electronics

MDPI

Article

Genetic Algorithm for High-Dimensional Emotion Recognition from Speech Signals

Liya Yue [1], Pei Hu [2], Shu-Chuan Chu [3] and Jeng-Shyang Pan [3,4,*]

[1] Fanli Business School, Nanyang Institute of Technology, Nanyang 473004, China
[2] School of Computer and Software, Nanyang Institute of Technology, Nanyang 473004, China
[3] College of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao 266590, China
[4] Department of Information Management, Chaoyang University of Technology, Taichung 413310, Taiwan
* Correspondence: jspan@cc.kuas.edu.tw

Abstract: Feature selection plays a crucial role in establishing an effective speech emotion recognition system. To improve recognition accuracy, people always extract as many features as possible from speech signals. However, this may reduce efficiency. We propose a hybrid filter–wrapper feature selection based on a genetic algorithm specifically designed for high-dimensional (HGA) speech emotion recognition. The algorithm first utilizes Fisher Score and information gain to comprehensively rank acoustic features, and then these features are assigned probabilities for inclusion in subsequent operations according to their ranking. HGA improves population diversity and local search ability by modifying the initial population generation method of genetic algorithm (GA) and introducing adaptive crossover and a new mutation strategy. The proposed algorithm clearly reduces the number of selected features in four common English speech emotion datasets. It is confirmed by K-nearest neighbor and random forest classifiers that it is superior to state-of-the-art algorithms in accuracy, precision, recall, and F1-Score.

Keywords: feature selection; speech emotion recognition; genetic algorithm; high-dimensional

Citation: Yue, L.; Hu, P.; Chu, S.-C.; Pan, J.-S. Genetic Algorithm for High-Dimensional Emotion Recognition from Speech Signals . *Electronics* **2023**, *12*, 4779. https://doi.org/10.3390/electronics12234779

Academic Editors: Alireza Mousavi and Zhengwen Huang

Received: 16 October 2023
Revised: 20 November 2023
Accepted: 23 November 2023
Published: 25 November 2023

1. Introduction

Human communication relies heavily on speech signals [1]. Therefore, speech emotion recognition (SER) poses a compelling challenge in human–computer interaction due to its multifaceted nature, particularly when the recognition is merely based on speech signals [2,3]. Speech carries linguistic information related to emotions, as well as implicit knowledge that can be extracted through speech processing methods [1,4].

SER primarily analyzes audio features without linguistic information to judge a person's emotional state [5,6]. In acoustics, speech processing techniques provide valuable information, which mainly comes from prosodic and spectral features.

Feature fusion improves the classification accuracy of SER systems; nevertheless, it increases the computational cost of classifiers. The reason is that certain features have a significant impact, while others may be completely useless for emotion recognition. Feature selection methods simplify the task of selecting the most relevant features for classification algorithms [7,8]. These methods mainly eliminate the loss and overfitting problems caused by the curse of dimensionality, and improve the model's generalization. Feature selection is the most effective way to enhance the accuracy of SER systems, and it decreases their computation time and memory.

Feature selection reduces the number of features by removing irrelevant and redundant ones [9,10]. However, it is computationally difficult and NP-hard to search the entire feature space. Metaheuristic algorithms provide a robust and flexible approach to solving complex optimization problems [11–13]. Due to their global search ability, adaptability, and potential for parallelization, they are a powerful tool for finding near-optimal solutions in feature

selection [14,15]. Genetic algorithm (GA) is a popular optimization technique inspired by the processes of natural selection and genetics, and it finds available solutions to complex issues by mimicking biological evolution [16,17]. A chromosome represents a potential solution of an optimization problem, and it is encoded as a binary string. Feature selection is a binary optimization, so GA is specifically suited for this task.

In GA, crossover and mutation play pivotal roles, as they determine how the next generation is produced. Researchers create various crossover and mutation operators that are designed to work with specific chromosome representations for optimization problems. Guan et al. introduced the crossover elitist preservation mechanism in which elite solutions are preserved during crossover to promote the retention of valuable genetic material [18]. Faraji and Naji utilized the newly developed crossover architecture that enables parallel crossover operations across multiple individuals within the population [19]. Kaya explained the significant role of crossover operators in GAs and emphasized their importance in facilitating the exploration and exploitation of solution space [20]. Zhang et al. introduced a new crossover mechanism tailored specifically for the Steiner tree problem [21]. This mechanism exploits the problem's characteristics and improves the convergence speed and solution quality of GA. Duan and Zhang proposed a precise mutation strategy that is integrated into GA and particle swarm optimization (PSO) [22]. This mutation enhances the exploration and exploitation of computationally expensive scenarios. Wang et al. divided the population into two groups, and each group was then subjected to a mutation operator with unique properties [23]. They employed the advantages of these mutation operators to enhance the search process's effectiveness and efficiency.

Based on the above analysis, we observe that existing GA algorithms primarily improve their performance through crossover and mutation. However, when dealing with high-dimensional features, they become time-consuming and ineffective. Consequently, we utilize a hybrid filter–wrapper model to address these issues, and the main contributions of this paper are summarized as follows:

1. We propose a novel feature selection based on filter and wrapper methods.
2. We bring an improved GA algorithm with adaptive crossover and novel initialization.
3. We validate the proposed algorithm on four English emotion speech datasets.

The structure of this paper is organized as follows. Section 2 introduces the related works of SER. Section 3 describes the proposed algorithm. Section 4 includes experimental results and discussions, and Section 5 provides the conclusions.

2. Related Works

Human speech often blends a person's emotions with sentence structure and meaning. SER categorizes speakers' emotions by studying their recorded speech. In this section, we will provide an overview of the primary research in the field of SER.

