A Fusion Approach of Firefly Algorithm and Adaptive Network based Fuzzy Inference System for Speaker Recognition

Samiya SILARBI
SIMPA Laboratory, University of Sciences and Technology Oran Mohammed Boudiaf, Algeria
dr.samiya.silarbi@gmail.com

Abstract

This paper introduces an evolutionary approach for training the adaptive network-based fuzzy inference system (ANFIS) in the field of speaker recognition. In contrast to previous methods that rely on gradient descent (GD), which often suffer from slow convergence and suboptimal local minima, this study employs Firefly Algorithm (FA), a swarm intelligence technique. FA is utilized to optimize the premise parameters of the rules, while the conclusion part is optimized using least-squares estimation (LSE).

To assess the effectiveness of the proposed FA-ANFIS model, experiments are conducted using the CHAINS speech dataset for speaker recognition. The results obtained from the hybrid model demonstrate a significant improvement in accuracy when compared to similar ANFIS models optimized using gradient descent. Overall, the integration of FA into the ANFIS framework yields promising outcomes, showcasing its potential for enhancing speaker recognition accuracy. The findings highlight the effectiveness of the FA-ANFIS hybrid model as an alternative optimization technique for training ANFIS in speaker recognition applications.

Keywords: *ANFIS, FA, FA-ANFIS, Speaker Recognition.*

---

1. Introduction

Speaker recognition is the process of automatically identifying a person based on their voice, utilizing speaker-specific information contained in speech waves [1][2]. This technology is widely used in real-world applications such as access controls, telephone applications, PC logins, and door control systems [3]. Despite extensive research has been conducted in this field, current approaches do not yet match the speed and accuracy of human recognition capabilities.

Human perception is highly adept at recognizing familiar voices and distinguishing between individuals based on subtle vocal cues and patterns. Significant strides have been made in speaker recognition systems due to advancements in machine learning, signal processing, and pattern recognition. Cutting-edge techniques, including convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have been successfully employed to enhance feature extraction and augment accuracy in speaker recognition. Continued research

focuses on refining feature extraction techniques, developing novel deep learning architectures, and incorporating additional contextual or multi-modal information **[4]**. Although current speaker recognition approaches fall short of human capabilities, ongoing advancements aim to bridge the gap and enhance the efficiency and accuracy of speaker recognition systems.

Neuro-fuzzy modeling offers an alternative approach to speaker recognition by combining the strengths of neural networks and fuzzy logic **[5]**. This approach aims to leverage the discriminative power of neural networks while incorporating the reasoning and deduction abilities of fuzzy logic. The model is trained as a neural network but incorporates a linguistic interpretation of variables using fuzzy logic.

Neuro-fuzzy models encode information simultaneously and distribute the architecture within a numerical framework. Different architectures have been proposed, such as Mamdani's or Sugeno's **[6] [7]**, depending on the type of rules included in the model. One influential fuzzy model is the Adaptive Network-based Fuzzy Inference System (ANFIS) proposed by Robert Jang **[8]**. ANFIS has been widely used in different domains, including speaker recognition **[9] [10]**. The rule base of ANFIS consists of fuzzy "if-then" rules of the Takagi and Sugeno's type, where the conclusion parts are represented as linear functions of the inputs instead of fuzzy sets. This approach reduces the number of required fuzzy rules and simplifies the modeling process. Neuro-fuzzy modeling has shown promise in improving speaker recognition performance. By combining the strengths of neural networks and fuzzy logic, these models can effectively capture complex patterns in speech signals and make accurate speaker recognition decisions.

Fuzzy model building involves two crucial phases: structure identification and parameter optimization. In the structure identification phase, the number of fuzzy if-then rules and the membership functions of the premise fuzzy sets are determined. This phase establishes the foundation of the fuzzy model.

The optimization of these parameters is one of the main challenges in ANFIS training. Classical learning methods, such as gradient descent, are commonly used but have the disadvantage of getting trapped in poor local minima. This limitation arises from the algorithm's reliance on a reduced search space around an initial random solution, which may not be suitable. To overcome this limitation, researchers have explored alternative techniques for parameter optimization in ANFIS models. One approach involves utilizing metaheuristic algorithms such as genetic algorithms, particle swarm optimization, or simulated annealing... These algorithms offer the advantage of exploring a wider search space and mitigating the risk of converging to suboptimal solutions.

Hybrid approaches that combine gradient descent with metaheuristic methods have also been proposed. By leveraging the strengths of both approaches, these hybrid algorithms aim to enhance the optimization process and improve the quality of parameter configurations. The optimization of parameters in ANFIS models is an active research area. Researchers aim to develop more robust and efficient algorithms that can overcome the limitations associated with local minima. The goal is to enhance the accuracy and reliability of fuzzy models used in speaker recognition and other applications.

In this study, we present an alternative training approach for optimizing ANFIS parameters more efficiently compared to the gradient method. Our proposed approach utilizes Firefly

Algorithm (FA).FA is an algorithm that explores a larger search space by utilizing multiple initial solutions, known as swarms. While FA is typically time-consuming, requiring a larger number of swarms and iterations based on the number of parameters to be optimized, we aim to reduce the training cost by applying FA only to the premise part of the rules. For the conclusion part, we employ a least square estimation (LSE) approach. To evaluate the effectiveness of our proposed learning algorithm, we conduct experiments using the CHAINS dataset for speaker recognition. We compare the results obtained from our proposed FA-based ANFIS with those achieved by the traditional ANFIS trained using the gradient approach.

The incorporation of FA for parameter optimization in ANFIS is anticipated to enhance the training process's efficiency and convergence. Our primary goal is to attain speaker recognition results that are more precise and dependable while minimizing the computational expenses related to training.

The remaining sections of the paper are structured as follows: Section 2 provides a comprehensive review of the relevant literature pertaining to speaker recognition. In section 3, we present a detailed explanation of the ANFIS model and the FA algorithm. We then outline the process of developing the FA-ANFIS model specifically for speaker recognition. While section 4 presents the experimental results obtained from our study and provides a thorough discussion of these results. Finally, in Section 5, we conclude the paper, summarizing the key findings and contributions of our research on speaker recognition using the FA-ANFIS model.

