EURASIAN JOURNAL OF MATHEMATICAL AND COMPUTER APPLICATIONS
ISSN 2306–6172
Volume 14, Issue 2 (2026) 4 – 34

AUTOMATING PUBLIC SPEAKING ANXIETY TREATMENT USING LSTM RECURRENT NEURAL NETWORK AND AUGMENTED REALITY

Abdollahzadeh F. , Kamel Tabbakh Farizani S. [1] , Forghani Y. Niazi Torshiz M. , Abdollahzadeh H.

Abstract Public Speaking Anxiety (PSA) is recognized as one of the most prevalent fears, significantly impacting individuals in both professional and daily contexts. While traditional psychological treatments have been enhanced by technologies like Virtual Reality (VR) and Augmented Reality (AR), less research has focused on automating the treatment model to reduce therapy time. Therefore, the aim of this research is to propose a novel model to automate the treatment process for PSA using an adaptive LSTM recurrent neural network. To train the proposed model, a dataset consisting of physiological information from patients treated by a psychologist was used. Considering the assignment of different weights to the training data by the psychologist based on the patient's improvement level, a weighted loss function (Weighted MSE) is proposed for training the suggested LSTM neural network model. The comparative results between the weighted MSE trained by LSTM and the unweighted MSE demonstrated that the LSTM algorithm with the weighted MSE improves diagnostic performance. This means that the proposed model can facilitate and accelerate patient treatment in accordance with the psychologist's opinion, which justifies and even recommends its use.

Keywords: Information Systems, Public Speaking Anxiety, Recurrent Neural Network, Modeling and Simulation, Data Science, Virtual Reality, Applied Mathematics, Machine Learning, Social Anxiety, Adaptive Normalization, Augmented Reality.

AMS Mathematics Subject Classification: 91C99, 68T01.

DOI: 10.32523/2306-6172-2026-14-2-4–34

1 Introduction

One of the most common and relevant social phobias in relation to all ages and demographics is Public Speaking Anxiety (PSA), which can be defined as anxiety when facing a speech in front of one or more live audiences [1]. In today's society, most people turn to classical methods to treat PSA, including Cognitive Behavioral Therapy (CBT) with exposure therapy and regular desensitization [2]. Despite their advantages, the use of these methods also presents several challenges. One major issue is that providing controlled and manageable environments for therapy can be complex and costly [3]. Another challenging issue in using traditional methods is the lengthy duration of

[1]Corresponding Author.

the treatment process. Technological advancements in treatment, such as the use of Virtual Reality (VR) and Augmented Reality (AR), offer potential solutions to address these challenges [4, 5, 6]. Although VR reduces the limitations of the exposure therapy method, its use comes with challenges such as the virtual and unrealistic nature of the entire exposure environment, as well as the high cost and discomfort associated with using head-mounted equipment [7, 8, 9, 10]. To address the challenges of using VR in therapeutic contexts, AR integrates virtual and synthetic elements into the fully real and dynamic environment surrounding the individual, rather than immersing the person in a completely virtual setting. These two technologies can provide the therapist with the desired conditions in the clinical environment, which reduces the cost, time of treatment, as well as the feasibility of treatment [11]. One of the advantages of AR is the ability of this technology to combine with other technologies such as deep learning and semantic web technologies, which can improve the performance of AR. Deep learning can instill intelligence into AR systems and can be used as a means of improving computer vision [12]. Deep learning techniques can be used to detect fear and anxiety [13, 14, 15]. But these algorithms alone are not suitable for predicting the patient's subsequent reactions. Therefore, the integration of Artificial Intelligence (AI) and AR is considered a novel approach in the treatment of anxiety disorders, emphasizing the importance of adapting therapy based on patients' psychophysiological responses (heart rate, Galvanic Skin Response (GSR), and respiration) [16]. In the treatment of PSA by psychologists, there is a delay in analyzing physiological feedback such as heart rate, GSR, and body temperature for making timely and accurate decisions [11]. Consequently, automating the treatment model can significantly reduce the overall duration. In this study, based on the aforementioned reasons and with the aim of achieving optimal results with minimal error, a proposed model based on the Long Short-Term Memory (LSTM) recurrent neural network and adaptive normalization has been utilized. In the proposed model, the following hypotheses have been considered, and their validity has been examined and confirmed. - The proposed model based on LSTM recurrent neural network network is capable of achieving performance comparable to that of a psychologist, with a very low error rate.
- Incorporating a weighted loss function, as opposed to a standard unweighted loss function, results in improved performance of the recurrent neural network.
- Adaptive normalization can enhance the accuracy of the results. Therefore, the aim of this study is to design a decision support system for psychologists to help them treat patients in an augmented reality environment. For this purpose, an adaptive LSTM recurrent neural network with a new loss function is proposed to receive physiological feedback from various sensors attached to the patient's body and, based on that, issue treatment suggestions like a psychologist.
The remainder of this paper is organized as follows: Section 2 presents the Related Work. Section 3 provides an overview of concepts related to machine learning. In Section 4, the therapy model derived from the psychologist's behavioral patterns and the proposed model based on the recurrent neural network algorithm are presented. Section 5 is Analysis and Evaluation. Finally, the Conclusion and Future Work are presented in Section 6.

2 Related Work

In the field of diagnosis and treatment of anxiety and fear, a lot of research work has been presented, especially in recent years. In study [17], researchers looked at the effectiveness of an AR-based application in reducing fear of spiders. The application, which ran on smartphones, allowed for gradual virtual exposure to spiders in the real world. In study [18], researchers conducted a randomized clinical trial to evaluate the effectiveness of AR-based exposure therapy in treating spider phobia. Participants in this study were gradually exposed to fear-inducing stimuli using AR technology.
Palau-Batet et al. [19], in a single-line multilinear study with feasibility design, examined the effectiveness of AR in treating cockroach phobias. In this study, AR technology was used to display live images of cockroaches in real environments, and participants were gradually exposed to these images.
Juan et al. [20], designed an AR system of the optical see-through type that allowed users to view virtual images of small animals (such as insects and rodents) in real environments.
Fatharany et al. [21], designed an application that is used to treat cockroach phobias and uses everyday objects as a marker substitute.
In study [22], a body sensor-based system was proposed for behavior recognition using an RNN. In this study, data integration from multiple body sensors such as electrocardiography (ECG), accelerometers, magnetometers, and others was performed. The extracted features were enhanced using Kernel Principal Component Analysis (PCA). These optimized features were then used to train the RNN, which was applied for behavior recognition. The system was evaluated on three standard datasets, and the results indicate that the proposed method can be effectively applied to larger and more complex datasets with intricate activities, enabling the development of a real-time human behavior monitoring system.
In study [23], the researchers investigated machine learning and deep learning methods for fear level detection and the treatment of acrophobia. They concluded that by leveraging machine learning algorithms such as Random Forest (RF) and K-Nearest Neighbors (KNN), along with deep neural networks, the therapeutic system can automatically adjust exposure scenarios in a VR environment based on the patient's emotional state. This approach enables personalized therapy without the need for continuous human intervention.
In a study [24], fear classification based on physiological features was investigated. The classification algorithms used included Decision Trees, Support Vector Machines (SVM), KNN, and Neural Networks. Feature extraction methods such as PCA and XGBoost were also employed. The dataset used was derived from the DEAP (Dataset for Emotion Analysis using Physiological Signals) database. The researchers' findings indicated that fear can be predicted by extracting the most relevant features from electrodermal activity and heart rate data, and by optimizing algorithm parameters to maximize performance. Additionally, they demonstrated that the SVM algorithm combined with PCA achieved an accuracy of 93.5%, outperforming the Gradient Boosting method by approximately 108%.
In another study [25], a comparison was made between machine learning techniques and deep learning approaches, involving four deep neural network models, a randomly