Sun et al. proposed a SER model based on GA [24]. To fully express emotional information, acoustic features are extracted from speech signals. The Fisher Score selects high-ranking features and removes unnecessary information. In the feature fusion stage, GA adaptively searches for the best feature weights. Finally, decision tree (DT) and fused features establish an SER model. Mao et al. proposed a hierarchical SER method based on improved support vector machine (SVM) and DT [25]. DT is established according to the confusion among emotions. In addition, a neural network filters the original features. GA is utilized to select the remaining features for each classification in DT while synchronously optimizing SVM's parameters. Kanwal and Asghar proposed a feature optimization method using cluster-based GA for SER [26]. Instead of randomly selecting a new generation, clustering is utilized during fitness evaluation to detect outliers and exclude them from the next generation. Two mel-frequency cepstral coefficients (MFCCs) improve the performance of SER systems [27]. Several effective feature subsets are determined through a fast correlation-based filter feature selection. Finally, a fuzzy neural network (FAMNN) recognizes emotions from speech. At the same time, GA determines the optimal values of the selection and alert parameters, and the learning rate of FAMNN.

Shahin et al. proposed an automatic SER method based on gray wolf optimizer (GWO) [28]. Speech signal data are processed by feature extraction and then passed to GWO to remove irrelevant and redundant features. Emotion recognition systems that are both accurate and robust can be achieved by employing correlated and meaningful acoustic features. This is because GWO effectively explores feature space and finds the optimal feature set with rich sentiment classification patterns.

Huang and Epps investigated the effectiveness of partitioning speech signals into small segments and extracting acoustic features from each segment [29]. These features acquire specific emotional cues present in different parts of speech and provide a more detailed representation of emotional dynamics in continuous speech. Özseven proposed a novel feature selection method that identifies the most informative and discriminative features from speech signals [30]. The novel feature selection method enhances the accuracy and efficiency of SER systems. Dey et al. combined golden ratio optimization (GRO) and equilibrium optimization (EO) to design a new hybrid metaheuristic feature selection algorithm for SER in audio signals [31]. They use the sequential single point flip (SOPF) technique to search for the nearest neighbors of the final candidate solutions. To address the issue of high-dimensional emotional features in SER, Ding et al. utilized the characteristics of biogeography-based optimization algorithm (BBO) and SVM to obtain features with rich emotional information [32].

Drawing from previous research, we employ a metaheuristic algorithm to choose relevant features in speech emotion recognition. The algorithm aims to enhance classification accuracy by reducing the number of features. The distinct advantages and robust search ability of GA (as explained in Section 1) drive our suggestion to adopt it as a feature selection method in SER.

3. Materials and Methods

The proposed system involves emotional databases, feature extraction, and feature selection.

3.1. Emotional Databases

This study employs eNTERFACE05, the Ryerson audio-visual database of emotional speech and song (RAVDESS), the Surrey audio-visual expressed emotion (SAVEE) database, and the Toronto emotional speech set (TESS) to implement emotion recognition. eNTERFACE05 offers multimodal emotional speech and facial expressions, and RAVDESS provides a diverse set of acted emotions in speech and song. SAVEE contains English speakers expressing various emotions, while TESS focuses on North American English speakers' emotional speech. These databases can complete comprehensive assessments across emotions, patterns, and cultural backgrounds.

1. eNTERFACE05

eNTERFACE05 contains audio recordings of speakers uttering scripted sentences or engaging in spontaneous conversations while expressing different emotions. These emotional states include happiness, sadness, anger, and neutral, among others.

2. RAVDESS

The database consists of 24 skilled actors who deliver statements with the same words but varying emotions, including anger, fear, calm, happiness, sadness, surprise, and disgust. There are two different levels of intensity for each emotion, as well as an additional neutral expression.

3. SAVEE

SAVEE contains recordings of emotional speech from actors portraying anger, happiness, sadness, fear, disgust, surprise, and neutral expressions. Each emotion is recorded in different intensity levels. The dataset contains recordings of four male native English speakers, one of whom is a postgraduate student, while the rest are researchers at the University of Surrey.

4. TESS

TESS contains a diverse range of emotional expressions, including but not limited to emotions like happiness, sadness, anger, fear, disgust, and surprise. Each recording is labeled with the corresponding emotional category, so it is suitable for supervised machine learning tasks.

3.2. Feature Extraction

Pre-emphasis, framing, and windowing are fundamental signal processing techniques commonly used in speech and audio analysis. The Hamming window splits speech signals into frames of 25 ms with an overlap of 10 ms, and the pre-emphasis filter coefficient is set to 0.97. These techniques work together to enhance the quality of audio signals and provide a more suitable representation for subsequent processing steps. Specifically, pre-emphasis boosts the higher frequencies; framing segments signals for analysis; windowing minimizes spectral leakage.

In this study, we employ the OpenSmile toolkit to gather acoustic features, which are part of the standard set used in the computational paralinguistics challenge at INTERSPEECH 2010. A total of 1582 features are extracted from this toolbox, and Table 1 provides further details.

Table 1. Summary of acoustic features.

Acoustic Features	**Numbers**
Loudness	42
MFCC coefficients	630
Logarithmic power of mel-frequency bands	336
Line spectral pair frequencies	336
Envelope of smoothed fundamental frequency contour	42
Voicing probability of the final fundamental frequency candidate	42
Smoothed fundamental frequency contour	40
Local jitter	38
Differential jitter	38
Local shimmer	38

3.3. Improved Genetic Algorithm for Feature Selection

In the proposed algorithm, as shown in Figure 1, we first rank acoustic features from Fisher Score and information gain, and then the feature space is divided according to the ranking. Finally, the genetic algorithm for high dimensionality (HGA) implements feature selection on classifiers.