2. Related work

Automatic Speaker Recognition has witnessed significant advancements in recent decades, with numerous studies proposing different mathematical approaches to improve recognition rates. For instance, **[11]** employed Hidden Markov Model (HMM) for identifying Arabic speakers using Mel Frequency Cepstral Coefficients (MFCCs) **[12]** to represent the speech signal. The experiments achieved a 100% identification rate for text-dependent scenarios and 80% for text-independent scenarios.

[13] Utilized formants and wavelet packet entropy as inputs to a neural network for classification. They reported recognition rates of 90.09% for vowel-dependent experiments and 82.50% for vowel-independent experiments. **[14]** Employed the Generalized Regression Neural Network (GRNN) and Back-Propagation Neural Network (BPNN) as classifiers to classify Chinese speakers. By employing the empirical mode decomposition (EMD) feature extraction method, they achieved a recognition rate of 78.16% and 88.82% with the respective methods.

[15] Utilized an artificial neural network (NN) model for classification. They extracted MFCC features from the speech signal, reduced the dimensionality of the input eigenvector using the K-mean Linde-Buzo-Gray algorithm, and achieved a training network error of 0.015 with 950 hidden layers. **[16]** Proposed an NN framework for text-independent speaker classification and verification using the TIMIT 8K database. They generated 39 MFCCs from preprocessed speech and achieved 100% classification accuracy. **[17]** Employed an ANN classifier with Back Propagation (BPNN) and extracted sixteen MFCC features from 50

users. They achieved identification accuracies ranging from 92% to 70% for 10 to 50 users, respectively.

[18] Proposed a multiclass SVM-based speaker clustering method using feature vectors composed of 13 MFCCs and 13 delta-MFCCs. They achieved 97% accuracy on the NIST-2002 speech corpus for 64 speakers. **[19]** Applied SVM, GMM, and a fusion of SVM with GMM for speaker and language recognition on the NIST 2003 data. They used 38 MFCCs and 36 LPC to describe the speech signal and reported an equal error rate (EER) of 6.46% for SVM, 7.47% for GMM, and 5.55% for SVM/GMM. **[20]** Proposed combining SVM with traditional GMM pattern classification using a 39-dimensional MFCC feature vector extracted from speech. They achieved a 100% identification rate when testing on 64 speakers from the TIMIT database.

While **[21]** constructed a support vector machine kernel using the GMM supervector and conducted experiments on the 2005 NIST speaker recognition corpus. They achieved an EER of 5.68% and a minimum decision cost value (minDCF) of 0.0222 using a 19-dimensional MFCC vector. **[22]** Proposed GMM/SVM-based automatic speaker identification using different acoustic features such as RASTA-MFCC, Gamma tone Frequency Cepstral Coefficients (GFCC), and Mean Hilbert Envelope Coefficients (MHEC) in various noisy conditions. They conducted experiments on the TIMIT phone-labeled database corpus and achieved accuracies of 70.32% for RASTA-MFCC, 68.49% for GFCC, and 73.27% for MHEC under street noise.

[23] Presented an Arabic speaker recognition system for forensic applications using GMM-UBM. They used 39 MFCCs for feature extraction and achieved a recognition rate of approximately 97.8% with an EER of 1.98% using mobile channel recording. **[24]** Explored various deep features for text-dependent speaker verification in both the GMM-UBM and identity vector framework. They evaluated their systems on the RSR2015 database, and the best system achieved an EER of 0.1%.

These related works demonstrate the diverse approaches and techniques employed in Automatic Speaker Recognition, showcasing improvements in recognition rates using different mathematical paradigms and feature extraction methods. Table 1 summarizes the key findings and results of these works in the field of speaker recognition.

The ANFIS has the advantage of good applicability, ability, and performance in system identification, prediction and control. It has been applied in many different systems. Since not many research works have used it in speech recognition. Table 2 presents some of those who do. Thus, it is necessary to carry on the exploration of the use of ANFIS in speech recognition and to more thoroughly research this topic.

Table1. State art of speaker recognition

Authors	Dataset	Features	Classifier	Results (Accuracy %)
[11]	10 Arabic speaker	MFCC	HMM	100%
[13]	Own dataset	five formants and seven entropies	ANN	90,09% Vowel-dependent 82,50% Vowel-independent
[14]	laboratory database 36	EMD	GRNN BPNN	78% 89%
[15]	Own dataset	MFCC Kmeanlbg	ANN	0,015 Error
[16]	TIMIT	39 MFCC	ANN	100%
[17]		16 MFCC	ANN BPNN	92% 70%
[18]	NIST 2002	MFCC CMS RASTA	SVM	97%
[19]	NIST 2003	38 MFCC 36 LPC	SVM GMM SVM+GMM	6,46% EER 7,47% 5,55%
[20]	TIMIT 64 speakers	39 MFCC	SVM+GMM	100
[21]	NIST 2005	19 MFCC	SVM+GMM	5,68% EER
[22]	TIMIT	RASTA MFCC GFCC MHEC	GMM+SVM	70,32% 68,49% 73,27%
[23]	KSU	39 MFCC	GMM- UBM	97,8% EER 1,98
[24]	RSR2015	Deep feautures	GMM –UBM	0,1% EER

The ANFIS (Adaptive Network-based Fuzzy Inference System) demonstrates favorable applicability, capability, and performance in system identification, prediction, and control tasks. Although its application in speech recognition has been relatively limited, there are some notable works that have explored its use in this domain. Table 2 provides an overview of such studies. However, given the scarcity of research utilizing ANFIS in speech recognition, further exploration and in-depth investigation of this topic are necessary to fully understand its potential and benefits in this field.

Table 2. State art of ANFIS in speech

Authors	Features	ANFIS for	Results (Accuracy %)
[25]	wavelet transform	Recognition English language speech signals-isolated words	99%
[26]	MFCC; LPC and the first five formants	Speaker verification	7.14% EER
[27]	LPC	Recognition of discrete words	58%
[28]	MFCC	Speech emotion verification	93% for angry; 89% for sad and 85.3% for happy
[29]	MFCC	Recognize of Malay speech digits	85.24%
[30]	LPC, RC, LPCC, LAR, ARCSIN, LSF	Identification of the speaker, language and the words spoken	83.15%
[31]	MFCC	Phonemes recognition	100%

3. Development of FA-ANFIS Model

This section provides a description of ANFIS (Adaptive Neuro-Fuzzy Inference System) and FA (Firefly Algorithm). Additionally, it explains the process of developing a FA-ANFIS model specifically for the purpose of speaker recognition.