configured network, SVM, RF and KNN. To rank fear levels, the researchers used two methods: the first was a binary scale where 0 indicated relaxation and 1 indicated fear; the second was a four-level scale: 0 for relaxation, 1 for low fear, 2 for moderate fear, and 3 for high fear. The dataset used was sourced from the DEAP database. The results showed that the Random Forest algorithm combined with PCA-based feature extraction achieved higher accuracy compared to the other algorithms.
In study [26], researchers proposed emotion recognition models based on facial expressions and features. The techniques used in this research include SVM, RF, and RNN-LSTM. The results demonstrated that the combination of LSTM or RNN networks provided higher accuracy and precision compared to the other classifiers.
In another study [27], a multimodal assessment system was proposed to evaluate stress levels using non-invasive EEG and ECG signals. The study involved 24 participants aged between 18 and 23. The acquired signals were processed using semi-supervised machine learning techniques. Model validation was performed using five supervised machine learning algorithms: KNN, SVM, Decision Tree (DT), RF, and Artificial Neural Network (ANN). The results of the study demonstrated the superiority of the SVM algorithm.
In study [28], researchers proposed a detection model based on EEG signals for the treatment of social anxiety disorder. The EEG data were classified using Convolutional Neural Networks (CNN), LSTM networks, and a hybrid model combining both (CNN + LSTM). The results demonstrated that the proposed CNN + LSTM model outperformed the individual models in the detection and treatment of social anxiety disorder.
As reviewed in the related studies, there has been a lot of interesting work in the use of traditional methods and new technologies in the treatment of anxiety, especially PSA, and researchers are trying to provide new treatments using new technologies such as VR and developed reality. Therefore, the integration of powerful artificial intelligence technologies into technology-based treatment models will undoubtedly enrich this research field.

3 Review of Concepts

This section provides a review of the key concepts used in the proposed model.

3.1 LSTM Network

Since this study focuses on body sensor data for anxiety treatment, a machine learning model capable of modeling sequence-based information should be adopted. Among the machine learning models used by researchers in various fields, RNN has become very popular due to the robustness in modeling events that lie in consecutive time data. RNNs have recurrent connections within their hidden units, which essentially maintain the relationship between historical and current states. As a result, an RNN utilizes the memory within the system to model a sequence. However, a conventional RNN suffers from the vanishing gradient problem, which occurs when modeling long sequences. In this study, physiological signals are recorded over long time periods, and for treatment-related decision-making, the expert or psychologist relies on previously

observed temporal intervals. Therefore, the model is also constructed based on this temporal dependency. Consequently, the data exhibit long-term dependencies, making it suitable to use a specific type of recurrent neural network known as LSTM, which overcomes the vanishing gradient problem observed in traditional RNNs. Moreover, LSTM is a memory-based recurrent neural network that performs better than standard memory-less RNNs when dealing with longer sequences. In our proposed approach, the inputs to the LSTM algorithm consist of the number of audience members and data collected from body sensors. These physiological signals are fed into the LSTM network, and the output of the proposed algorithm determines the next action in the treatment process, whether to increase, decrease, or maintain the current number of audience members. An LSTM network comprises an input gate i, forget gate ft , output gate o, and a cell activation vector c. Each gate consists of a sigmoid neural network layer and an element-wise multiplication operation. The sigmoid layers output values in the range [0,1], representing the fraction of the input information allowed to pass through. Equations (1) to (6) define the computations of this model [29].

$$f_t = \sigma(W_{fh}h_{t-1} + W_{fx}x_t + b_f) \tag{1}$$

$$i_t = \sigma(W_{ih}h_{t-1} + W_{ix}x_t + b_i) \tag{2}$$

$$\tilde{c}_t = \tanh(W_{c\sim}h_{t-1} + W_{c\sim x}x_t + b_{c\sim}) \tag{3}$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tilde{c}_t \tag{4}$$

$$o_t = \sigma(W_{oh}h_{t-1} + W_{ox}x_t + b_o) \tag{5}$$

$$h_t = o_t \cdot \tanh(c_t) \tag{6}$$

In Equations (1 to 6), C_t represents the long-term memory of the network at time t, and W_c, W_i, and W_o denote the weights, or more specifically, the kernels associated with h_{t-1}. The symbols W_f and b_f correspond to the weight matrix and bias of the forget gate, respectively. (W_i, b_i) and (W_c, b_c) represent the weight matrices and biases of the input gate and the memory cell state, respectively. W_o and b_o denote the weight matrix and bias of the output gate, which determine the portion of the cell state that is output at each time step. h_{t-1} represents the output of the previous cell, and i_t is the first input gate path value after applying the sigmoid function to the inputs X_t and h_{t-1}. o_t denotes the output gate value, and h_t represents the final output of the cell at time t. When updating the cell state, the input gate decides which new information can be stored in the cell state, while the output gate determines which information can be produced based on the current cell state. Figure (1) illustrates the structure of the LSTM cell [29].

When training the network, the i-th training sample represented as a sequence of input values denoted by X_i, is fed into the LSTM network equipped with adaptive normalization. The LSTM processes these signals and, at each time step, generates a time-series output in the form of audience variation actions (increase, decrease, or no change). The goal of the model is for the LSTM output to closely match the psychologist's judgment, represented by y_i, or to minimize the discrepancy between

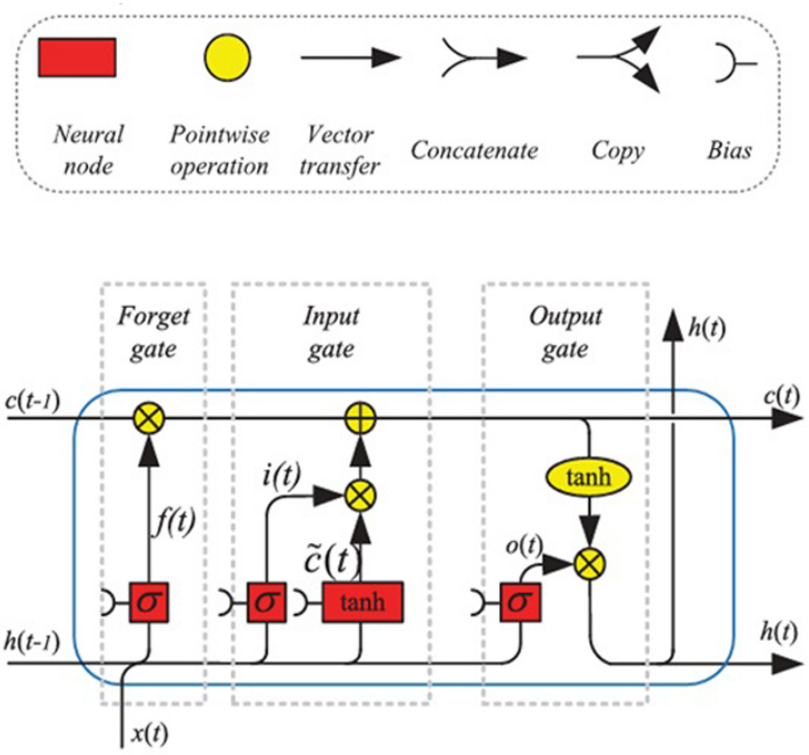


Figure 1: The Structure of an LSTM Cell [29]

them. Based on this objective, the loss function is derived. This loss is computed over all training samples and defined formally by Equation (7).