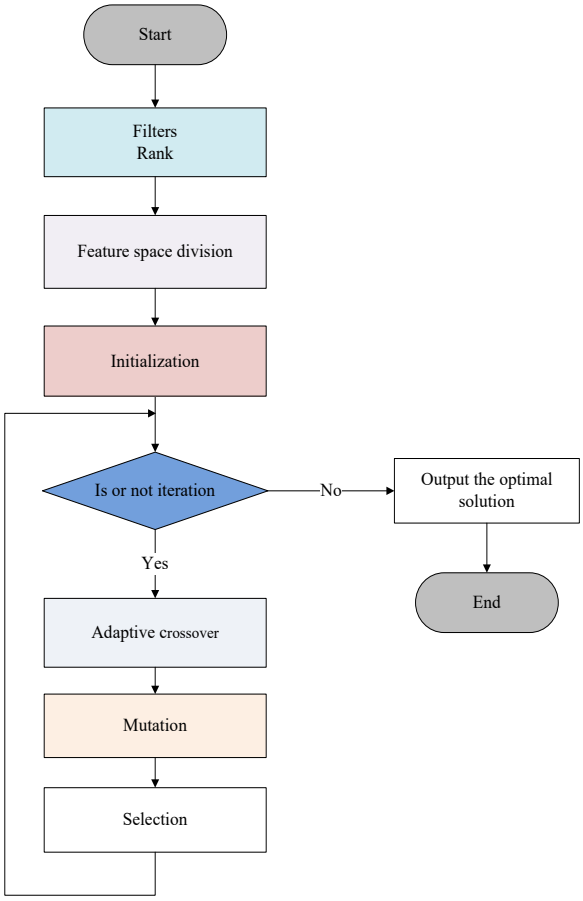

Figure 1. The flowchart of the proposed algorithm.

3.3.1. Filters

1. Fisher Score

Fisher Score, also known as Fisher discriminant analysis or Fisher's linear discriminant, is a statistical method in machine learning and pattern recognition. It finds a linear combination of features in a dataset which maximizes the separation among different classes.

Suppose n_i is the number of samples contained in the i-th class, and u_i^r and $(\sigma_i^r)^2$ are the mean and variance of the r-th feature of the i-th class. The Fisher Score of the r-th feature is calculated as follows:

$$F_r = \frac{\sum_{i=1}^{C} n_i(u_i^r - u_r)}{\sum_{i=1}^{C} n_i(\sigma_i^r)^2} \tag{1}$$

It is important to minimize the variation in a feature when considering data samples from the same category. On the other hand, we aim to maximize the variation when comparing a feature across different categories.

2. Information gain

Information gain calculates the reduction in uncertainty or entropy achieved by considering a particular feature in a dataset. The information entropy of a random variable X is computed in the following manner:

$$H(X) = -\sum_{x} p(x) log p(x) \tag{2}$$

where $p(x)$ represents the probability distribution of X. The joint entropy $H(X,Y)$ of two random variables, X and Y, is defined as follows:

$$H(X,Y) = -\sum_{x}\sum_{y} p(x,y) log p(x,y) \tag{3}$$

where $p(x,y)$ represents the probability distribution of X and Y. Equation (4) is the conditional entropy $H(X|Y)$.

$$H(X|Y) = H(X,Y) - H(Y) \tag{4}$$

Features with higher information gain are more influential in separating and categorizing data. Fisher Score and information gain rank features. Subsequently, the TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) method is used for a comprehensive evaluation of each feature.

3.3.2. Feature Space Division

According to the filtering ranking of features, the whole feature space is divided into four parts, as shown in Figure 2.

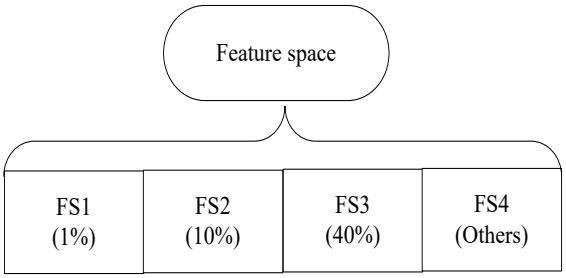

Figure 2. Feature space division.

The top-ranking features (FS1) make up 1% of the space. These features are the most important, so they have the highest probability of being selected. The probabilities of features in other spaces gradually decrease, as shown in Equation (5):

$$r(i) = \begin{cases} 0.9 & if(i \in FS1) \\ 0.6 & if(i \in FS2) \\ 0.1 & if(i \in FS3) \\ 0.01 & if(i \in FS4) \end{cases} \tag{5}$$

3.3.3. Improved Genetic Algorithm

Crossover and mutation play important roles in GA, as they involve mixing and matching genes to create new generations of potential solutions. They expand the scope and explore various possibilities for solving a problem. While many studies have been conducted in this area, it is still a challenge to determine which crossover and mutation methods to use in a given situation. In this study, our objective is to provide insights into selecting the appropriate method for emotion recognition problems, and we improve GA from initialization, crossover, and mutation.

1. Initialization

When randomizing the initial population, features with high importance are more likely to be selected, while those with a low ranking have a low probability of being chosen. The initialization of HGA relies on the feature ranking, as depicted in Equation (6).

$$X_i^j = \begin{cases} 1 & if(r(i) > rand) \\ 0 \end{cases} \tag{6}$$

where X_i^j represents the position of chromosome i in the j-th dimension.

