# Multi-objective Multi-label Feature Selection using NSGA-II

Zahra Hashemi
Department of Computer Engineering, University of Kurdistan, Iran
zahrahashemy2023@gmail.com, zahra.hashemi@uok.ac.ir

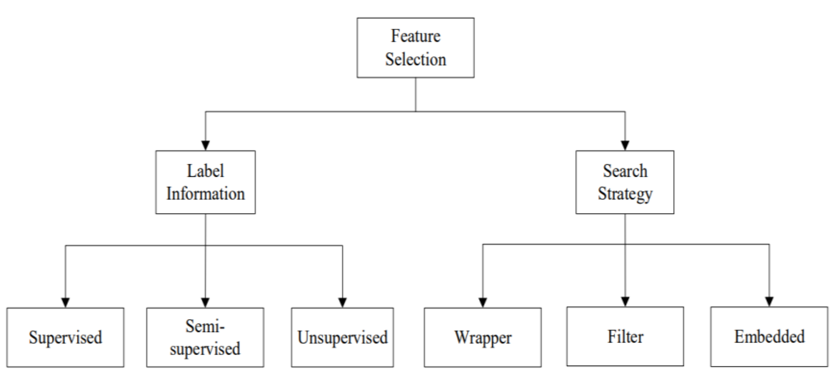
## 1- Introduction

In the field of machine learning, feature selection stands as a pivotal pre-processing step, aimed at enhancing model performance by eliminating redundant or irrelevant features. This project focuses on using Non-dominated Sorting Genetic Algorithm II (NSGA-II) for multi-objective feature selection in multi-label classification tasks. The main objectives of this approach are to minimize the hamming loss and the number of features used.

Multi-label classification refers to a scenario where a single instance can have multiple class labels, and the labels are no longer mutually exclusive. To improve the classification performance in this case, feature selection is applied to select the best subset of the original feature set. There are three main methods for feature selection in multi-label classification: filter, wrapper, and embedded methods.

- Filter methods utilize some measure criteria to sort the feature importance in descending order or select some informative features using a greedy search process (e.g., sequential forward selection), in which no classifier is used.
- Embedded methods combine classifier design with feature selection in a single optimization process, whose computational costs are extremely high. Additionally, the final subset of the selected feature depends on the specific classifier to some extent.
- Wrapper methods mainly consist of two elements, which are iteratively executed until converging. One is to apply some search strategy (e.g., genetic algorithm) to find several subsets of original features and the other is to train a multi-label classifier using selected subsets to evaluate this subset fitness. On computational cost, wrapper methods run faster than embedded ones. Typically, NSGA-II for feature selection is a wrapper method.

Also, according to the absence or presence of class labels of data, the feature selection methods can be divided into supervised, unsupervised, or semi-supervised.



Feature selection is a crucial task that aims to enhance the accuracy of a classification model and reduce its computational complexity. However, selecting too few relevant features can degrade the accuracy, while selecting too many can increase the computational complexity. Therefore, multi-objective feature selection algorithms are used to simultaneously optimize multiple objectives, including classification accuracy and the number of selected features. The primary goal of such algorithms is to minimize the number of features while ensuring an acceptable classification accuracy, which is a challenging task due to the conflicting nature of these objectives.

The NSGA-II (Non-dominated Sorting Genetic Algorithm II) [1] is a popular and widely used algorithm in the field of evolutionary computing for solving multi-objective optimization problems. Developed by Kalyanmoy Deb et al. in the early 2000s, it is an improvement over the original NSGA algorithm, designed to provide a better sorting method and a more efficient way to maintain diversity among solutions.

For this project, I employed the multi-label k-nearest neighbors (ML-KNN) [2] algorithm as the foundation for a new feature selection wrapper. To optimize the selection of features within this wrapper, I adopted two objective functions: minimizing the number of features and reducing the Hamming loss. The evolutionary multi-objective optimization algorithm, NSGA-II, was utilized to navigate this optimization process.

## 2- PREVIOUS WORK

The article "A Multi-label Feature Selection Algorithm Based on Multi-objective Optimization" [3] presents a multi-label feature selection method using the NSGA-II algorithm to optimize two objective functions: average precision and Hamming loss. The study highlights the conflict between these objectives in multi-label classification and proposes a wrapper feature selection method using the multi-label k-nearest neighbor (ML-kNN) method. The proposed approach, evaluated through experiments, shows improved performance over existing techniques. The paper emphasizes the importance of optimizing multiple objectives simultaneously in multi-label classification contexts.

