Find most optimal features based on pheromone values
Find neighbor solution using Random Restructure Local Search
Evaluate solution on KNN, Naive Bayes and Logistic Regression Classifiers
Update best solution
Iterations completed?
No
Yes
End

Methodology – Pheromone initialization

- For each feature, the pheromone value is initialized as the maximum of the cosine similarity between the feature and different class labels.

$$cosine\_similarity\ (A, B) = \cos(\theta) = \left| \frac{\sum_{i=1}^{n} (A_i B_i)}{\left(\sqrt{\sum_{i=1}^{n} A_i^2}\right)\left(\sqrt{\sum_{i=1}^{n} B_i^2}\right)} \right|$$

- Because these values can be very small, we apply min–max normalization to scale them in the interval [0 , 1].

$$\tau_{scaled} = \frac{\tau - \tau_{min}}{\tau_{max} - \tau_{min}}$$

Measure features redundancy and relevancy

1. **Redundancy matrix:**
   a. Let the number of features = d.
   b. Redundancy matrix is a (d x d) matrix of the absolute value of the pearson correlation coefficient (PCC) between each pair of features.
2. **Relevance Matrix:**
   a. Let the number of labels = l.
   b. Relevance matrix is a (d x l) matrix of the absolute value of the pearson correlation coefficient (PCC) between each feature and the label (encoding vector).

$$correlation\,(A,B) = \left| \frac{\sum_{i=1}^{n}\left(A_i-\overline{A}\right)\left(B_i-\overline{B}\right)}{\left(\sqrt{\sum_{i=1}^{n}\left(A_i-\overline{A}\right)^2}\right)\left(\sqrt{\sum_{i=1}^{n}\left(B_i-\overline{B}\right)^2}\right)} \right|$$

Methodology- State transition rule

- $\eta_1(F_j)$ is the maximum cosine similarity between feature j and its corresponding class labels.
- $\eta_2(F_i, F_j)$ is the inverse of the correlation between feature i and feature j.
- τ_j is the pheromone deposited on node j.
- Based on a randomly chosen value q, the next node is selected using the following transition rule.

$$P_{ij}^{k}(t) = \begin{cases} \dfrac{[\tau_j]\left[\eta_1\left(F_j\right)\right][\eta_2(F_i, F_j)]^{\beta}}{\sum_{u \in N_i^k}[\tau_u][\eta_1(F_u)][\eta_2(F_i, F_u)]^{\beta}}, & \forall j \in N_i^k, \quad if \;\; q > q_0 \\ 0, & otherwise \end{cases}$$

$$j = \underset{u \in N_i^k}{\arg\max} \left\{[\tau_u][\eta_1(F_u)]\left[\eta_2\left(F_i, F_u\right)\right]^{\beta}\right\}, \;\; if \;\; q \leq q_0$$

Methodology – Global pheromone updating rule

- Global updates take place after each iteration when all deployed ants have found solutions.
- The number of times a feature is visited by an ant will be stored in a vector called “feature counter (FC)”.
- Each time an ant traverses a feature, it increments the “FC” value corresponding to that feature by one.
- Finally during global updates, a fraction of the pheromone is evaporated and additional pheromone is added based on "FC".

$$\tau_i\,(t+1) = (1-\rho)\,\tau_i\,(t) + \frac{FC(i)}{\sum_{i=1}^{d} FC(i)}$$

Methodology – Generating and Evaluating the solution

- Once all the iterations are completed, generating the solution involves selecting the top features with the highest pheromone content as the set of optimal features.
- Three classifiers i.e KNN, Naive Bayes and Logistic Regression are used to evaluate the solution.
- The performance is measured as the average of the three accuracies.

Methodology – Improving Solution using Local Search

- Given a candidate solution, local search attempts to move to a neighbour solution.
- The neighbour solution is such that it differs from the candidate solution by a minimum extent.
- In **Random Restructure** Local Search Algorithm, out of the selected features that are part of the generated solution, **only half** of them are retained.
- The remaining half of the features are regenerated by applying the state transition rule of the ant colony optimisation.
- The next features are selected according to equation:-

$$j = \underset{u \in N_i^k}{\arg \max} \left\{ [\tau_u] [\eta_1(F_u)] [\eta_2 (F_i, F_u)]^{\beta} \right\}$$

- This corresponds to the exploitative search of the ACO state transition rule.

Methodology – Improving Solution using Local Search

- After generating a feature set using local search, the solution is evaluated on the three machine learning classifiers.
- The current best solution is updated based on the accuracy of the solution generated using local search.
- Local search is applied for a fixed number of iterations.

Innovation

- The proposed approach is a wrapper method, whereas the base paper implements a filter method which are known to be less effective.
- The proposed approach uses local search techniques to avoid getting stuck at local minima, which is a common problem with ant colony based algorithms.
- Wrapper methods can generate solutions that are only effective for the selected learning algorithm, hence we use an average of the performance across multiple learning algorithms to remove any bias caused by the learning algorithm.