2. Crossover

Crossover is an evolutionary method that utilizes two or more solutions to generate a new one. Original solutions are called parents, and new individuals are known as children. One of the simplest crossover methods is the 1-point crossover. It selects a cut-off point along the parents' genetic representation and swaps the segments before and after this point to produce two children. Each child inherits at least one element from their parents. Multiple cut-off points can be produced in the same manner. Instead of choosing cut-off points, it is possible to determine the probability of swapping several of the parents' elements.

Individuals in the population are randomly paired, and excellent individuals act as parents to generate offspring through crossover. An adaptive crossover operator is utilized in HGA. The features in FS1, FS3, and F4 have a higher/lower probability of being selected. FS2 is an area that requires exploration; therefore, a multi-point crossover strategy is adopted to expand search range and increase population diversity. As indicated in line 3 of Algorithm 1, the crossover rate of FS2 decreases with the execution of the algorithm. To illustrate this concept, let us consider the following example, shown in Figure 3. In FS1 and FS3, there are no crossover chromosomes in parent1 and parent2. In FS2 and FS4, parents adopt 2- and 1-point crossovers, respectively.

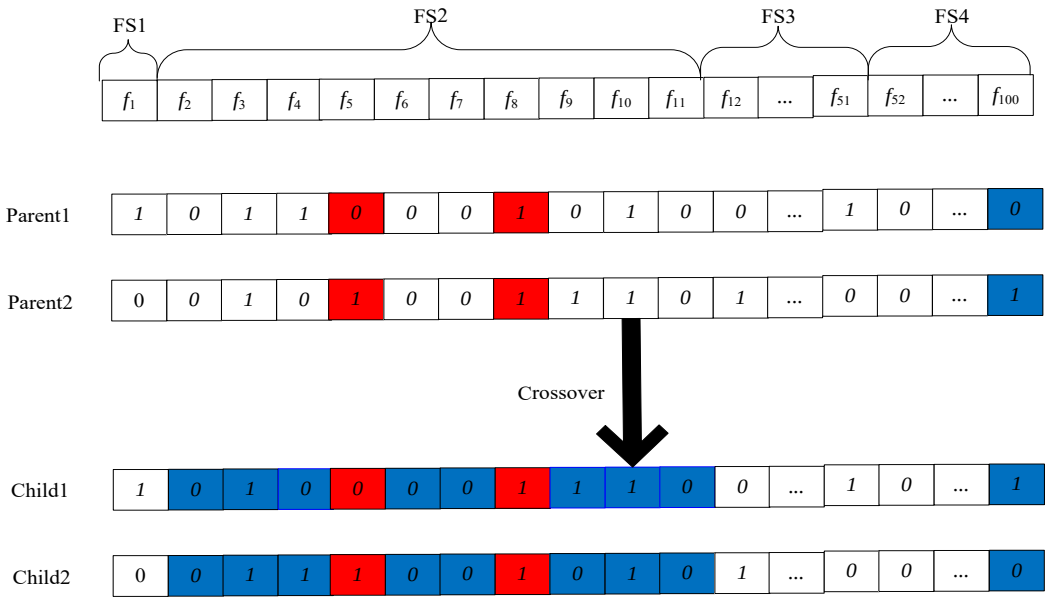

Figure 3. The example of adaptive crossover.

3. Mutation

Mutation changes a single individual in a population. It is frequently utilized for local search or breaking local maxima through further improving a promising solution. Minor adjustments to candidate solutions are employed to produce individuals that are slightly different from their parents. The number of mutations is controlled by Equation (5). Features in FS1, FS3, and FS4 have a large/small probability of being selected, resulting in small mutation. Mutation occurs at the same rate as the original GA for features in FS2. Algorithm 2 describes the detailed process of the mutation.

Algorithm 1: Adaptive crossover

```
dim = length(x1);
[~, index] = sort(r);
x11 = x1(index(1:dim));
x21 = x2(index(1:dim));
y1 = x11;
y2 = x21;
y3 = zeros(1,dim);
y4 = zeros(1,dim);
split1 = round(0.01 × dim);
split2 = round(0.1 × dim);
split3 = round(0.4 × dim);
if rand() > 0.7 then
    t = randi([1 split1]);
    y1(1:split1) = [x11(1:t) x21(t + 1:split1)];
    y2(1:split1) = [x21(1:t) x11(t + 1:split1)];
end
m = it/MaxIt;
mp = round(split2*(1 − m) / 4);
t = randi([split1 + 1 split2 − 1],1,mp);
[t1,index2] = sort([t split2]);
a = x11(split1 + 1:t1(1));
b = x21(split1 + 1:t1(1));
for i = 2 : length(t1) do
    starti = t1(i − 1) + 1;
    endi = t1(i);
    if mod(i, 1) == 0 then
        a = [a x21(starti:endi)];
        b = [b x11(starti:endi)];
    end
    else
        a = [a x11(starti:endi)];
        b = [b x21(starti:endi)];
    end
end
y1(split1 + 1:split2) = a;
y2(split1 + 1:split2) = b;
if rand() > 0.7 then
    t = randi([split2 + 1 split3]);
    y1(split2 + 1:split3) = [x11(split2 + 1:t) x21(t + 1:split3)];
    y2(split2 + 1:split3) = [x21(split2 + 1:t) x11(t + 1:split3)];
end
if rand() > 0.7 then
    t = randi([split3 + 1 dim]);
    y1(split3 + 1:dim) = [x11(split3 + 1:t) x21(t + 1:dim)];
    y2(split3 + 1:dim) = [x21(split3 + 1:t) x11(t + 1:dim)];
end
for i = 1 : dim do
    y3(i) = y1(index(i));
    y4(i) = y2(index(i));
end
```

Algorithm 2: Mutation

```
for j = 1 : dim do
    if if(r(j) == 0.9) then
        if rand() < 0.3 × mu then
            x(j) = 1 − x(j);
        end
    end
    if if(r(j) == 0.6) then
        if rand() < mu then
            x(j) = 1 − x(j);
        end
    end
    if if(r(j) == 0.1) then
        if rand() < 0.3 × mu then
            x(j) = 1 − x(j);
        end
    end
    if if(r(j) == 0.01) then
        if rand() < 0.01 × mu then
            x(j) = 1 − x(j);
        end
    end
end
```

4. Experimental Results and Analysis

To validate the superiority of the proposed HGA, the classification performance is compared with GA [24], GWO [28], and JAYA [33]. JAYA first uses ReliefF, correlation, ANOVA (Analysis of Variance), information gain, and information gain rate to sort features, and then employs TOPSIS to extract the top 10% of features to implement feature selection. Table 2 offers the more details regarding the algorithms.