$$\text{Min} \sum_{i=1}^{n} \left(LSTM(x_i) - y_i\right)^2 \tag{7}$$

In fact, the goal is to optimize the weights of the LSTM network and the adaptive normalization parameters in such a way that the objective function is minimized. This strategy assumes that all individuals have been successfully treated; that is, the training data provided by the psychologist corresponds to cases where all subjects have achieved complete recovery. However, if there are individuals who have not been fully treated through the combined approach of psychologist + sensor + AR, then our proposed innovative solution involves assigning different weights to the training data, denoted by w_i. These weights are essentially determined based on the psychologist's assessment and reflect the importance of the network's error for the i-th data point. In other words, the psychologist's evaluation of each individual's recovery level or percentage of improvement at the end of each session is incorporated through these weights. The corresponding objective function is defined as follows:

$$\min \sum_{i=1}^{n} w_i \left(LSTM(x_i) - y_i\right)^2 \tag{8}$$

3.2 Adaptive Normalization

In order to analyze signals in the Train and Test stages by the algorithm, normalization is required. All signals within the LSTM network will pass through the adaptive normalization component, which is defined as shown in Figure (2).

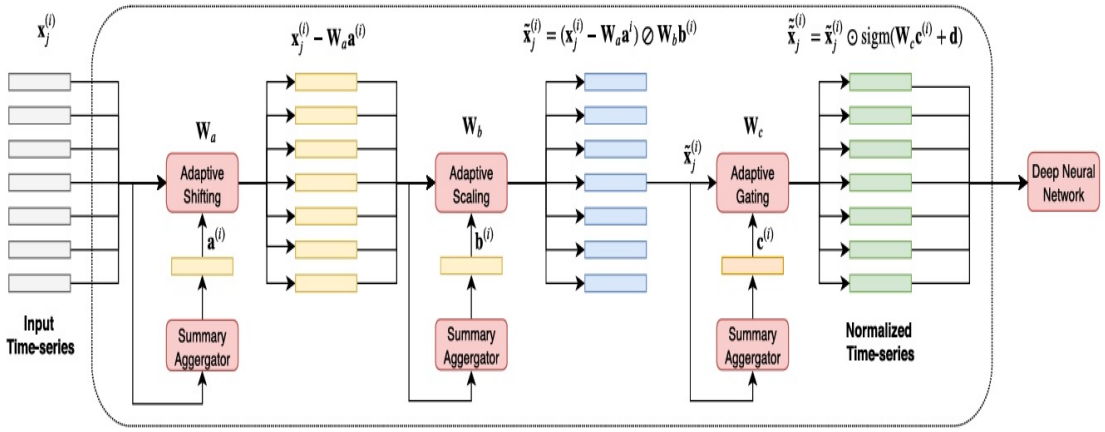


Figure 2: Adaptive Data Normalization [30]

If the data is not properly normalized, the performance of deep learning models degrades, which is especially critical for time-series data. Therefore, the use of a neural layer capable of normalizing the input time-series while considering the data distribution is recommended. This component is trained in an end-to-end manner using backpropagation. As illustrated in Figure (2), this component is defined as a sequence of three sub-layers: First Layer (Adaptive Shifting): Responsible for shifting the data to a more suitable region in the feature space (centering). Second Layer (Adaptive Scaling): Performs linear scaling of the data to increase or decrease their variance (standardization). Third Layer (Adaptive Gating): Applies non-linear gating to suppress irrelevant or non-informative features based on the task.

It is assumed that $\{X^{(i)} \in \mathbb{R}^{N\times L}; i = 1, \ldots, N\}$ represents a set of N time series, where each time series consists of L features (or measurements), each of d-dimensional size [30]. The notation $x_j^{(i)} \in \mathbb{R}^d; j = 1.2.\ldots.L$ refers to the d-dimensional features observed at time step j in the i-th time series. The goal of adaptive normalization is to learn how to normalize the measurements $x_j^{(i)}$ through appropriate shifting and scaling. In Equation (9), the operator $\oslash$ denotes the Hadamard (element-wise) division [30].

$$\tilde{x}_j^{(i)} = \left(x_j^{(i)} - \alpha^{(i)}\right) \oslash \beta^{(i)} \tag{9}$$

The Equation (10) represents the extracted time series, which is obtained by averaging all the L measurements [30].

$$a^{(i)} = \frac{1}{L}\sum_{j=1}^{L} x_j^{(i)} \in \mathbb{R}^d \tag{10}$$

This value provides an initial estimate for the mean of the current time series; therefore, it can be used to estimate the distribution from which the current time

series has been generated, allowing for the proper adjustment of the normalization method. Then, the transfer operator $\alpha^{(i)}$ is defined by a linear transformation using Equation (11) [30].

$$\alpha^{(i)} = W_a a^{(i)} \in \mathbb{R}^d \tag{11}$$

The symbol $W_a \in \mathbb{R}^{d\times d}$ is used to store the weight matrix of the time series data X. $\alpha^{(i)}$ refers to the distribution of the data in the time series X. The use of a linear transformation layer ensures that data, which might not have been properly normalized (or even normalized at all), is handled appropriately, operating in an end-to-end fashion without the need for activation functions. This layer is called the Adaptive Shifting Layer because it estimates the manner of data transfer before entering the network. This approach leverages potential correlations between different features to perform more robust normalization. After centering the data using the process described in Equation (11), the data must be scaled to an appropriate range using the scaling operator $\beta^{(i)}$. To achieve this, the standard deviation of the data is computed using the updated Equation (12) [30].

$$b_k^{(i)} = \sqrt{\frac{1}{L}\sum_{j=1}^{L}\left(x_{j,k}^{(i)} - a_k^{(i)}\right)^2}, \quad k = 1, 2, \ldots, d \tag{12}$$

L denotes the feature dimensions, and d represents the number of features; accordingly, the scaling function can similarly be defined as a linear transformation that enables the scaling of each transferred measurement individually [30].

$$\beta^{(i)} = W_b b^{(i)} \in \mathbb{R}^d \tag{13}$$

In Equation (13), $W_b \in \mathbb{R}^d$ represents the weight matrix of the scaling layer. This layer is referred to as the adaptive scaling layer because it estimates the appropriate scaling method prior to data entering the network. Finally, the data is passed through an adaptive gating layer, which is capable of discarding features that are irrelevant or uninformative for the current activity. This layer is defined by Equation (14) [30].

$$\tilde{\tilde{x}}_j^{(i)} = \tilde{x}_j^{(i)} \odot y^{(i)} \tag{14}$$

$\odot$ denotes the Hadamard product operator.

$$y^{(i)} = \mathrm{sigm}(W_c c^{(i)} + d) \in \mathbb{R}^d \tag{15}$$

The notation $\mathrm{sigm}(x) = \frac{1}{1+\exp(-x)}$ represents the sigmoid function. $W_c c^{(i)} \in \mathbb{R}^{d\times d}$ and $d \in \mathbb{R}^{d\times d}$ are the parameters of the gating layer.

$$c^{(i)} = \frac{1}{L}\sum_{j=1}^{L}\tilde{x}_j^{(i)} \in \mathbb{R}^d \tag{16}$$