The article "An evolutionary decomposition-based multi-objective feature selection for multi-label classification" [4] presents a novel approach to feature selection in multi-label classification tasks. The authors introduce a multi-objective optimization algorithm that enhances the decomposition-based multi-objective optimization method. This method divides the multi-label feature selection problem into single-objective subproblems and solves them simultaneously using an evolutionary algorithm, which speeds up the optimization process and leads to a more diverse set of feature subsets. They implement a local search operator to refine the solutions and a pool of genetic operators to generate new feature subsets. The proposed method is evaluated against other multi-objective feature selection approaches using real-world datasets, and it shows improvements in classification accuracy with a reduced number of features.

The article "MLACO: A multi-label feature selection algorithm based on ant colony optimization" [5] details the development of a feature selection algorithm tailored for multi-label classification problems. The algorithm applies the principles of ant colony optimization (ACO) to identify the most informative and least redundant features from datasets. It employs both unsupervised and supervised metrics to guide the search process. The study demonstrates that this MLACO algorithm outperforms other feature selection methods, achieving better classification performance while also being computationally efficient. This work is particularly relevant for projects aiming to enhance multi-label classifiers by optimizing feature subsets.

## 3- Methodology

This project involves multi-label classification, which requires a more nuanced approach than traditional single-objective optimization due to its complexity. We aim to strike a balance between two key objectives: minimizing the Hamming loss and reducing the number of features. The Hamming loss is a crucial metric that measures the average difference between predicted and actual labels, indicating the accuracy of the classification. The feature count, on the other hand, affects the model's complexity and computational demands. However, there is no linear relationship between these objectives. Reducing the number of features too much could simplify the model at the expense of valuable information, leading to higher Hamming loss. Alternatively, increasing the number of features indiscriminately could cause overfitting and unnecessary computational overhead, without a proportional gain in accuracy. To navigate this trade-off efficiently, we have opted for multi-objective optimization, specifically using NSGA-II.

The Non-dominated Sorting Genetic Algorithm II (NSGA-II) is an evolutionary algorithm designed for solving multi-objective optimization problems. It works by evolving a population of candidate solutions over several generations. NSGA-II uses a fast non-dominated sorting approach to classify solutions into different fronts based on their level of dominance. It also applies a crowding distance mechanism to maintain diversity among solutions. The algorithm selects, crosses, and mutates solutions to explore the search space, aiming to find a set of optimal solutions that represent the best trade-offs among the objectives.

The multi-label k-Nearest Neighbors (ML-kNN) method is an extension of the traditional k-NN algorithm tailored for multi-label classification. It operates in two main phases: training and testing. During training, the algorithm learns from the given dataset, considering the labels associated with each instance. In the testing phase, it predicts the labels for new, unseen instances based on the learned model. This method leverages the simplicity and effectiveness of k-NN while addressing the unique challenges posed by multi-label data.

The Multi-label Multi-objective Feature Selection Method using the NSGA-II algorithm is an advanced approach designed to tackle the complexities of multi-label classification problems. By employing the NSGA-II algorithm, this method optimizes for multiple objectives simultaneously, such as minimizing error rates (like Hamming loss) and reducing the number of features. This dual focus helps in identifying the most relevant and informative features while ensuring the classification model remains efficient and effective. The method leverages the strengths of evolutionary algorithms to navigate the trade-offs between these conflicting objectives, resulting in a balanced and optimized feature set.

**Algorithm: NSGA-II for Feature Selection in Multi-label Classification**

**Input:**

Data: Multi-label dataset divided into training and test sets

NP: Population size

R: Number of iterations (generations)

p_crossover: Crossover probability

p_mutation: Mutation probability

mu: Mutation rate

**Output:**

Optimal feature subsets based on the Pareto front

**Initialization:**

Randomly generate an initial population of feature subsets.

Evaluate each subset using a multi-label classifier (e.g., ML-KNN) based on objectives like Hamming loss and feature count.

**Non-dominated Sorting:**

Sort the population into Pareto fronts based on dominance relations.

**Crowding Distance Calculation:**

For each front, calculate crowding distances to maintain diversity among solutions with similar ranks.