Table 2. The main parameter settings.

Algorithms	Main Parameters
GA	beta = 1; pC = 1; mu = 0.02;
GWO	a = 2;
JAYA	lb = 0; ub = 1; thres = 0.5; top 10% features;
HGA	beta = 1; pC = 1; mu = 0.02;

The maximum number of iterations for the algorithms is set to 100, with 20 runs, and the population size is 20. The Wilcoxon rank sum test and Frideman test are used to determine if there are any significant differences in the experimental results obtained. The significant level is selected as 0.05. If the *p*-value is less than or equal to 0.05, it represents that an approach significantly outperforms other approaches with 95% confidence.

4.1. Objective Function

In feature selection and SER, classification accuracy is the main indicator for evaluating algorithms, so it is used as the objective function in the experiments, as shown in Equation (7). We also compare the algorithms in precision, recall, F1-Score, the number of selected features, and running time.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{7}$$

$$precision = \frac{TP}{TP + FP} \tag{8}$$

$$recall = \frac{TP}{TP + FN} \tag{9}$$

$$F1\text{-}Score = \frac{2 * TP}{2 * TP + FP + FN} \tag{10}$$

4.2. Experimental Analysis

In this research, K-nearest neighbor (KNN) and random forest (RF) are utilized to establish classification models in which the K is set to 5 and RF is constructed by 20 decision trees, and 10-fold cross validation is adopted to evaluate the performance of the models.

4.2.1. Simulation Results on the KNN Classifier

Table 3 presents recognition accuracy (*Accuracy*) and the number of selected features (*Length*) using the KNN classifier. From recognition accuracy, HGA outperforms GA in eNTERFACE05, RAVDESS, and TESS, but it is inferior to GA in SAVEE. Through the Wilcoxon rank sum test, it is found that they have similar experimental statistical data in eNTERFACE05. The experimental results illustrate that HGA is superior to GA, and the adaptive crossover and improved mutation help to advance speech emotion recognition. GWO and JAYA have no similar experimental results with HGA, and they perform worse than HGA. Although JAYA uses filter methods, it discards low-ranking features. The experimental results indicate that low-ranking features also play an important part in recognition, and the strategy of comprehensively considering features proposed in this paper is effective. The Friedman test reveals that their average ranks are 1.75, 3.25, 3.75, and 1.25, respectively, with p-values less than 0.05. The Wilcoxon rank sum test and the Friedman test validate the superiority of HGA.

Table 3. The experimental results of the compared algorithms based on KNN.

Datasets	GA		GWO		JAYA		HGA	
	Accuracy	**Length**	**Accuracy**	**Length**	**Accuracy**	**Length**	**Accuracy**	**Length**
eNTERFACE05	0.4504	706.9	0.3632	1072.6	0.3674	35.9	**0.4634**	161.05
RAVDESS	0.4958	815.7	0.3708	1147.2	0.2741	59.5	**0.5123**	193.5
SAVEE	**0.5507**	763	0.5058	1115.3	0.4904	56.7	0.5364	181.8
TESS	0.8565	746.7	0.8270	1134.3	0.8002	94.8	**0.9748**	223.8
>/=/<	1/1/2		0/0/4		0/0/4		3/0/1	
Rank	1.75		3.25		3.75		1.25	
p-Value	1.69×10^{-2}							

JAYA only selects features from the top 10%, so it has the minimum number of features. In contrast, GA and GWO implement selection on all features, and they have the highest number of features. Although HGA operates on all features, it avoids selecting many low-ranking features. HGA effectively balances emotional recognition accuracy and the number of selected features.

Figure 4 displays the precision, recall, and F1-Score of the algorithms. The algorithms have the best data in TESS. GA, GWO, and HGA have the worst performance in eNTERFACE05, and JAYA performs the worst in RAVDESS. In SAVEE, GA outperforms other algorithms, and HGA has obvious advantages in eNTERFACE05, SAVEE, and TESS. HGA offers flexibility and robustness in SER and decreases feature space.