3.1. The Firefly Algorithm

The Firefly Algorithm (FA) is a nature-inspired optimization algorithm that was developed by Xin-She Yang in 2008 **[32]**. It is inspired by the flashing behavior of fireflies and their attraction to each other. The algorithm is particularly useful for solving optimization problems, such as finding the global minimum or maximum of a function.

In nature, fireflies use their bioluminescent light to attract mates or communicate with each other. The Firefly Algorithm simulates this behavior by using the attractiveness of fireflies to guide the search for the optimal solution in a given optimization problem.

The Firefly Algorithm simulates the behavior of fireflies through several key steps. Here's a brief overview of its functioning **[33]**:

1. Initialization: A population of fireflies is randomly generated to represent potential solutions to the optimization problem.
2. Evaluation: Each firefly is evaluated by calculating its fitness value based on the objective function of the problem.
3. Attraction: Fireflies are attracted to each other based on their brightness, which is determined

by their fitness values. Where a higher fitness indicates a brighter firefly. Brighter fireflies are considered more attractive, and fireflies move towards brighter ones. Attractiveness between two fireflies depends on their distance and brightness. Closer and brighter fireflies have stronger attraction, while attraction decreases with increasing distance.

4. Movement: Each firefly adjusts its position by moving towards a more attractive firefly. The movement is influenced by the distance between fireflies and their relative brightness.
5. Intensity: Fireflies adjust their brightness based on their distance from the global best solution found so far. Closer fireflies become brighter, while fireflies farther away become dimmer.
6. Updating: After the movement step, evaluate the fitness of each firefly in the new positions. If a firefly has a better fitness than the firefly it moved toward, the positions are updated.

The Firefly Algorithm aims to find the best solution to the optimization problem by iteratively updating the positions of fireflies based on their attractiveness and movement. The algorithm continues until a stopping criterion is met, such as reaching a maximum number of iterations or achieving a satisfactory solution.

The Firefly Algorithm has been successfully applied to various optimization problems, including function optimization, parameter estimation, feature selection, and clustering. The Firefly Algorithm is widely recognized for its simplicity and straightforward implementation, which has contributed to its popularity in solving optimization problems. However, in order to attain optimal outcomes for specific optimization tasks, it may be necessary to fine-tune its parameters.

The Firefly Algorithm is derived from the natural behavior of fireflies, where they use their self-luminosity to approach each other in the dark. Yang proposed three assumptions to explain the behavior of fireflies. Firstly, all fireflies are considered unisex, meaning they can be attracted to any other firefly regardless of gender. Secondly, the attractiveness of a firefly is determined by its intensity, which is a function of the distance to other fireflies. As the distance increases, the attractiveness decreases. Lastly, the luminosity or luminous intensity of a firefly corresponds to the value of the cost function associated with the problem being solved. Mathematically, the Firefly Algorithm can be described by the following equations [32].

Let's represent the position of a firefly i as $x_i = (x_{i1}, x_{i2}, ..., x_{in})$, where n is the dimensionality of the problem.

- **The light intensity** of a firefly is given by: $I(r) = I_0 \exp(-\gamma.rij)$ (1)

  Where γ is the absorption coefficient and (I0) is the initial value at (r = 0)
- **Attractiveness:** The attractiveness A_{ij} of firefly i towards firefly j can be defined as:

  $$A_{ij} = \beta_0 \exp(-\gamma . r_{ij}^2) \quad (2)$$

  Here, β_0 represents the initial attractiveness (r=0), and r_{ij} represents the Euclidean distance between fireflies i and j and can be defined as:

  $$r_{ij} = \|x_i - x_j\| = \text{sqrt}(\sum_{k=1}^{d} (x_{i,k} - x_{j,k})2) \quad (3)$$

  Where x_{ik} is the k^{th} element of the spatial coordinate x_j of the i^{th} firefly and D denotes the dimensionality of the problem [19].
- **Movement:** The movement of a firefly i towards a more attractive firefly j can be achieved by adjusting its position x_i using the following equation:

  $$x_i(t + 1) = x_i(t) + \beta(x_j(t) - x_i(t)) + \alpha(rand - 0.5) \quad (4)$$

Here, t represents the current iteration, β is the step size, A_{ij} is the attractiveness, $x_j(t)$ and $x_i(t)$ are the positions of fireflies j and i, α is the random parameter and can be constant. "rand" is a random number generator that produces random numbers uniformly distributed in the range [0, 1]. [33]

Algorithm 1. Firefly Algorithm

Initialization of the parameters of FA (Population size, α, β_0, γ and the number of iterations).The Light intensity is defined by the cost function $f(x_i)$ where $x_i(i = 1, \ldots, n)$.

While (iter < Max Generation).
for i = 1:n (all n fireflies)
for j =1:n (all n fireflies)
if ($f(x_i) < f(x_j)$), move firefly i towards j,
end if.
Update attractiveness β with distance r.
Evaluate new solution and update $f(x_i)$ in the same way as (4).
end for j
end for i
rank the solutions and find the best global optimal
end while.
Show the results.

3.2. Adaptive Network-Based Fuzzy Inference System ANFIS

ANFIS proposed by Jang in 1993 multi-layered neural network which connections are not weighted or all weights equal 1**[8]**, ANFIS implement a first order Sugeno style fuzzy system; it applies the rule of TSK Takagi Sugeno and Kang form in its architecture **[34]**. This rule produces crisp outputs directly, as it uses polynomials as rule consequences **[35]**.

Rule: if x is A_1 and y is B_1 then $f = px + qy + r$

Where x and y are the inputs, A1 and B1 are the fuzzy sets, f are the output, p, q and r are the design parameters that determined during the training process.

ANFIS is composed of two parts is the first part is the antecedent and the second part is the conclusion, which are connected to each other by rules in network form. Five layers are used to construct this network. Each layer contains several nodes its structure shows in Figure 1.