$c^{(i)}$ is computed as defined in Equation (16), which, unlike the previous layers, introduces non-linearity and has the capability to eliminate normalized features. In this

way, features that are either irrelevant to the target activity or detrimental to the network's generalization ability; such as features with excessive variance are appropriately filtered out before entering the network. Overall, $a^{(i)}$, $\beta^{(i)}$ and $\gamma^{(i)}$ depend on the current local data over window i and on the global parameters W_a, W_b, W_c and d, which are learned from multiple samples in the time series $\{X^{(i)} \in \mathbb{R}^{N\times L}; i = 1 \dots N\}$ where N is the number of samples in the training dataset. The output of the normalization layer, referred to as Deep Adaptive Input Normalization, is simply obtained by feed-forwarding through the three layers as illustrated in Figure 2. During the inference process, the parameters of these layers are considered fixed; therefore, no additional training is required at inference time. All parameters of the deep model are trained directly in an end-to-end manner using gradient descent [30].

$$\Delta(W_a.W_b.W_c.d.W) = -\eta(\eta_a \frac{\partial L}{\partial W_a}.\eta_b \frac{\partial L}{\partial W_b}.\eta_c \frac{\partial L}{\partial W_c}.\frac{\partial L}{\partial W}) \tag{17}$$

In Equation (17), L denotes the loss function used for training the network, and W refers to the weights of the neural network. Additionally, separate learning rates η_a, η_b and η_c are employed for the parameters of each sub-layer. This is essential to ensure the convergence of the normalization method due to the differences in gradients produced across the various sub-layer parameters. Therefore, the normalization layer is applied to the LSTM network, enabling the LSTM to possess adaptive normalization capability.

4 Patient Treatment Stages for Constructing a Model Based on Psychologists' Behavioral Patterns

This section illustrates the detailed stages of treating individuals with PSA for the purpose of constructing a model based on psychologist behavioral patterns, as depicted in Figure (3).

After the patient visited the psychologist either through a clinical setting or as part of a research recruitment, necessary steps were taken to diagnose PSA, and relevant information was collected [11]. After selecting and evaluating the treatment group, the psychologist chooses the appropriate therapeutic approach. Specifically, using AR and sensors, treatment is administered to individuals with public speaking anxiety, as detailed in Section 4-1. Based on all therapy sessions with these individuals, a treatment model is then constructed. This model is used to train the algorithm, which is based on recurrent neural network learning, as described in Section 4-2. The goal is for the proposed method to intelligently mimic the psychologist's behavior. Finally, the algorithm is tested, and the model's error is calculated.

4.1 Treatment with the Help of a Psychologist Alongside the use of Sensors and AR

In this section, the psychologist initiates the treatment of individuals with PSA. This is performed utilizing the AR scenario to simulate a public speaking environment and employing a feedback system (a set of sensors) to capture the patientś physiological

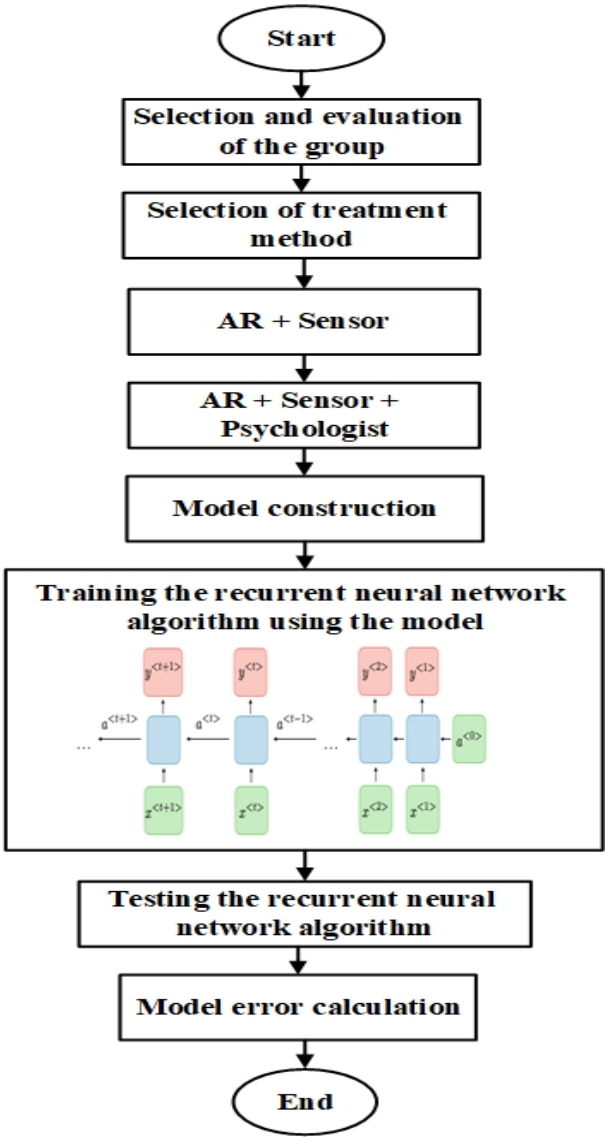


Figure 3: Stages of Treatment for Patients

signals. Accordingly, when an individual with anxiety is placed in the AR-generated public speaking environment, sensors for heart rate (ECG), temperature, and GSR are attached to the person. During the treatment, the psychologist monitors these physiological signals in real time and at specific intervals to provide appropriate interventions based on the patientś responses. In this approach, the dataset is state-based, meaning that the next reaction and treatment state are determined based on the previous actions taken by the psychologist, allowing for the execution of appropriate therapeutic interventions. Such interventions depend on the analysis of the physiological signals received from the individual; based on whether the person is calm or experiencing fear, the number of virtual audience members may be increased, decreased, or remain unchanged. The detailed steps of the treatment at this stage are as follows:

- Utilization of a feedback system

- Creation of the public speaking environment using the AR model
- Treatment by a Psychologist

4.1.1 Utilization of a Feedback System

In this section, a feedback system including an Arduino board and ECG, GSR and body temperature sensors were used to evaluate physiological signals (Measuring the level of anxiety and calm), which are connected to the Arduino board based on the microcontroller ATmega328, which has 14 digital input and output pins and a USB port. Arduino board at any moment displays the signals received by the patient's body based on sensors in the system. Then, in the process of execution, after connecting the sensors to the body, the level of anxiety and relaxation of the person with PSA is measured [11].

4.1.2 Creation of the Public Speaking Environment using the AR

Creating a lecture community based on AR technology is defined by the information written in reference [11] and its results were used in this research. The following elements are used in this scenario:

- Conference Room
- Podium
- Monitor
- Camera
- Physical Chairs

The individual stands behind the podium in the conference room for the speech. The person faces the monitor so that their line of sight is focused, allowing them to view the speech environment (the conference room) through the monitor. The camera is placed behind the monitor, enabling the individual to see the camera's output on the monitor, creating the illusion of an interactive public speaking scenario. In this scenario, the conference room, the podium, the monitor, the camera and the chairs are real, but in addition to the real audience, there is a database of virtual people based on real people's photos, models or styles to replicate on the chairs. At the beginning of the work, the person is placed in the empty hall behind the podium, and the sensors are connected to his body and he sees the environment of the speech by the monitors. The person begins to speak; then the psychologist observes the feedback of the patient's body (temperature, ECG and GSR) in the system. Based on the evaluation of these signals, it is decided that the audience in the conference room is extra, low or unchanged. For example, if the person is relaxed, a number of attendees will be added to the conference room and will continue to speak for a few minutes based on the psychologist's diagnosis, and the audience will be unchanged, and if the person's calm continues, more attendees will be added to the speech environment based on the psychologist's diagnosis. If a psychologist detects a person's anxiety, the audience will be reduced from the conference room or will continue to speak without change. The process of increasing, decreasing and unchanging the audience will continue so much during

treatment sessions that at the end, the conference room will be filled with the audience and the person will give a speech without anxiety. As mentioned, individuals are present in the speech environment both physically and virtually; therefore, in addition to real participants, virtual individuals will be added to or removed from the speech environment using the Unity software, as detailed in reference [11].