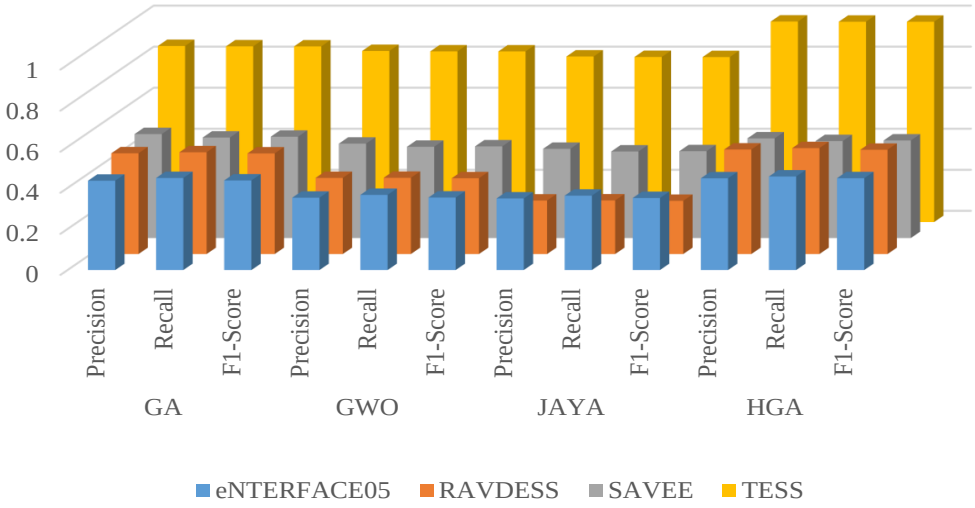

Figure 4. The precision, recall, and F1-Score of the algorithms based on KNN.

Table 4 presents the running time of the algorithms. The execution time of feature selection mainly depends on classifiers. The maximum time complexity of KNN is $O(D * N * N)$, where D represents the number of samples and N is the number of features used. Since JAYA uses the minimum number of features, it has the highest computational efficiency. HGA has better efficiency than GA and GWO. TESS contains a large number of samples, so the algorithms run on it for the longest time.

Table 4. The average running time of the compared algorithms (in seconds) based on KNN.

Datasets	GA	GWO	JAYA	HGA
eNTERFACE05	755.2	1622.5	**187.6**	462.8
RAVDESS	1969.8	5279.1	**449.8**	1560.4
SAVEE	794.0	2156.9	**201.1**	1044.3
TESS	5524.9	17391	**1269.6**	4492.9

Based on the previous discussion, it is evident that HGA presents exceptional performance in terms of classification accuracy, precision, recall, F1-Score, and the number of selected features. Therefore, HGA is a highly appropriate choice for speech emotion recognition.

4.2.2. Simulation Results on the RF Classifier

Table 5 displays recognition accuracy and the number of selected features using the RF classifier. The accuracy achieved with RF surpasses the value obtained by KNN. The results in Table 5 highlight the superior performance of HGA in eNTERFACE05, RAVDESS, SAVEE, and TESS, demonstrating its superiority over GA, GWO, and JAYA. According to the Wilcoxon rank sum test, GA, GWO, JAYA, and HGA excel in three, three, zero, and four datasets, respectively. GA and GWO have similar statistical results to HGA in eNTERFACE05, RAVDESS, and SAVEE. JAYA employs the minimum number of features to complete classification, followed by HGA, GA, and GWO. The results of the Friedman test indicate that the average ranks of GA, GWO, JAYA, and HGA are two, three, four, and one, respectively, with a p-value of 7.38×10^{-3}. This analysis is further proved by the data in Table 5, which clearly reveals the superior performance of HGA in speech emotion recognition.

Table 5. The experimental results of the compared algorithms based on RF.

Datasets	GA		GWO		JAYA		HGA	
	Accuracy	**Length**	**Accuracy**	**Length**	**Accuracy**	**Length**	**Accuracy**	**Length**
eNTERFACE05	0.5153	742.75	0.5152	1088.3	0.4579	96.5	**0.5172**	177.3
RAVDESS	0.5521	785.4	0.5488	1164.6	0.4750	105.5	**0.5612**	190.7
SAVEE	0.6533	729.3	0.6451	1110.8	0.6314	94.25	**0.6575**	184
TESS	0.9905	742.8	0.9888	1155.65	0.9589	112.2	**0.9931**	181.8
>/=/<	0/3/1		0/3/1		0/0/4		4/0/0	
Rank	2		3		4		1	
p-Value	7.38×10^{-3}							

Figure 5 describes the precision, recall, and F1-Score of the algorithms. In terms of these metrics, the algorithms achieve their highest scores in TESS, followed by SAVEE, RAVDESS, and eNTERFACE05. Notably, HGA excels in RAVDESS and SAVEE, while GWO outperforms the other algorithms in eNTERFACE05 and TESS. Figure 5 effectively illustrates that HGA is adept at selecting the most pertinent features from the speech emotion datasets, and it achieves a harmonious balance between precision and recall.

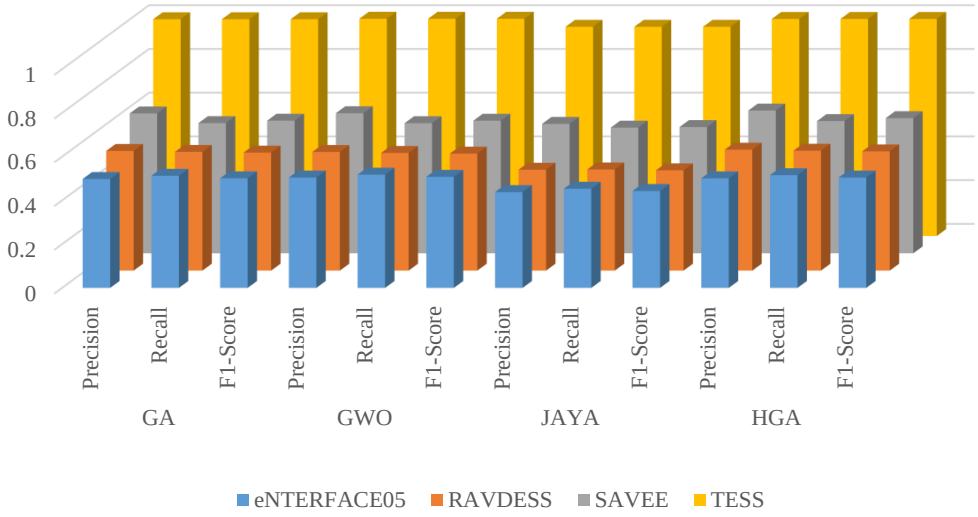

Figure 5. The precision, recall, and F1-Score of the algorithms based on RF.