Layer 1: executes a fuzzification process which denotes membership functions (MFs) to each input. In this paper, we choose Gaussian functions as membership functions:

$$= U_{Ai} = exp\left[\frac{-(x-c)^2}{2\sigma^2}\right] \qquad (5)$$

Where c and σ are the center and the standard deviation values of input variable x respectively.

Layer 2: executes the fuzzy AND of antecedents part of the fuzzy rules

$$= W_i = \mu_{Ai}(x_1) * \mu_{Bi}(x_2) \qquad i = 1,2,3,4 \qquad (6)$$

Layer 3: normalizes the MFs

$$= \overline{W_i} = \frac{W_i}{\sum_{j=1}^{4} W_j} \qquad j = 1,2,3,4 \qquad (7)$$

Layer 4: executes the conclusion part of fuzzy rules

$$= \overline{W_i} * Y_i = \overline{W_i} * (p_i x_1 + q_i x_2 + r_i) \qquad 1,2,3,4 \qquad (8)$$

Layer 5: computes the output of fuzzy system by summing up the outputs of the fourth layer which is the defuzzification process.

$$= \frac{\sum_{i=1}^{4} W_i Y_i}{\sum_{i=1}^{4} W_i} \qquad 1,2,3,4 \qquad (9)$$

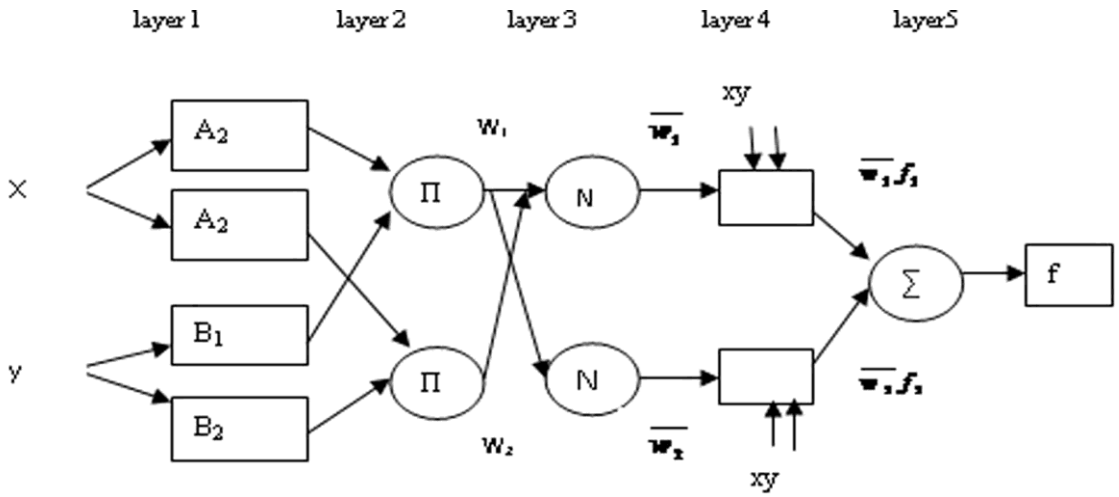

Figure 1. ANFIS architecture

Training Algorithm

The training algorithm you described follows a two-step process: Structure Learning and Parametric Learning, Here's a breakdown of each step:

- Structure Learning: In this step, the algorithm determines the appropriate structure of the network, which involves deciding the partitioning of the input space, i.e., the number of membership functions for each input and the number of rules. When the input dimension is large, the number of rules can grow exponentially, making it crucial to find an appropriate structure. To overcome this problem, clustering techniques are often applied. These techniques help in grouping similar data points together, reducing the complexity of the network structure **[36]**;
- Parametric Learning: Once the structure is determined, the algorithm moves on to

adjust the antecedent and consequent parameters of the fuzzy inference system. The objective of this step is to minimize a specified objective function. **[37]** proposed four methods for updating these parameters, but the most common approach is hybrid learning, which combines Gradient Descent (GD) and Least Squares Estimation (LSE) techniques.

This algorithm is carried out in two steps:

1) **Forward Pass:** During the forward pass, the input patterns are propagated through the network to generate output values. In this step, the optimal consequent parameters are estimated using the LSE method, while the antecedent parameters are assumed to be fixed in the current training cycle. The LSE method involves finding the parameters that minimize the squared error between the predicted and target outputs.

2) **Backward Pass:** In the backward pass, the patterns are propagated through the network again. This time, the antecedent parameters are updated using the GD method. Gradient Descent is an optimization algorithm that iteratively adjusts the parameters in the direction of steepest descent of the objective function. During this pass, the consequent parameters remain fixed since they were already updated in the forward pass.

By iteratively performing the forward and backward passes, the algorithm continues to refine the antecedent and consequent parameters until convergence is reached, or a stopping criterion is met. This iterative training process helps the network improve its ability to make accurate predictions or decisions based on the given training data.

The hybrid procedure described above is repeated until the output error reaches a desired goal or a maximum number of training cycles is reached. However, it is important to note that this algorithm may encounter the issue of getting trapped at local optima. To address this problem, evolutionary algorithms can be an effective solution. These algorithms are capable of exploring a wider search space but can be computationally expensive, especially when there are numerous parameters to optimize. As a result, a combination of techniques is employed in this approach. Specifically, the Least Squares Estimation (LSE) method is utilized to optimize a subset of the parameters, namely the consequent parameters, while an evolutionary technique, such as Firefly Algorithm (FA), is employed to optimize the antecedent parameters cij and σij. This combination aims to strike a balance between efficiency and effectiveness in parameter optimization.

4. Experimental and Discussions

4.1. Database

The CHAINS corpus, introduced by Cummins, Leonard, Leonardo, and Simko in 2006 **[37]**, serves as the database for this study. It consists of recordings from 36 speakers, including 28 individuals from the Eastern region of Ireland who speak Eastern Hiberno-English. The remaining speakers originate from the United Kingdom (UK) and the United States of America (USA). The speakers were recorded in various scenarios, such as reading texts individually, in synchronization with a dialect-matched co-speaker, imitating a dialect-matched co-speaker, whispering, and speaking rapidly under different conditions. Four speaker sets are available in this corpus: 8-speaker, 16-speaker, 24-speaker, and 36-speaker

sets. In this study, the 8-speaker and 16-speaker sets were utilized. The datasets are formatted in WEKA arff format and include speech samples encoded using 25 Mel Frequency Cepstral Coefficients (MFCC).