4.1.3 Treatment by a Psychologist

At this stage, the psychologist's behavioral patterns are created based on all of the patients' treatment sessions. The psychologist first familiarizes the individual with the AR-generated environment and all the tools involved, such as the feedback system, to prepare them for the treatment. Afterward, the individual is placed in the generated AR public speaking environment, and the necessary sensors are attached to their body. Initially, the psychologist evaluates the individual's level of calmness and fear based on the signals received from the sensors (ECG, temperature, and GSR). This assessment helps determine the appropriate actions during the course of treatment. If signs of fear are detected, the psychologist takes necessary measures such as reducing the number of audience members to help the individual regain calmness. Conversely, if the individual shows sufficient calmness, the psychologist proceeds with the next steps in treatment by either increasing the audience or keeping their number unchanged, depending on the individual's condition and progress. Throughout all therapy sessions and during each moment of the treatment process, the psychologist operates based on the procedures outlined in Sections 4-1-1 and 4-1-2 to treat individuals with PSA. At this stage, the behavioral patterns of the psychologist is constructed. At the beginning of the treatment, when the individual initiates their speech, the expert or psychologist continuously monitors the physiological signals and evaluates them in real time. Based on this ongoing assessment, the psychologist determines the next decision or reaction to guide the therapy process effectively. As a result, the psychologist makes decisions based on the previous state. For example, given the current state n, the psychologist determines what the next state $n+1$ should be. As a result, the psychologist conducts the treatment of the individual with PSA based on the designated therapeutic scenario. Ultimately, by analyzing the psychologist's behaviors and responses over time, a behavioral pattern is established representing the treatment process and the corresponding reaction at each state. Finally, this pattern is used to train the proposed model, based on adaptive LSTM recurrent neural network learning, described in Section 4-2, so that the proposed model can act intelligently like a psychologist.

4.2 Proposed Adaptive LSTM Recurrent Neural Network Model

The primary objective at this stage is to design a decision support system to help the psychologist treat the patient in an augmented reality environment. At this stage, the behavioral pattern created by the psychologist, in section 4-1-3, is applied to train the LSTM network so that the proposed model performs like a psychologist and can make the best decision at any moment with the least error. Therefore, the physiological signals collected in Section 4 are used to train a Recurrent Neural Network of the LSTM type. The details of the proposed model are shown in Figure (4).

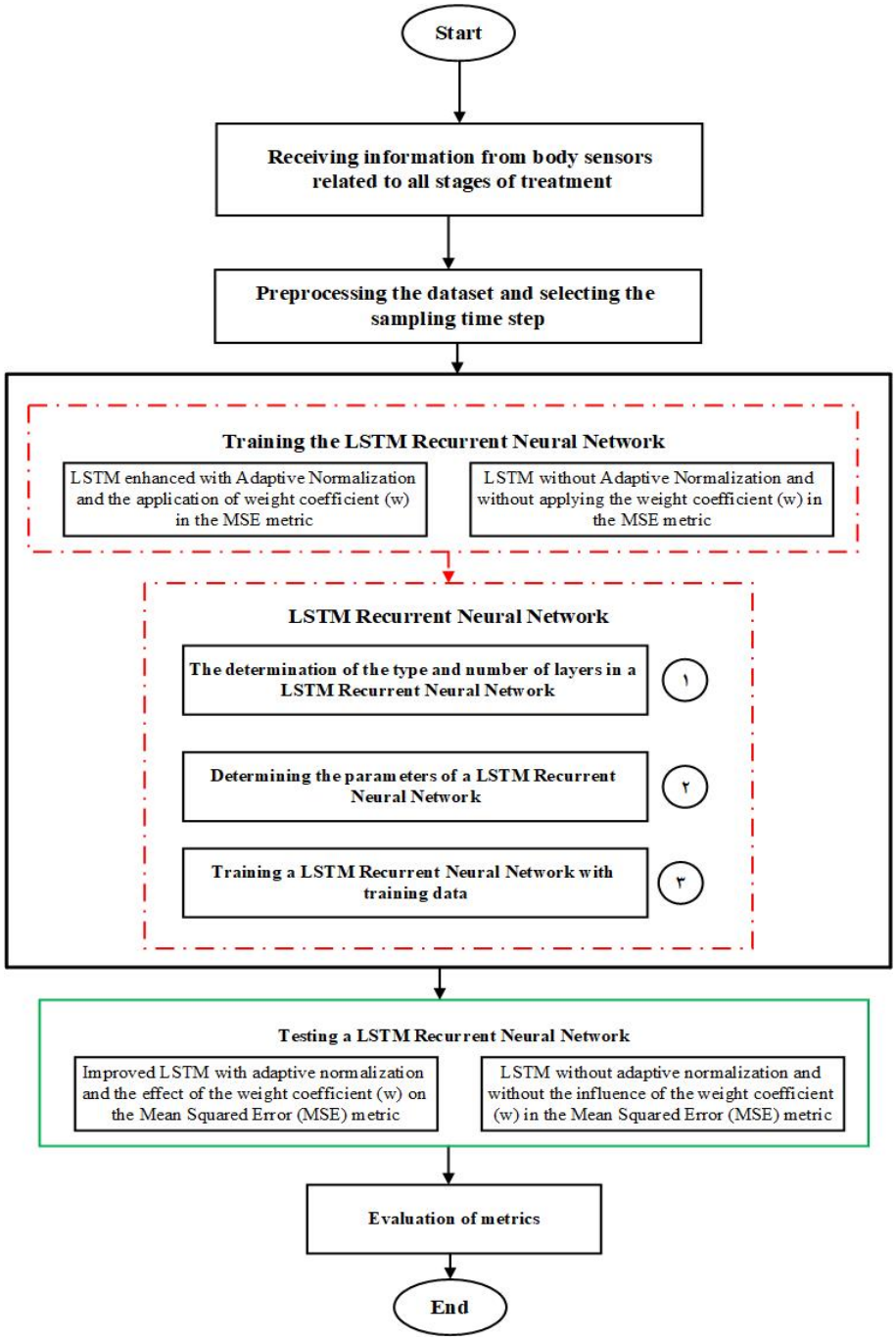


Figure 4: Proposed Model

Based on Figure 4, the proposed model consists of the following components:

4.2.1 Receiving Body Sensor Data for All Treatment Stages

The model proposed in this study consists of two parts: input and output. The input includes:

- The physiological signals obtained from the body of the individual with PSA, which were collected throughout the entire treatment duration in Step 4-1 (Psychologist + Sensors + AR Scenario), using the connected sensors (ECG, body temperature, and GSR). The input signals are considered as time-stamped data, meaning that the next state is determined based on the signals received from the patient in the previous states. In fact, throughout all treatment sessions, the received signals are analyzed to make the best possible decision for determining the next state. In this way, the signals received from the patient during all treatment sessions in stage 4-1 (psychologist + sensors + AR scenario) are collected and, along with the behavioral pattern (the psychologist's actions and intended reactions in various states), are fed into the proposed model for training. This enables the model to intelligently predict the next step similar to a psychologist but without their direct intervention so it can make the necessary decisions during the treatment process with minimal error, effectively mimicking the psychologist's behavior.
- The age of the patients
- The number of treatment sessions
- The duration of each treatment session
- The maximum number of audience
- An appropriate weighting coefficient (a number between 0 and 1), determined based on the patient's treatment progress and the psychologist's assessment. This coefficient reflects the patient's recovery percentage and is considered as the weight of the training data.