Table 6 depicts the running time of the algorithms. The maximum time complexity of RF is $O(M * (D * N * logD))$, where M is the number of decision tress. The execution time of the algorithms with the RF classifier is significantly longer compared to that with KNN, primarily due to the higher time complexity of RF. JAYA demonstrates the quickest execution efficiency, followed by HGA, GA, and GWO. RAVDESS and TESS display longer running time, while eNTERFACE05 and SAVEE require relatively less time for execution.

Table 6. The average running time of the compared algorithms (in seconds) based on RF.

Datasets	GA	GWO	JAYA	HGA
eNTERFACE05	6306.0	6260.3	6290.9	**5990.6**
RAVDESS	21,050.8	21,124.5	21,175.2	**19,343.1**
SAVEE	7135.5	7031.1	7117.0	**6611.9**
TESS	25,160.6	26,345.8	24,241.2	**22,994.6**

HGA is unequivocally the top-performing method based on the experimental data of the algorithms using KNN and RF classifiers. The novel hybrid filter–wrapper technique

employs a few features for classification. The adaptive crossover strategy enhances population diversity and retains the potential to discover the optimal solution. The empirical evidence derived from HGA confirms its suitability for emotion recognition.

5. Conclusions

Emotion-related features are always extracted from speech signals. People attempt to use more features for recognition due to uncertainty about which features are effective for classification, causing high-dimensional problems. We investigate feature selection based on filter and wrapper methods. Fisher Score and information gain rank features. However, unlike traditional filter methods, low-ranking features also enter the wrapper operation. An improved GA is proposed to effectively search for the optimal solution, and the performance of the algorithm is tested on four different datasets using KNN and RF classifiers. HGA is superior to the compared algorithms in terms of accuracy, precision, recall, and F1-Score. It acquires an accuracy of 0.4634, 0.5123, 0.5364, and 0.9748 on eNTERFACE05, RAVDESS, SAVEE, and TESS based on KNN and an accuracy of 0.5172, 0.5612, 0.6575, and 0.9931 based on RF. The future research work for the proposed algorithm involves exploring its adaptability to real-world scenarios, enhancing its robustness in diverse cultural and linguistic contexts, and integrating it into practical applications such as human–computer interaction, mental health monitoring, and personalized services.

Author Contributions: Conceptualization, L.Y. and P.H.; Formal analysis, L.Y. and S.-C.C.; Methodology, L.Y., S.-C.C. and J.-S.P.; Software, L.Y. and P.H.; Writing—original draft, L.Y.; Writing—review and editing, P.H., S.-C.C. and J.-S.P. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the Henan Provincial Philosophy and Social Science Planning Project (2022BJJ076), and the Henan Province Key Research and Development and Promotion Special Project (Soft Science Research) (222400410105).

Data Availability Statement: Data are available on request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Liu, X.; Wang, X.; Sahidullah, M.; Patino, J.; Delgado, H.; Kinnunen, T.; Todisco, M.; Yamagishi, J.; Evans, N.; Nautsch, A.; et al. Asvspoof 2021: Towards spoofed and deepfake speech detection in the wild. *IEEE/Acm Trans. Audio Speech Lang. Process.* **2023**, *31*, 2507–2522. [CrossRef]
2. Sánchez-Rada, J.F.; Araque, O.; Iglesias, C.A. Senpy: A framework for semantic sentiment and emotion analysis services. *Knowl.-Based Syst.* **2020**, *190*, 105193. [CrossRef]
3. Makhmudov, F.; Kutlimuratov, A.; Akhmedov, F.; Abdallah, M.S.; Cho, Y.I. Modeling Speech Emotion Recognition via Attention-Oriented Parallel CNN Encoders. *Electronics* **2022**, *11*, 4047. [CrossRef]
4. Han, T.; Zhang, Z.; Ren, M.; Dong, C.; Jiang, X.; Zhuang, Q. Speech Emotion Recognition Based on Deep Residual Shrinkage Network. *Electronics* **2023**, *12*, 2512. [CrossRef]
5. Kim, S.; Lee, S.P. A BiLSTM–Transformer and 2D CNN Architecture for Emotion Recognition from Speech. *Electronics* **2023**, *12*, 4034. [CrossRef]
6. Baek, J.Y.; Lee, S.P. Enhanced Speech Emotion Recognition Using DCGAN-Based Data Augmentation. *Electronics* **2023**, *12*, 3966. [CrossRef]
7. Chen, Y.; Ye, Z.; Gao, B.; Wu, Y.; Yan, X.; Liao, X. A Robust Adaptive Hierarchical Learning Crow Search Algorithm for Feature Selection. *Electronics* **2023**, *12*, 3123. [CrossRef]
8. Yue, L.; Hu, P.; Chu, S.C.; Pan, J.S. English Speech Emotion Classification Based on Multi-Objective Differential Evolution. *Appl. Sci.* **2023**, *13*, 12262. [CrossRef]
9. Sowan, B.; Eshtay, M.; Dahal, K.; Qattous, H.; Zhang, L. Hybrid PSO feature selection-based association classification approach for breast cancer detection. *Neural Comput. Appl.* **2023**, *35*, 5291–5317. [CrossRef]
10. Fahy, C.; Yang, S. Dynamic Feature Selection for Clustering High Dimensional Data Streams. *IEEE Access* **2019**, *7*, 127128–127140. [CrossRef]
11. Ma, L.; Liu, Y.; Yu, G.; Wang, X.; Mo, H.; Wang, G.G.; Jin, Y.; Tan, Y. Decomposition-Based Multiobjective Optimization for Variable-Length Mixed-Variable Pareto Optimization and Its Application in Cloud Service Allocation. *IEEE Trans. Syst. Man Cybern. Syst.* **2023**, *53*, 7138–7151. [CrossRef]