4.2. Initial ANFIS Structure

The initial step in the process was to normalize all datasets, ensuring they were scaled within the range of (0,1). This normalization procedure adjusted the values of the datasets to a standardized scale (using equation 10), facilitating consistent comparisons and analysis:

$$Xn = \frac{X - Xmin}{Xmax - Xmin} \qquad (10)$$

Where X_n and X are the normalized and measured values, respectively. Also, the minimum and maximum values of the measured dataset are X_{min} and X_{max} respectively.

Once the initial ANFIS structure is determined, the fuzzy clustering algorithm (FCM) is employed to cluster the data space and determine the optimal number of rules and membership functions for both the antecedents and consequents.

In this application, the initial ANFIS structure is generated with 10 rules. Subsequently, 10 Gaussian-type membership functions are generated for each input. The center of each membership function is initialized based on the corresponding fuzzy cluster obtained from the clustering process. The clustering is performed with a radius of 0.5, which helps define the boundaries and distribution of the membership functions within the input space.

4.3. Training Methodology

Certainly, here is a more detailed explanation of the training conditions for the two methodologies used in the speaker recognition process:

4.3.1. ANFIS Model

a. **Training Methodology:** The ANFIS model is trained using a hybrid learning approach that combines Gradient Descent (GD) and Least Squares Estimation (LSE).

b. Training Options:

- Training Epoch Number: The ANFIS model undergoes 10 training epochs, where each epoch represents a complete pass through the training data. This iterative process allows the model to gradually learn and improve its performance.
- Training Error Goal: The training error goal is set to 0, indicating that the objective is to minimize the difference between the model's predicted outputs and the actual outputs of the training data. Achieving a training error of 0 signifies optimal accuracy.
- Initial Step Size: The initial step size is set to 0.01, which determines the magnitude of parameter updates during the GD and LSE optimization process. This step size helps control the rate at which the model parameters are adjusted.

- Step Size Decrease Rate: The step size decrease rate is set to 0.9, meaning that the step size is reduced by 10% after each training epoch. Gradually decreasing the step size allows for finer adjustments and improves the convergence of the optimization process.
- Step Size Increase Rate: The step size increase rate is set to 1.1. If the improvement in the training error rate is not satisfactory, the step size is increased by 10%. This allows for faster progress towards the optimal solution if the model's performance is not improving significantly.

4.3.2. Hybrid FA-ANFIS:

a. **Training Methodology:** The ANFIS model's antecedent parameters are tuned using a combination of Firefly Algorithm (FA) and ANFIS.

b. Training Process:

ANFIS Antecedent Parameters: The focus of the training is on optimizing the antecedent parameters of the ANFIS model, which significantly influence the model's ability to capture speaker-specific characteristics.

FA Optimization: The FA algorithm is employed to iteratively search for the optimal values of the ANFIS antecedent parameters. Inspired by the behavior of firefly, FA uses a population of fireflies to explore the search space.

c. **Particle Positions:** Each firefly in the FA algorithm represents a potential solution in the search space. The positions of the fireflies are updated based on their individual best positions (the best solution found by each firefly) depending on their brightness and the global best position (the best solution found by any firefly in the population).

d. **Tuning Strategy:** The objective function used in FA is based on the performance of the ANFIS model for speaker recognition. By iteratively updating the firefly positions, the algorithm aims to find the optimal combination of antecedent parameter values that yield the highest accuracy in speaker recognition.

To determine the optimal parameters for Firefly Algorithm (FA) algorithm, various parametric studies were conducted. In this study, a trial-and-error approach was employed to find the most suitable FA parameters. The model is illustrated in Figure 2.

The objective of the optimization process was to minimize the function for Recognition Rate. This function quantifies the accuracy of the speaker recognition system and serves as the criterion for evaluating the performance of the FA algorithm. The aim was to find the parameter values that would yield the highest recognition rate, indicating the most effective configuration for the FA algorithm.

$$Acuuracy = (\frac{1}{N} SUM(Di = Oi)) * 100 \quad (11)$$

Where d_i is the desired output, O_i is the ANFIS output for the ith sample from the training data, and N is the number of training samples.

4.4. Results and Discussions

The results obtained from the experiments and their discussions are as follows:

Figure 3 presents the results obtained from applying the ANFIS model to the 8 speakers in the CHAINS database. The accuracy achieved was 87.99%. This means that the ANFIS model successfully recognized the speakers with a high level of accuracy.

Similarly, Figure 4 showcases the results of the ANFIS model applied to the 16 speakers in the database. In this case, the accuracy achieved was 93.80%. The ANFIS model performed well in recognizing the speakers, achieving a high accuracy rate. Moving on to Table 3, it displays the results obtained from the FA-ANFIS model when varying the size of the population (NPop) for the 8 speakers. The experiment explored different population sizes ranging from 20 to 300 fireflies, with 1000 iterations. The findings revealed that increasing the population size generally improved the accuracy of the model. The highest accuracy of 90.25% was achieved when using 200 fireflies. However, when the population size was further increased to 300 fireflies, a slight decrease in accuracy to 89.95% was observed.