Therefore, after applying the inputs, the output of the LSTM network is obtained, considering the physiological conditions of the patient at each moment. The proposed model is a regression model; therefore, the output determines the number of audience that should be increased or decreased at each moment.

4.2.2 Data Preprocessing and Selection of Sampling Time Step

Data preprocessing consists of two stages:

- **Handling Missing Data:** In this step, the dataset is examined for any missing data, which may occur due to issues such as sensor failures or incomplete records from the psychologist. The method for correcting these missing data samples is the KNN approach. In this method, the missing values (NaN) are replaced with values from the nearest neighbor column. The nearest neighbor column refers to the column with the most similar data points based on available features [31].

- **Outlier Data Correction:** In this step, the dataset obtained will be reviewed for outlier data, which could have been recorded by the sensors or provided by the psychologist's observations. The method used for detecting outliers is the MEDIAN approach. In this method, outlier points that lie outside three times the scaled Median Absolute Deviation (MAD) are replaced using Equation (18) [32].

$$\frac{-1}{\sqrt{2} \times \text{erfcinv}\left(\frac{3}{2}\right)} \times \text{median}\left(|A - \text{median}(A)|\right) \tag{18}$$

The sampling time steps for the data are considered to be 15, 30, and 60 seconds. This approach is implemented to reduce the volume of data. Instead of using the entire dataset in the training and testing stages, sampling is performed at the specified time intervals. For example, in the 15-second movement step, sampling is conducted from the available samples at times 1, 16, 31, 46, and so on.

4.2.3 Training the LSTM Network

In this section, the training of the LSTM network, as well as the analysis and determination of its parameters, was addressed. At this stage, the LSTM network was trained twice: once without adaptive normalization and without using training data weights, using the standard MSE loss function; and again with adaptive normalization, considering training data weights, and using the proposed weighted MSE loss function. The results obtained from training both models were analyzed and compared. Thus, the proposed model was evaluated through the following stages, where in each stage, analyses were conducted with and without adaptive normalization. Removing the weight coefficient w from the dataset: In this case, the weight coefficient is removed from the dataset, and the dataset only includes the physiological information recorded by the sensors. As a result, during the training process of the LSTM network, the Mean Squared Error (MSE) loss function, as defined in Equation (19), is used due to the absence of the weight coefficient.

$$\min \sum_{i=1}^{n} \left[\text{LSTM}(x_i) - y_i\right]^2 \tag{19}$$

Considering the weight coefficient w in the dataset: At this stage, the LSTM network was trained by incorporating the weights of the training data and using the proposed weighted MSE criterion. Therefore, the weighted MSE criterion is proposed as shown in Equation (20). This criterion leads to improved results and enhanced performance of the LSTM network.

$$\min \sum_{i=1}^{n} w_i \left[\text{LSTM}(x_i) - y_i\right]^2 \tag{20}$$

Equation (20) represents the proposed weighted MSE criterion. In this equation, w_i denotes the assignment of different weights to the training data samples. In fact, the assigned weights are based on the psychologist's evaluation, which determines the

importance of the network's error for the i-th data sample. Therefore, the psychologist's assessment of the patient's treatment progress or their improvement percentage based on the implementation of the Public Speaking Anxiety Scale (PSAS) by Bartholomay and Houlihan [33] at the end of each session. This assessment is reflected through the assigned weight values. When the network is being trained, the i-th training data instance representing a sequence of the desired input features for the network is denoted by the parameter x_i. Then, the LSTM network predicts the desired output based on the input sequence x_i. Also, y_i refers to the target or the sequence of desired outputs determined based on the psychologist's judgment. The goal of the proposed weighted MSE is to minimize this function over n training samples meaning that the LSTM output should match the psychologist's decision represented by y_i, or at least the difference between the two should be as small as possible.

4.2.3.1 Parameter Tuning and Evaluation in the Proposed Adaptive LSTM Network Model

To utilize the model effectively, it is essential to determine key influential parameters such as the number of layers, the number of hidden units, learning rate, and the dropout rate, in order to enhance the performance of the network. To achieve this, several scenarios combining different parameter values were evaluated, and ultimately, the optimal values for the parameters were determined. For this purpose, in each iteration, one parameter was varied while the others were kept constant. The defined scenarios are as follows:
Scenario 1: change the number of layers of the LSTM network (based on values 1, 2 and 3)
Scenario 2: change the number of hidden units (based on values of 30, 60 and 90), taking into account the best value of the LSTM network layer obtained in Scenario 1.
Scenario 3: change the Learning Rate Based on Values 0.001, 0.01, and 0.02
Scenario 4: change the Dropout Rate with Values 0.1, 0.2, and 0.5
For instance, in tuning the parameters such as learning rate, number of hidden units, total number of layers, and dropout rate, Scenario 1 focuses solely on varying the number of layers with values 1, 2, and 3. The goal of this scenario is to identify the optimal number of layers that yields the best performance for the proposed LSTM network. In this case, the remaining parameters in each stage are kept constant with the following values: number of hidden units = 30, learning rate = 0.001, and dropout rate = 0.1. Finally, after execution, the optimal value for the number of LSTM layers is determined to be 1, and this value is used in Scenario 2 for further evaluations. In Scenario 2, the number of hidden units is varied using the values 30, 60, and 90, while the number of LSTM layers is fixed based on the optimal value obtained from Scenario 1. The remaining parameters are kept constant with the values: learning rate = 0.001 and dropout = 0.1. Then, the optimal value for the number of hidden units is determined to be 90, and this value is considered in Scenario 3. In Scenario 3, the optimal values for the number of layers and number of hidden units obtained from Scenarios 1 and 2 are used. Then, the learning rate parameter is varied across values of 0.001, 0.01, and 0.02, while keeping the Dropout rate fixed at 0.1. As a result, the optimal learning rate is determined to be 0.01 in this stage. Finally, in

Scenario 4, the optimal values of the parameters number of layers, number of hidden units, and learning rate obtained from the previous three scenarios are held constant. The Dropout parameter is then varied across values of 0.1, 0.2, and 0.5, in order to determine the best value, which is found to be 0.5.

The structure of the adaptive LSTM recurrent neural network ultimately used in this study is illustrated in Figure (5).

