Introduction

The assignment is to investigate the effectiveness of feature selection using a Genetic Algorithm (GA) in improving the accuracy of emotion classification, specifically for discriminating between “happy” and “sad” emotions using the RAVDESS Facial Landmark Tracking dataset.

1. Data Preprocessing

We started by preprocessing the dataset as follows:

- We cleaned the dataset using the conditions on the csv files i.e. modality == '01' and (emotion == '03' or emotion == '04') and intensity == '02' and statement == '02' and repetition == '02'.
- We removed the columns from 1-3 and 298-433. We added a last column having value 1 for happy and 0 for sad emotion.
- At the end, we had 575 columns and about 11,000 rows in the dataframe.
- Then we loaded the dataset and filtered entries corresponding to happy and sad emotions.
- We separated the features (facial landmarks) and labels (emotions) for further processing.
- Split the dataset into training and testing sets using a 80-20 split ratio.

2. GA Initialization

For GA initialization, we set the following parameters:

- Chromosome representation: Binary string of length 575 indicating whether a feature (landmark) is selected (1) or not (0).
- Population size: 2 individuals.
- Mutation rate: 0.1

3. Fitness Function

The function evaluate_fitness(chromosome) is responsible for computing the fitness score of a chromosome (binary string representing selected features) in a Genetic Algorithm for feature selection. It first selects features from the training data based on the binary chromosome representation. Then, it uses a neural network model using Keras Sequential API with a specific architecture consisting of densely connected layers. The number of input neurons is determined by the number of selected features, and the output layer size corresponds to the number of output classes. After compiling the model with appropriate loss and optimization functions, it is trained on the selected features using training data. Finally, the accuracy of the trained model on the training set is evaluated, and this accuracy value is returned as the fitness score for the chromosome. This function is crucial within the GA framework as it guides the selection of features by evaluating their effectiveness in classification tasks.

4. Selection, Crossover, and Mutation

In the Genetic Algorithm (GA) framework for feature selection, the fourth step includes selection, crossover, and mutation operations that lead the population towards better solutions.

The **selection** function normalizes fitness scores to probabilities and using a roulette wheel selection method, selects individuals from the population based on these probabilities. The individuals with higher fitness scores have a higher chance of being selected for reproduction.

An offspring is generated by exchanging genes between parents at a random crossover point in the **crossover** function. This promotes exploration of the search space by combining promising solutions.

By using the probability determined by the mutation rate, the **mutate** function introduces genetic diversity by randomly flipping bits in the chromosome. This prevents premature convergence and allows the algorithm to explore different regions of the solution space. Together, these operations drive the evolution of the population towards optimal solutions for feature selection in the context of emotion

5. GA Iteration

This part directs towards the evolution of the population towards optimal feature subsets for improving emotion classification accuracy.

- Initially, a population of binary chromosomes representing feature subsets is randomly generated.

- Then, for a specified number of iterations (we can define the iteration count), the GA iterates through the selection, crossover, and mutation operations to evolve the population towards better solutions.
-
- At every iteration, the fitness scores for all chromosomes within the population are calculated. This evaluation is conducted using the evaluate_fitness() function, which assesses how accurately each feature subset can classify emotions.
- The chromosome with the highest fitness score is singled out as the best solution.
- Following this, a selection process is employed to pick individuals from the population based on their fitness scores. New offspring are then generated through the crossover and mutation operations.
- This process continues iteratively, with the population gradually evolving towards improved solutions for the feature selection task in emotion classification.

6. Evaluation

The final evaluation of the model is performed after the Genetic Algorithm (GA) process has completed. The **best_chromosome**, representing the optimal feature subset selected by the GA, is used to filter the selected features from the training and testing data. Then, a neural network model is constructed using the Keras Sequential API. The model architecture consists of densely connected layers with specific activation functions. The model's performance is evaluated using the testing data, and the test accuracy is computed. Finally, the accuracy value is printed, providing an assessment of the model's ability to classify emotions based on the selected features. This process concludes the feature selection and evaluation phase, demonstrating the effectiveness of the GA in improving emotion classification accuracy.

Population	**Iteration/Generations**	**Test Accuracy**
2	5	0.94
7	7	0.73
10	7	0.86