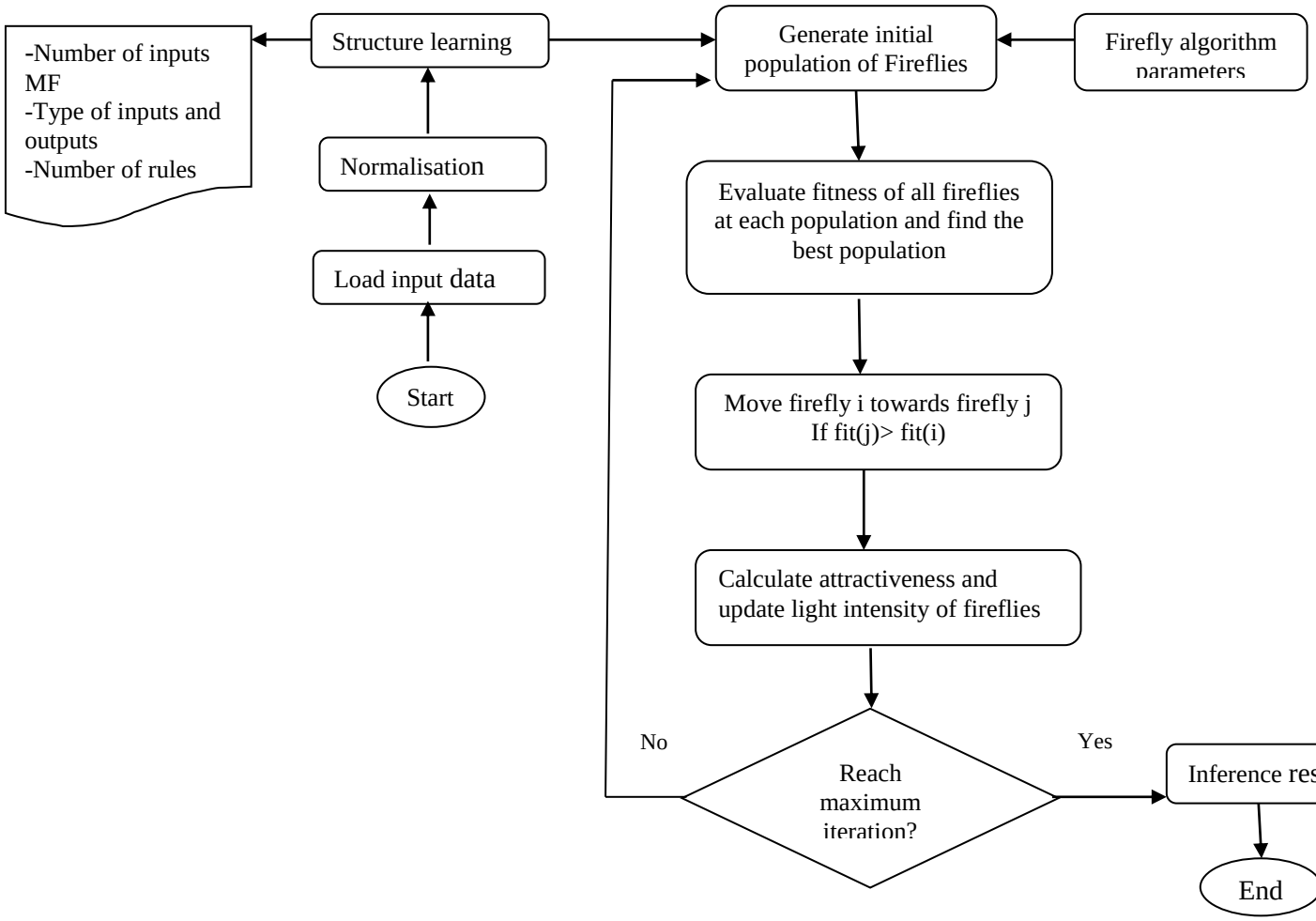

Figure 2. Flowchart of hybrid FA-ANFIS

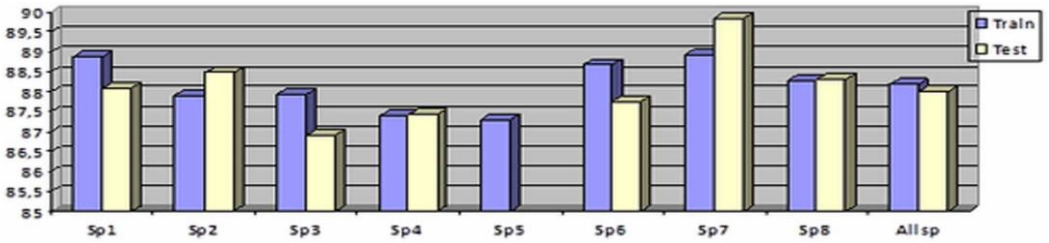

Figure 3. Accuracy on 8 speakers using ANFIS

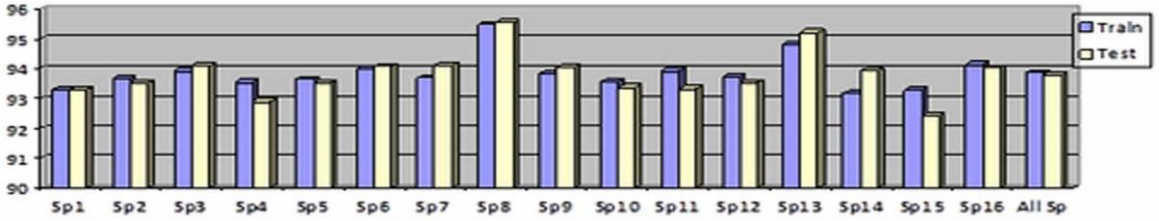

Figure 4. Accuracy on16 speakers using ANFIS

In Table 4, the focus shifts to the application of the FA-ANFIS model on the 16 speakers, considering variations in both the population size and the number of iterations. The best accuracy obtained for the 16 speakers was 95.20%, achieved with 1000 iterations and a population size of 100.

Table 3. Accuracy on 8 speakers using FA-ANFIS (1000 iterations)

NPop	20	50	100	150	200	300
Train	89.53	90.37	90.43	90.58	90.95	90.58
Test	89.19	89.87	89.89	89.97	90.25	89.95

Table 4. Accuracy on 16 speakers using FA-ANFIS

Iteration	NPop	500 100	500 200	1000 100	1000 200
Train		95.20	95.26	95.34	95.36
Test		95.97	95.14	95.20	95.18

To compare the two training methods, Table 5 and Table 6 present a side-by-side analysis for the 8 speakers and 16 speakers, respectively. The results clearly demonstrate the improvement in accuracy achieved by the FA-ANFIS model compared to the ANFIS model. For the 8 speakers, the accuracy increased from 89.99% (ANFIS) to 90.29% (FA-ANFIS). Similarly, for the 16 speakers, the accuracy improved from 94.80% (ANFIS) to 95.19% (FA-

ANFIS).