Evaluation Metric

- Accuracy is used as the evaluation metric to compare different feature subsets.
- For multi label evaluation accuracy metrics that is used in all performed experiments defines accuracy as the fraction of the correctly predicted labels among all predicted and actual labels.

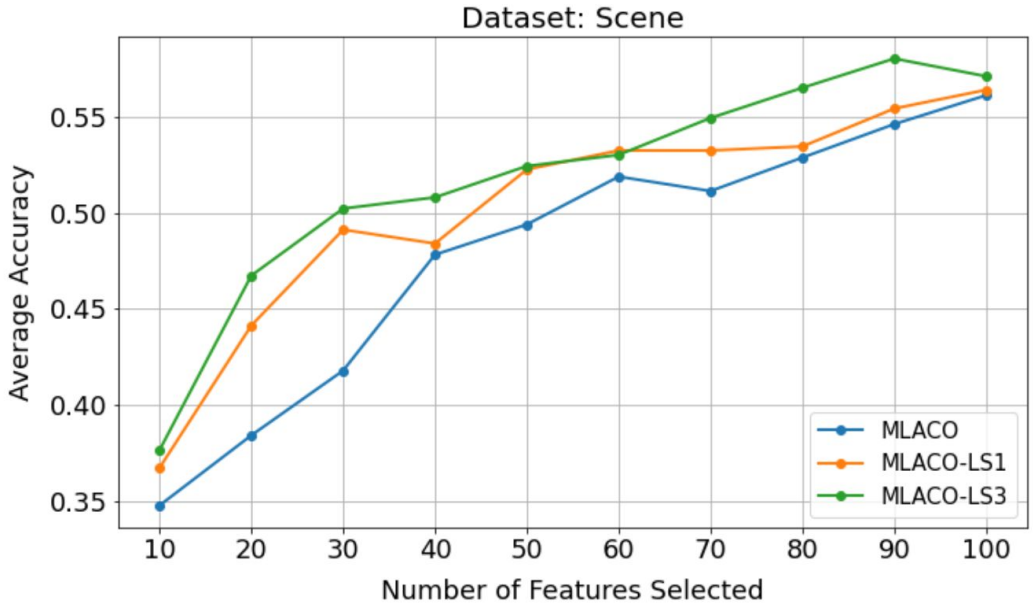
$$Accuracy(Y, Z) = \frac{1}{n} \sum_{i=1}^{n} \frac{|Y_i \cap Z_i|}{|Y_i \cup Z_i|}$$

Dataset

For this project we have considered three datasets :

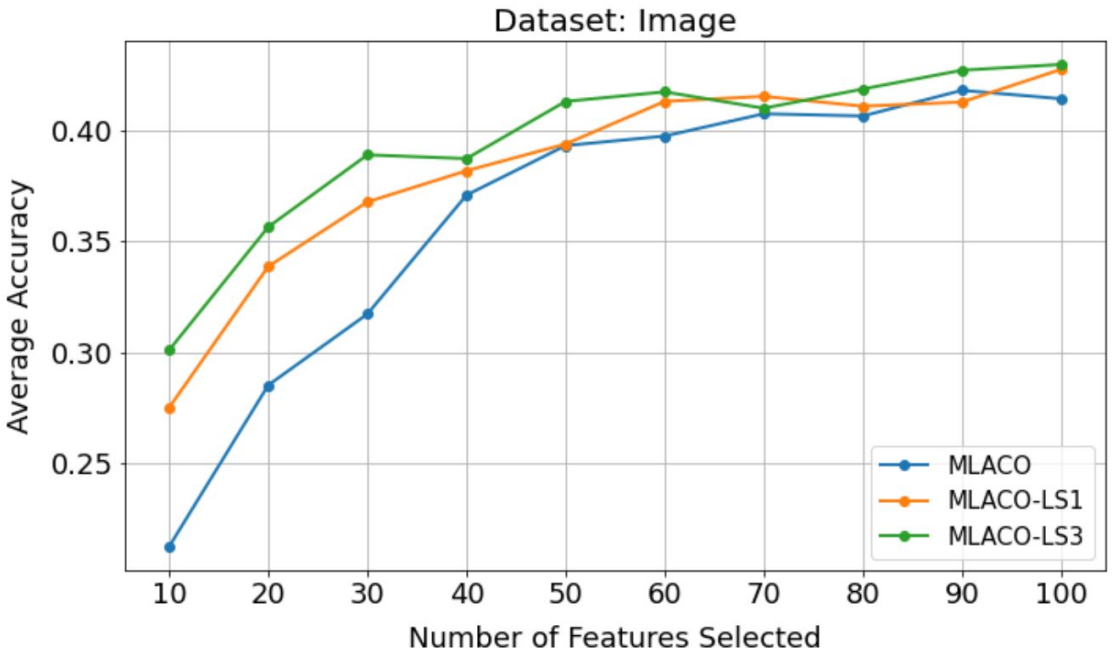
1) **Emotions Dataset**: Classifying music into 6 emotions, namely amazed-surprised, happy-pleased, relaxing-calm, quiet-still, sad-lonely and angry-aggressive. A total of 593 instances are present in the dataset.
2) **Image Dataset**: The Image dataset consists of 2000 images. he images are then stored in the dataset as 294 - dimensional feature vectors. The number of class labels in the Image dataset is 5.
3) **Scene Dataset**: Scene is an image dataset containing 2407 images and 6 classes, i.e. beach, sunset, fall foliage, urban, mountain and field. All images are represented as 294-dimensional feature vectors