12. Zhao, X.; Zhao, Y.; You, L.; Liu, Z.; Xuan, H.; Li, Y. Multi-Strategy Particle Swarm Optimization based MOEA/D for VNF-SC Deployment. *J. Netw. Intell.* **2021**, *6*, 741–752.
13. Chen, X.; Sun, Y.; Zhang, M.; Peng, D. Evolving deep convolutional variational autoencoders for image classification. *IEEE Trans. Evol. Comput.* **2020**, *25*, 815–829. [CrossRef]
14. Tian, Y.; Si, L.; Zhang, X.; Cheng, R.; He, C.; Tan, K.C.; Jin, Y. Evolutionary large-scale multi-objective optimization: A survey. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–34. [CrossRef]
15. Bacanin, N.; Stoean, C.; Zivkovic, M.; Jovanovic, D.; Antonijevic, M.; Mladenovic, D. Multi-swarm algorithm for extreme learning machine optimization. *Sensors* **2022**, *22*, 4204. [CrossRef]
16. Chowdhury, S.; Katangur, A.; Sheta, A.; Psayadala, N.R.; Liu, S. Genetic Algorithm Based Service Broker Policy to find Optimal Datacenters in Cloud Services. In Proceedings of the 2023 8th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA), Chengdu, China, 26–28 April 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 270–278.
17. Li, Y.S.; Pai, P.F.; Lin, Y.L. Forecasting inflation rates be extreme gradient boosting with the genetic algorithm. *J. Ambient Intell. Humaniz. Comput.* **2023**, *14*, 2211–2220. [CrossRef]
18. Guan, B.; Zhang, C.; Ning, J. Genetic algorithm with a crossover elitist preservation mechanism for protein–ligand docking. *Amb Express* **2017**, *7*, 174. [CrossRef] [PubMed]
19. Faraji, R.; Naji, H.R. An efficient crossover architecture for hardware parallel implementation of genetic algorithm. *Neurocomputing* **2014**, *128*, 316–327. [CrossRef]
20. Kaya, M. The effects of two new crossover operators on genetic algorithm performance. *Appl. Soft Comput.* **2011**, *11*, 881–890. [CrossRef]
21. Zhang, Q.; Yang, S.; Liu, M.; Liu, J.; Jiang, L. A new crossover mechanism for genetic algorithms for Steiner tree optimization. *IEEE Trans. Cybern.* **2020**, *52*, 3147–3158. [CrossRef] [PubMed]
22. Duan, X.; Zhang, X. A hybrid genetic-particle swarm optimizer using precise mutation strategy for computationally expensive problems. *Appl. Intell.* **2021**, *52*, 8510–8533. [CrossRef]
23. Wang, F.; Xu, G.; Wang, M. An improved genetic algorithm for constrained optimization problems. *IEEE Access* **2023**, *11*, 10032–10044. [CrossRef]
24. Sun, L.; Li, Q.; Fu, S.; Li, P. Speech emotion recognition based on genetic algorithm–decision tree fusion of deep and acoustic features. *ETRI J.* **2022**, *44*, 462–475. [CrossRef]
25. Mao, Q.; Wang, X.; Zhan, Y. Speech emotion recognition method based on improved decision tree and layered feature selection. *Int. J. Humanoid Robot.* **2010**, *7*, 245–261. [CrossRef]
26. Kanwal, S.; Asghar, S. Speech emotion recognition using clustering based GA-optimized feature set. *IEEE Access* **2021**, *9*, 125830–125842. [CrossRef]
27. Gharavian, D.; Sheikhan, M.; Nazerieh, A.; Garoucy, S. Speech emotion recognition using FCBF feature selection method and GA-optimized fuzzy ARTMAP neural network. *Neural Comput. Appl.* **2012**, *21*, 2115–2126. [CrossRef]
28. Shahin, I.; Alomari, O.A.; Nassif, A.B.; Afyouni, I.; Hashem, I.A.; Elnagar, A. An efficient feature selection method for arabic and english speech emotion recognition using Grey Wolf Optimizer. *Appl. Acoust.* **2023**, *205*, 109279. [CrossRef]
29. Huang, Z.; Epps, J. An investigation of partition-based and phonetically-aware acoustic features for continuous emotion prediction from speech. *IEEE Trans. Affect. Comput.* **2018**, *11*, 653–668. [CrossRef]
30. Özseven, T. A novel feature selection method for speech emotion recognition. *Appl. Acoust.* **2019**, *146*, 320–326. [CrossRef]
31. Dey, A.; Chattopadhyay, S.; Singh, P.K.; Ahmadian, A.; Ferrara, M.; Sarkar, R. A hybrid meta-heuristic feature selection method using golden ratio and equilibrium optimization algorithms for speech emotion recognition. *IEEE Access* **2020**, *8*, 200953–200970. [CrossRef]
32. Ding, N.; Ye, N.; Huang, H.; Wang, R.; Malekian, R. Speech emotion features selection based on BBO-SVM. In Proceedings of the 2018 Tenth International Conference on Advanced Computational Intelligence (ICACI), Xiamen, China, 29–31 March 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 210–216.
33. Chaudhuri, A.; Sahu, T.P. A hybrid feature selection method based on Binary Jaya algorithm for micro-array data classification. *Comput. Electr. Eng.* **2021**, *90*, 106963. [CrossRef]