Table 5. Comparison of ANFIS and FA-ANFIS on 8 speakers

Rate	ANFIS	FA-ANFIS
Train	88.28	89.87
Test	89.09	90.29

Table 6. Comparison of ANFIS and FA-ANFIS on 16 speakers

Rate	ANFIS	FA-ANFIS
Train	93.97	94.21
Test	94.91	95.19

These results indicate that the FA-ANFIS model, with its optimization capabilities and parameter tuning through FA, enhances the accuracy of the speaker recognition system compared to the ANFIS model alone. The experiments conducted highlight the effectiveness of the hybrid FA-ANFIS approach in achieving higher accuracy rates for speaker recognition tasks. Our research demonstrates a notable advancement compared to previous studies in the field, as evidenced by the results presented in Table 7.

Table 7. Comparison of proposed method and existing model

Dataset version	Proposed	38]		9]		0]		[41]
8 Speakers	90.29	MLP	71.77	Self SSL 81.45		SVM	87.44	Entropy 83.94
16 Speakers	95.19	MLP	58.08	Self SSL 78.19		SVM	83.70	SMargin 83.10

5. Conclusion

In conclusion, this study introduced a novel approach for speaker recognition by developing an Adaptive Network based Fuzzy Inference System (ANFIS) tuned using Firefly Algorithm (FA). Unlike conventional learning algorithms that depend on Gradient Descent (GD), the use of FA provides advantages by avoiding local optima during the training phase. FA was specifically applied to optimize the antecedent part of the ANFIS model, while the consequent parameters were optimized using Least Squares Estimation (LSE), which yielded superior results compared to the GD method.

As for future perspectives, the research will expand to explore the application of FA optimization techniques to temporal and recurrent neural models, which can further improve the accuracy and performance of speaker recognition systems. Additionally, the focus will be on processing larger benchmark datasets that contain more than the current 36 speakers. By scaling up the experiments, a more comprehensive evaluation of the FA-ANFIS approach can be conducted, providing valuable insights and potential improvements for real-world applications of speaker recognition.

References

[1] Huang, X., Acero, A., Hon, H., & Reddy, R. (2001). Spoken Language Processing: A guide to Theory, Algorithm, And System Development. Prentice-Hall.

[2] He, X., & Deng, L. (2008). Discriminative Learning for Speech Recognition: Theory and practice. Morgan & Claypool.

[3] Snyder, D., Garcia-Romero, D., Sell, G., Povey, D., & Khudanpur, S. (2018). X-vectors: Robust DNN embeddings for speaker recognition. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 5329-5333.

[4] Snyder, D., Chen, G., Povey, D., & Khudanpur, S. (2020). Spoken language processing: A guide to theory, algorithm, and system development. Morgan & Claypool Publishers.

[5] Bojadziev, G., & Bojadziev, M. (1995). Fuzzy Sets, Fuzzy Logic, Applications, Advances in fuzzy systems applications and theory. World Scientific. doi:10.1142/2867

[6] Abraham, A. (2001). Neuro Fuzzy Systms: State-of-the-Art Modeling Techniques. In Artificial and Natural Neural Networks IWANN, 6th International Work-Conference on, June, Vol. 2084, pp. 269-276. Academic Press.

[7] Kosko, B. (1991). Neural Networks and Fuzzy Systems A Dynamic Systems Approach. Prentice-Hall.

[8] Jang, J. S. R. (1993). ANFIS: Adaptive Network Based Fuzzy Inference Systems. IEEE Transactions on Systems, Man, and Cybernetics, 23(3), 665–685. doi:10.1109/21.256541

[9] Fatemeh, Z., & Zahra, Z. (2018). A review of neuro-fuzzy systems based on intelligent control. Journal of Electrical and Electronic Engineering, 3(2-1), 58-61.

[10] Gunasekaran, M., Varatharajan, R., & Priyan, M. K. (2018). Hybrid Recommendation System for Heart Disease Diagnosis based on Multiple Kernel Learning with Adaptive Neuro-Fuzzy Inference System. An International Journal Multimedia Tools and Applications, 77(4), 4379–4399. doi:10.1007/s11042-017-5515-y

[11] Tolba, H. (2011). A high-performance text-independent speaker identification of Arabic speakers using a CHMM- based approach. Alexandria Engineering Journal, 50(1), 43–47. doi:10.1016/j.aej.2011.01.007

[12] Logan, B. (2000). Mel frequency cepstral coefficients for music modeling. In ISMIR (pp. 1-5).

[13] Daqrouqa, K., & Tutunjib, T. A. (2015). Speaker identification using vowels features through a combined method of formants, wavelets, and neural network classifiers. Applied Soft Computing, 27, 231–239. doi:10.1016/j. asoc.2014.11.016

[14] Wu, J. D., & Tsai, Y. J. (2011). Speaker identification system using empirical mode decomposition and an artificial neural network. Expert Systems with Applications, 38(5), 6112–6117. doi:10.1016/j.eswa.2010.11.013

[15] Chen, Y., Wang, L., Lin, H., & Li, J. (2012, October). Design of speaker recognition system based on artificial neural network. In Advanced Optical Manufacturing and Testing Technologies: Optical System Technologies for Manufacturing and Testing. Proceedings of the SPIE, 2012. AOMATT 6th International Symposium on (Vol. 8420, pp. 1-7). doi:10.1117/12.970642

[16] Ge, Z., Iyer, A. N., Cheluvaraja, S., Sundaram, R., & Ganapathiraju, A. (2017, September).

Neural Network Based Speaker Classification and Verification Systems with Enhanced Features. In Intelligent Systems Conference IntelliSys (pp. 1–6). IEEE. doi:10.1109/IntelliSys.2017.8324265

[17] Wali, S. S., & Hatture, S. M. (2015). MFCC Based Text-Dependent Speaker Identification Using BPNN. International Journal of Signal Processing Systems, 3(1), 30–34.