Results



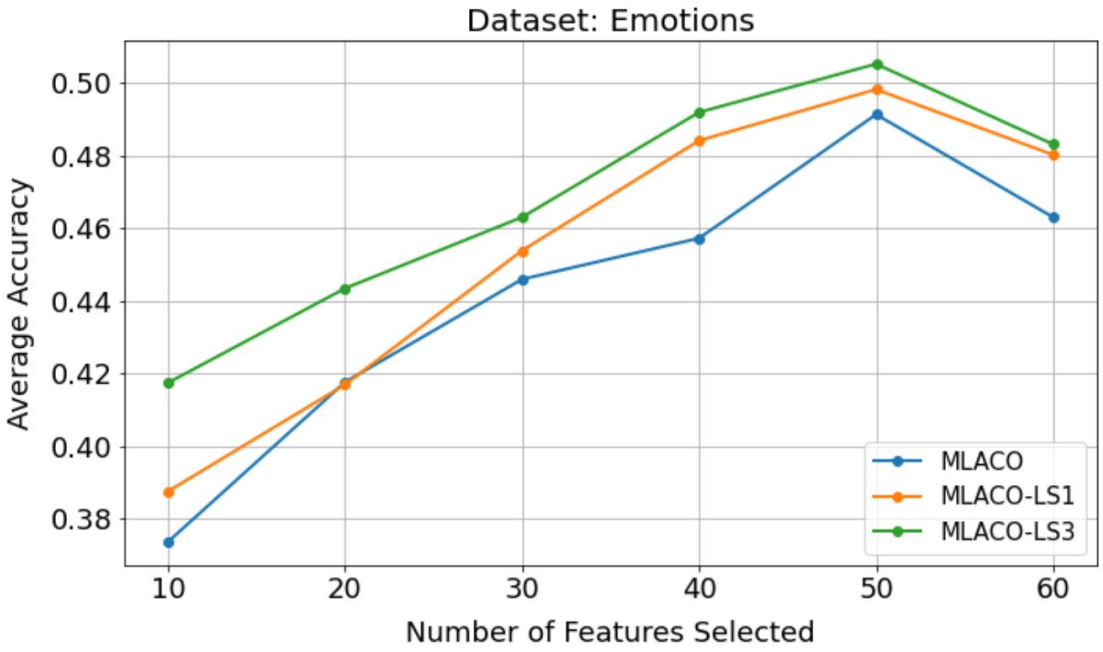
- The figure shows the average accuracies obtained for different numbers of selected features for the Scene dataset.
- MLACO gives the least average accuracy.
- MLACO-LS1 performs better due to the local search optimisation using a KNN-based wrapper.
- MLACO-LS3 gives the best performance, and this is because a combination of 3 different classifiers removes inherent bias of the classifiers.

Results



- Figure shows the average accuracies obtained for different number of selected features for the Image dataset.
- When the number of selected features is small (< 60), the results are similar to that of the Scene dataset.
- However, when a large number of features are selected, the performance of all 3 feature selection techniques is almost identical.

Results



- Fig. 4 shows the average accuracies obtained for different number of selected features for the Emotions dataset.
- MLACO-LS3 outperforms the other algorithms. When the number of selected features is 60, the performance drops.
- This can be due to increased noise, overfitting or the curse of dimensionality.

Results

AVERAGE ACCURACY OVER 10 FINAL FEATURES' SUBSET

Dataset	MLACO	MLACO-LS1	MLACO-LS3
Scene	0.348	0.367	0.376
Image	0.212	0.275	0.301
Emotions	0.374	0.388	0.417

- MLACO-LS3 gives the best performance, followed by MLACO-LS1.
- This shows that local search combined with a multi-classifier based wrapper method helps to significantly improve the quality of the selected features by avoiding local optima through local search and by removing inherent bias by evaluating the generated solution using multiple classifiers of different types.

THANK YOU