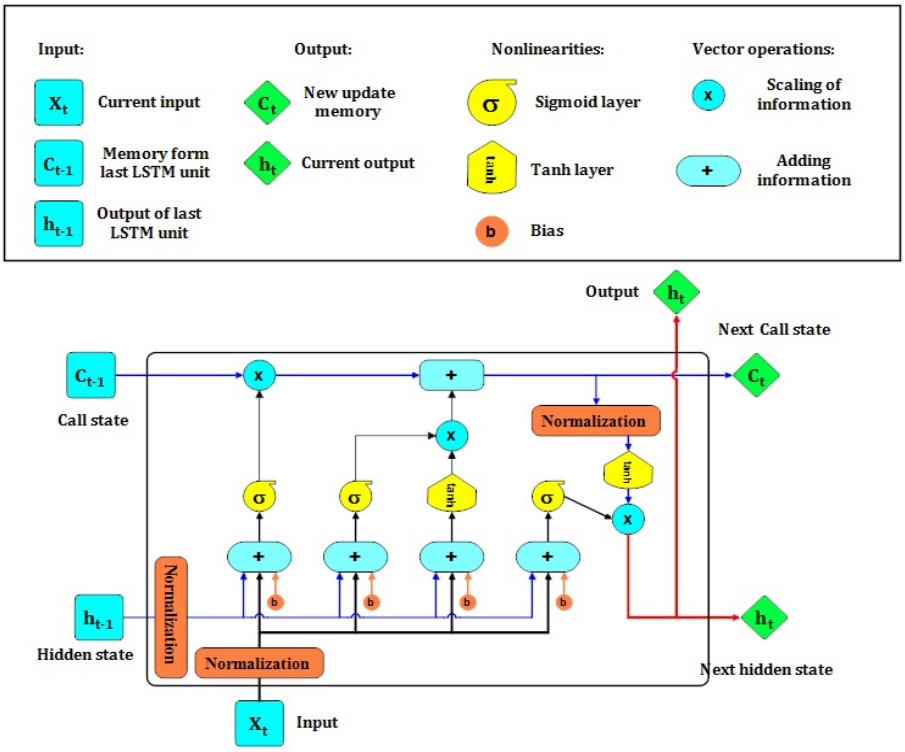


Figure 5: The Structure of the LSTM Recurrent Neural Network

4.2.4 Testing the LSTM Network

The proposed model, based on the data collected in Step 4-1, is trained to analyze the signals received from the patient's body at any moment in time; also to make the right decision and suggest the next reaction during the treatment in order to treat and reduce anxiety; that is, to predict the next movement in the treatment direction like a psychologist without its intervention; also to function like a psychologist and to properly control the number of audience. In this stage, similar to the training phase, the algorithm is tested in the following four configurations:

1. "LSTM network without adaptive normalization and without the influence of the weight coefficient (W) in the MSE criterion."
2. "Improved LSTM network with adaptive normalization and the influence of the weight coefficient (W) in the MSE criterion."
3. "LSTM network without adaptive normalization but with the influence of the weight coefficient (W) in the MSE criterion."
4. "Improved LSTM network with adaptive normalization and without the influence

of the weight coefficient (W) in the MSE criterion."
The proposed model will be trained based on the data collected during the treatment phase 4-1, so that it can accurately respond to the physiological signals received from the patient's body at any given moment in time, in order to determine the most appropriate output. This is because the goal is for the proposed model to function like a psychologist and make optimal decisions throughout the treatment process with minimal error, accurately controlling the number of audience members in response to the patient's varying physiological conditions. In Section 5, the calculation and analysis of the metrics are discussed.

5 Experimental Results and Discussion

In this section, MATLAB R2025a was used to evaluate the criteria.

5.1 Datasets

The dataset used in this study is derived from physiological feedback collected via sensors (heart rate, body temperature, and Galvanic Skin Response) attached to 15 participants during Public Speaking Anxiety treatment sessions. The input data includes parameters such as age, treatment duration, number of sessions, number of audience members, and a weighted coefficient. All data were recorded at one-second intervals [11]. A portion of the collected dataset is presented in Table 1.

In Table 1, a sample of data from the seventh session for a nineteen-year-old patient with a recovery percentage of 0.58, measured in minutes, is presented. For data recording, the treatment duration and number of sessions were fixed for all patients (45-minute sessions with a maximum of 8 sessions). Out of 15 patients with PSA who underwent treatment in stages 4-1, 3 were not treated, while 1 patient completed treatment in 5 sessions, 2 patients in 6 sessions, 2 patients in 7 sessions, and 7 patients in 8 sessions [11]. In this study, out of the 15 patients, the data from 14 individuals were used as training data and the data from one individual were used as test data in each stage. To better evaluate the results of the proposed model, the data from all 15 participants were used in both training and testing phases.

5.2 Model Evaluation Metrics

The proposed model presented in this study was evaluated using the metrics defined in Equations (21) to (27). The evaluation metrics include proposed Weighted MSE (New MSE), MSE, Normalized MSE (NMSE), Root MSE (RMSE), Normalized Root MSE (NRMSE), Mean Absolute Error (MAE), and the Correlation Coefficient (R). The evaluation metrics have been analyzed as follows:

- Proposed Weighted MSE Metric: This metric was fully explained in Section 4.2.3.1 and is based on Equation (20). Proposed Weighted MSE Metric is defined

Table 1: A Sample of Data from the Seventh Session for a 19-Year-Old Patient with a Recovery Rate of 0.58

Time (min)	**Number of Audience**	**GSR**	**Temperature**	**Ecg**
0	0	667.78	31.12	76
1	0	672.276	31.153	78
2	0	677.923	31.153	78
3	0	674.804	31.128	76
4	0	674.944	31.131	76
5	11	686.370	31.260	75
6	11	687.184	31.329	77
7	11	689.730	31.498	75
8	11	690.886	31.418	73
9	11	691.856	31.425	73
10	24	701.880	31.510	74
11	24	715.059	31.529	72
12	24	723.223	31.525	73
13	24	719.203	31.519	75
14	24	723.745	31.523	74
15	34	739.370	31.550	74
16	34	739.606	31.587	72
17	34	740.002	31.562	74
18	34	751.645	31.565	72
19	34	752.217	31.573	73
20	42	758.010	31.590	74
21	42	765.511	31.641	74
22	42	779.118	31.675	74
23	42	760.663	31.649	75
24	42	762.244	31.631	73
25	42	777.922	31.690	75
28	42	763.101	31.624	72
29	42	780.527	31.615	75
30	42	771.789	31.645	74
31	42	781.074	31.598	72
32	46	783.460	31.710	73
33	46	811.299	31.842	74
34	46	813.239	31.868	75
35	46	789.766	31.830	73
36	46	783.984	31.813	72
37	46	809.298	31.711	72
38	46	807.418	31.742	75
39	46	791.482	31.753	75
40	56	813.790	31.890	73
41	56	814.577	31.911	74
42	56	817.558	32.032	74
43	56	828.344	31.955	73
44	56	815.074	32.039	74
45	56	837.410	32.050	73

in Equation (21).

$$\text{New MSE} = \sum_{i=1}^{n} \left[w_i \left(\text{LSTM}\left(x_i \right) - y_i \right) \right]^2 \tag{21}$$

- MSE: It is a method for estimating the error magnitude, which essentially measures the difference between the estimated values and the actual values. MSE is defined in Equation (22) [34, 35].

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} \left(Y_i - \hat{Y}_i \right)^2 \tag{22}$$

In Equation (22), Y_i represents the actual output values, and $\hat{Y}_i$ represents the predicted output values. In a dataset of size n, where $\frac{1}{n} \sum_{i=1}^{n}$ performs the averaging operation and $\left(Y_i - \hat{Y}_i \right)^2$ calculates the squared error for each data point, the MSE is the average of the squared errors.

- NMSE: To measure the average relative dispersion and reflect random errors, Equation (23) is used [36].

$$\text{NMSE} = \sqrt{\frac{\text{MSE}}{\frac{1}{n-1} \sum_{i=1}^{n} \left| Y_i - \left(\frac{1}{n} \sum_{i=1}^{n} Y_i \right) \right|^2}} \tag{23}$$

- RMSE: This parameter is primarily used to calculate the difference between the predicted values generated by the model and the observed values. When the RMSE for a specific model is lower than that of another model, it can be concluded that the proposed model has higher accuracy. RMSE is defined in Equation (24) [37].

$$\text{RMSE} = \sqrt{\text{MSE}} \tag{24}$$

- NRMSE: When the RMSE value is divided by the dependent variable data range, it is called the NRMSE. This metric is suitable for comparing different models. It is worth noting that an NRMSE value below 10% indicates a highly accurate model, between 10% and 20% indicates a reasonably good model, between 20% and 30% reflects moderate accuracy, and above 30% suggests a weak model. This metric is calculated based on Equation (25) [38].

$$\text{NRMSE} = \sqrt{\frac{\text{RMSE}}{\frac{1}{n-1} \sum_{i=1}^{n} \left| Y_i - \left(\frac{1}{n} \sum_{i=1}^{n} Y_i \right) \right|^2}} \tag{25}$$

- MAE: In the context of model error analysis, MAE represents the average difference between the actual and predicted values across all training samples. For n test samples, and for each actual output value Y_i and predicted output value $\hat{Y}_i$, the MAE is defined by Equation (26) [35, 39].