[18] Apsingekar, V. R., & De Leon, P. L. (2009). Support Vector Machine Based Speaker Identification Systems Using GMM Parameters. In Signals, Systems and Computers, 2009 Conference Record of the Forty- Third Asilomar Conference on, November, pp. 1766-1769, IEEE. doi:10.1109/ACSSC.2009.5470201

[19] Campbell, W. M., Campbell, J. P., Reynolds, D. A., Singer, E., & Torres-Carrasquillo, P. A. (2006). Support vector machines for speaker and language recognition. Computer Speech & Language, 20(2-3), 210–229. doi:10.1016/j.csl.2005.06.003

[20] Chakroun, R., Zouari, L. B., Frikha, M., & Hamida, A. B. (2015, December). A hybrid system based on GMM- SVM for Speaker Identification. In Intelligent Systems DeSign and Applications(ISDA), 2015 15th International Conference on, (pp. 645-658). IEEE. doi:10.1109/ISDA.2015.7489195

[21] Campbell, W. M., Sturim, D. E., & Reynolds, D. A. (2006). Support Vector Machines Using GMM Supervectors for Speaker Verification. IEEE Signal Processing Letters, 13(5), 308–311. doi:10.1109/LSP.2006.870086

[22] Dhineshkumar, R., Ganesh, A. B., & Sasikala, S. (2016). Speaker Identification System using Gaussian Mixture Model and Support Vector Machines (GMM-SVM) under Noisy Conditions. Indian Journal of Science and Technology, 9(19), 1–6.

[23] Algabri, M., Mathkour, H., Bencherif, M. A., Alsulaiman, M., & Mekhtiche, M. A. (2017). Automatic Speaker Recognition for Mobile Forensic Applications. Hindawi Mobile Information Systems, 1–6. doi:10.1155/2017/6986391

[24] Liu, Y., Qian, Y., Chen, N., Fu, T., Zhang, Y., & Yu, K. (2015). Deep feature for text-dependent speaker verification. Speech Communication, 73, 1–13. doi:10.1016/j.specom.2015.07.003

[25] Elwakdy, A. M., Elsehely, B. E., Eltokhy, C. M., & Elhennawy, D. A. (2008, July). Speech recognition using a wavelet transform to establish fuzzy inference system through subtractive clustering and neural network (ANFIS). In ICS'08 Proceedings. the 12th WSEAS international conference on Systems (pp. 381-386). Academic Press.

[26] Srihari, V., Karthik, R., Anitha, R., & Suganthi, S. D. (2010, December). Speaker verification using combinational features and adaptive neuro-fuzzy inference systems. In Intelligent Interactive Technologies and Multimedia. IIMT'10 the First International Conference on (pp. 98-103). doi:10.1145/1963564.1963580

[27] Helmi, N., & Helmi, B. H. (2008, October). Speech recognition with fuzzy neural network for discrete words. In Natural Computation 2008, Proceedings ICNC Fourth International Conference on (Vol. 7, pp. 265–269). IEEE. doi:10.1109/ICNC.2008.666

[28] Kamaruddin, N., & Wahab, A. (2008, July). Speech Emotion Verification System (SEVS) based on MFCC for real time application. In Intelligent Environments, IET 4th International Conference on (pp. 1-7). IEEE.

[29] Sabah, R., & Ainon, R. N. (2009, May). Isolated Digit Speech Recognition in Malay Language using Neuro- Fuzzy Approach. In Modelling & Simulation, 2009. AMS '09. Third

Asia International Conference on, (pp. 336– 340). IEEE.

[30] Pandey, B., Ranjan, A., Kumar, R., & Shukla, A. (2010, July). Multilingual Speaker Recognition Using ANFIS. Signal Processing Systems (ICSPS), 2010 2nd International Conference on, 3, 714-718.

[31] Anonymous. (2014). Adaptive Network Based Fuzzy Inference System For Speech Recognition Through Subtractive Clustering. International Journal of Artificial Intelligence & Applications, 5(6), 43–52. doi:10.5121/ijaia.2014.5604

[32] Yang, X. S. (2008). Firefly algorithms for multimodal optimization. In International symposium on stochastic algorithms (pp. 169-178). Springer.

[33] Yang, X. S. (2012). Nature-inspired optimization algorithms. Elsevier.

[34] Takagi, T., & Sugeno, M. (1985). Fuzzy identification of systems and its applications to modeling and control. *Syst Man and Cybern IEEETrans SMC*, *15*(1), 116–132. doi:10.1109/TSMC.1985.6313399

[35] Li, J., Yang, L., Qu, Y., & Sexton, G. (2018). An extended Takagi–Sugeno–Kang inference system (TSK+) with fuzzy interpolation and its rule base generation. Soft Computing, 22(10), 3155–3170. doi:10.1007/s00500- 017-2925-8

[36] Priyono, A., Ridwan, M., Alias, A. J., Rahmat, R. A. O. K., Hassan, A., & Ali, M. A. M. (2005). Generation of fuzzy rules with subtractive clustering. Journal Teknologi, 43(1), 143–153. doi:10.11113/jt.v43.782

[37] Jang, J. S. R., Sun, C. T., & Mizutani, E. (1997). Neuro-Fuuzy and Soft Computing: a Computational Approach to Learning and Machine Intelligence. Prentice-Hall.

[38] Cummins, F., Leonard, M., Leonardo, T., & Simko, J. (2006, June). The CHAINS corpus CHAracterizing INdividual Speakers. In Speech and Computer SPECOM, The International Conference on (pp. 431-435). Academic Press.

[39] Fazakis, N., Karlos, S., Kotsiantis, S., & Sgarbas, K. (2015). Speaker Identification Using Semi-supervised Learning. Lecture Notes in Computer Science, 9319, 389–396. doi:10.1007/978-3-319-23132-7_48

[40] Karlos, S., Fazakis, N., Karanikola, K., Kotsiantis, S., & Sgarbas, K. (2016). Speech Recognition CombiningMFCCs and Image Features. Lecture Notes in Computer Science, 9811, 651–658. doi:10.1007/978-3-319-43958-7_79

[41] Karlos, S., Kaleris, K., Fazakis, N., Kanas, V. G., & Kotsiantis, S. (2018). Optimized Active Learning Strategy for Audiovisual Speaker Recognition. Lecture Notes in Computer Science, 11096, 281–290. doi:10.1007/978- 3-319-99579-3_30