**Selection**:

Use roulette wheel selection to choose parent solutions for mating, considering both their rank and crowding distance.

**Crossover**:

Perform crossover operations (single-point, double-point, or uniform) on selected parents to produce offspring, introducing new feature combinations:

Single-Point: Split two parents at a random position and exchange segments.

Double-Point: Exchange the segments between two randomly chosen points from each parent.

Uniform: Mix genes from both parents with an equal probability to form offspring.

**Mutation**:

Apply mutation to offspring by flipping feature inclusion bits, governed by the mutation rate mu, to add variability.

**Merge and Select:**

Combine the new offspring with the current population and reapply non-dominated sorting and crowding distance calculation.

Select the top NP solutions for the next generation.

**Loop**:

Repeat steps 2-7 for R iterations, refining the population towards the Pareto optimal front.

**Final Selection:**

Choose the solutions from the first Pareto front as the final set of optimal feature subsets.

**Evaluation:** Assess the final feature subsets on the test set to determine the Hamming loss and other relevant metrics.

## 4- EXPERIMENTS

Before presenting experimental results, the two benchmark data sets and the compared algorithm are briefly introduced.

### 4.1) Datasets

I use two benchmark data sets: Emotions (music) and Scene (image) to validate my project. We can access these multi-label datasets from the Mulan database [6]. Some useful statistics of these data sets are provided in Table I, such as the number of instances in training and test sets, dimensions of features, the number of labels, and the cardinality of average labels.

Table I: Statistics for two benchmark multi-label datasets.

| Dataset | train | test | Features | Labels | Cardinality |
|---|---|---|---|---|---|
| **Emotions** | 474 | 119 | 72 | 6 | 1.869 |
| **Scene** | 1926 | 481 | 294 | 6 | 1.074 |

### 4.3) Parameter setting

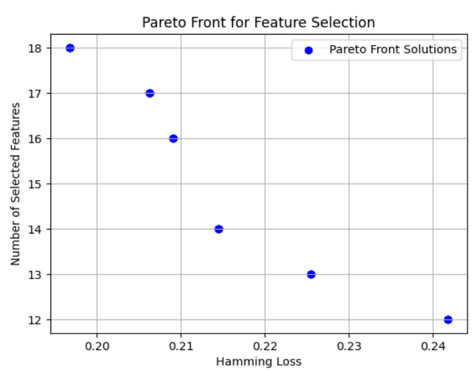
The dataset with multiple labels must be divided into training and test sets. For this, 80% of the data is used for training, and the remaining 20% is used for testing. During the optimization process, the accuracy of the classification is determined by using the selected features on the training data. The test set is used to evaluate the obtained features, and the final results are reported at the end of the algorithm. Table I shows the number of training and test samples for each dataset. The ML-KNN classifier algorithm was used in multi-label learning. To simplify the evaluation process, Euclidean distance with k = 10 was used in this project. The values of parameters P1 and P2 for selecting the crossover operator were set to 0.1 and 0.2, respectively. Some parameters need to be adjusted before the implementation of the algorithm as below:

- Population size: 30
- Max iteration or generation: 100
- Crossover probability: 0.7
- Mutation probability: 1/number of features
- Mutation rate: 0.05

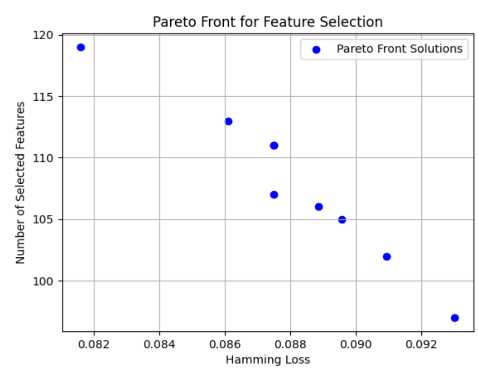
**4.4) Results**

Each dataset introduced in the previous section was given to the algorithm as the input data for evaluating the method. The output of this project showcases the final population of solutions generated by the NSGA-II algorithm for multi-objective feature selection. Each solution represents a set of features selected (denoted by '1') or not selected (denoted by '0') for the classification task. Alongside each solution's feature set ('Position'), there are two associated costs ('Cost'): the first value represents the Hamming Loss, and the second value indicates the number of features selected.

- **Position**: A binary vector where each element corresponds to a feature in your dataset. A '1' indicates that the feature is selected for use in the classification model, and a '0' means the feature is not selected. This binary encoding allows the NSGA-II algorithm to explore different combinations of features to find the optimal subset.
- **Cost**: This is a tuple with two elements. The first element is the Hamming Loss, which measures the average number of incorrect label predictions per instance in the dataset. Lower values are better, as they indicate higher accuracy. The second element is the number of features selected by that solution. The goal is to minimize both the Hamming Loss and the number of features to achieve a balance between accuracy and model simplicity.




a) Emotions




b) Scene

### 4.2) Comparison

I use the MLACO algorithm for comparing the outcomes of my project. The MLACO algorithm, based on Ant Colony Optimization (ACO), is a novel approach designed for feature selection in multi-label classification tasks. It leverages the principles of ACO, where a colony of artificial ants simulates the process of finding the shortest paths to food, to identify the most relevant and least redundant set of features. This method optimizes feature selection by considering both the individual and collective relevance of features, akin to how ants assess multiple paths to determine the most efficient route.

The relevance of MLACO to NSGA-II lies in their shared goal of optimizing complex problems through bio-inspired algorithms. While NSGA-II is grounded in the principles of evolutionary biology, applying concepts like mutation, crossover, and selection to evolve a population of solutions, MLACO draws inspiration from the foraging behavior of ants. Both algorithms are adept at handling multi-objective optimization, making them suitable for tasks like feature selection where trade-offs between objectives (e.g., maximizing classification accuracy while minimizing feature set size) are common.

In this section, I report the results of the comparison between the NSGA-II algorithm and MLACO algorithm for multi-label data feature selection. Table 2 shows the minimum Hamming loss in different datasets. Table 2 shows that the NSGA-II algorithm has achieved significantly better results compared with the other algorithm.

Table II: results of the NSGA-II algorithm and the MLACO algorithm.

| **Datasets** | **NSGA-II minimum HL** | **MLACO minimum HL** | **Minimum number of features** |
|---|---|---|---|
| **Emotions** | 0.19 | 0.25 | 12 |
| **Scene** | 0.0816 | 0.174 | 97 |

To evaluate the effectiveness of various methods, it's important to consider the number of features in their obtained subsets, even if the classification error is high. However, it's crucial to compare these solutions with those that achieve a lower classification error than using all features. Table 2 reports the minimum number of selected features from these solutions. This table represents the smallest subset of features that can achieve a classification with a lower Hamming loss than using all features. In the implementation of the ML ACO algorithm, 20% of all features are always selected.

## 5- Conclusion

This project investigated a method for efficiently selecting the best features for classifying data with multiple labels, known as NSGA-II. The results indicated that this algorithm strikes a good balance between accuracy and simplicity. A comparison was made between NSGA-II and MLACO, and the former was found to be a useful tool for accurately classifying complex data in a variety of situations.

## 6- References

[1] K. Deb, A. Pratap, S. Agrawal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," IEEE Transactions on Evolutionary Computation, vol. 6, no. 2, pp. 182-197, Apr. 2002.

[2] M. L. Zhang and Z. H. Zhou, "ML-kNN: a lazy learning approach to multi-label learning," Pattern Recognition, vol. 40, no. 7, pp. 2038-2048, Jul. 2007.

[3] Yin, Tao & Xu (2015) Yin J, Tao T, Xu J. A multi-label feature selection algorithm based on multi-objective optimization. International Joint Conference on Neural Networks (IJCNN); Killarney: IEEE; 2015. pp. 1–7.

[4] Asilian Bidgoli, A., Ebrahimpour-Komleh, H., & Rahnamayan, S. (2020). An evolutionary decomposition-based multi-objective feature selection for multi-label classification. PeerJ Computer Science, 6, e261. pp. 1–23.

[5] Paniri, M., Dowlatshahi, M. B., & Nezamabadi-pour, H. (2019). MLACO: A multi-label feature selection algorithm based on ant colony optimization. Knowledge-Based Systems, [Article Number], [Page Range].

[6] Tsoumakas et al. (2011) Tsoumakas G, Spyromitros-Xioufis E, Vilcek J, Vlahavas I. Mulan: a Java library for multi-label learning. Journal of Machine Learning Research. 2011;12(July):2411–2414.