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} \left| Y_i - \hat{Y}_i \right| \tag{26}$$

- R: The correlation between the estimated results of the model and the actual data is calculated based on Equation (27). In this equation, Y_i represents the actual output values, and $\hat{Y}_i$ represents the predicted output values. R is defined in Equation (27) [40].

$$R = \frac{1}{n-1} \sum_{i=1}^{n} \left(\frac{\overline{Y_i} - \text{mean}(\overline{Y_i})}{\text{std}(Y_i)} \right) \left(\frac{\hat{Y}_i - \text{mean}(\hat{Y}_i)}{\text{std}(\hat{Y}_i)} \right) \tag{27}$$

5.3 Scenario Evaluation

Based on the sampling time steps of the data explained in Section 4-2-2, the evaluation of the obtained results is presented according to the following scenarios:

- Evaluation of the proposed LSTM neural network model at sampling time steps of 15, 30 and 60 seconds from the constructed dataset (without adaptive normalization and with fixed network parameters).
- Evaluation of the proposed LSTM neural network model (with changes in network parameters, without adaptive normalization, and with adaptive normalization).

5.3.1 Evaluation of the Proposed Model at Time Steps of 15, 30 and 60 Seconds

As explained in Section 4, using the weight coefficient W helps reduce the error during network training. In this phase, the network parameters (number of layers = 1, number of hidden units = 30, learning rate = 0.001, and dropout = 0.5) were kept fixed. The results were calculated and analyzed based on the sampling time steps of 15, 30 and 60 seconds, without adaptive normalization. Comparison of the results of the evaluation indicators with the time step of 15, 30 and 60 seconds, without the weighted coefficient W and taking into account the coefficient W is presented in Table (2).

Table 2: Comparison of Evaluation Metrics Across Three Sampling Time Steps (15, 30 and 60 Seconds), With and Without Weight Coefficient W (Fixed Network Parameters and Without Adaptive Normalization

Metric	15s (With W)	30s (With W)	60s (With W)	15s (Without W)	30s (Without W)	60s (Without W)
MSE	0.02898	0.03074	0.02673	0.03338	0.03504	0.03426
NMSE	0.42420	0.46118	0.40583	0.49921	0.51974	0.51330
RMSE	0.15999	0.16607	0.15354	0.16589	0.16775	0.16768
NRMSE	0.59223	0.61835	0.57237	0.61561	0.62189	0.62278
MAE	0.11639	0.12149	0.11282	0.12203	0.12494	0.12492
R^2	0.93966	0.93379	0.93783	0.90598	0.93331	0.93289

As shown in Table (2), (3) and (4), the proposed model demonstrates better performance across all evaluation metrics when the weight coefficient W is considered. The comparison of results for sampling time steps of 15, 30, and 60 seconds shows that the value of the weighted loss function (using coefficient W) at the 60-second time step is

0.02673. Meanwhile, in the 15- and 30-second sampling time steps, the weighted loss function was higher by 0.00225 and 0.00401, respectively, compared to the 60-second step. This highlights the greater impact of the weight coefficient W during training when using data sampled at 60-second intervals. Furthermore, the results indicated that using a 60-second sampling time step led to improved performance of the LSTM recurrent neural network and required less memory for storing the results. Based on the results presented in Table (2), which indicate superior network performance using a 60-second sampling time step (with weight coefficient W), the subsequent analysis focuses on evaluating the performance of the proposed model using 60-second data sampling, both with and without adaptive normalization. Additionally, variations in the LSTM recurrent neural network parameters including the number of layers, number of hidden layers, learning rate, and dropout rate were investigated in order to determine the optimal configuration for enhancing the network's performance.

5.3.2 Evaluation of the Proposed LSTM Recurrent Neural Network Model (Parameter Tuning with and without Adaptive Normalization)

In this section, the proposed model is evaluated based on changes in network parameters (number of layers, number of hidden units, learning rate, and dropout rate), both with and without adaptive normalization.

5.3.2.1 Evaluation of the Proposed LSTM Recurrent Neural Network Model by Varying the Number of Layers

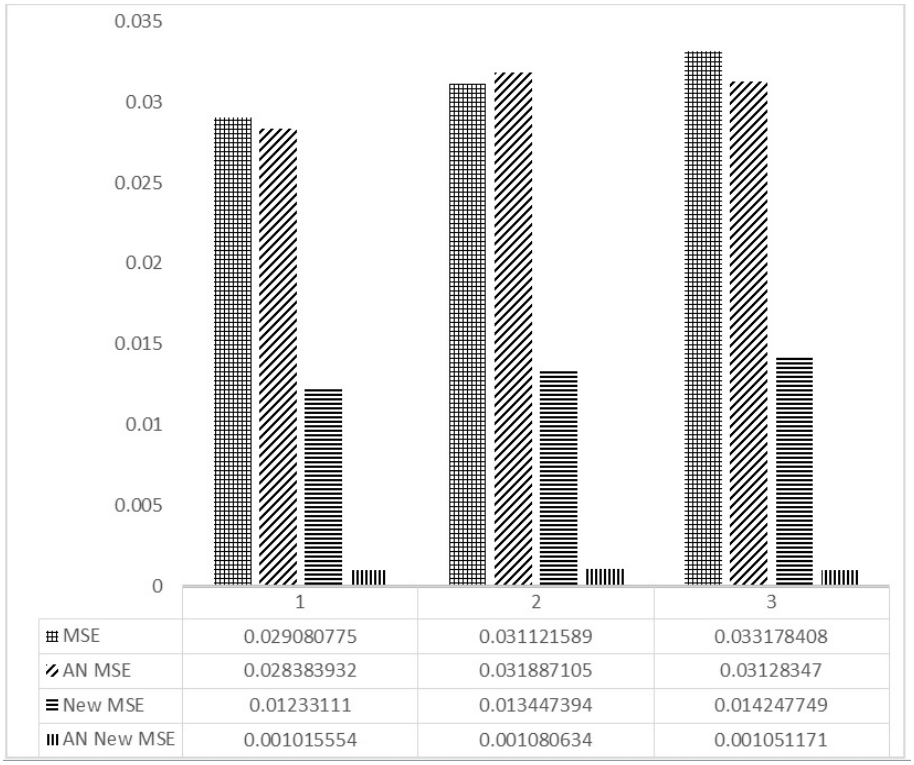
To simulate the LSTM recurrent neural network, the number of LSTM layers was varied and set to 1, 2 and 3, while keeping the remaining parameters fixed (number of hidden units = 30, learning rate = 0.001, and dropout rate = 0.1). The results were evaluated once with "adaptive normalization" and once again "without adaptive normalization". Table (3) shows the results of the MSE and the Weighted MSE. In Table (3), (4), (5) and (6), "New MSE" refers to the Weighted MSE.

Table 3: Evaluation of the Proposed LSTM Recurrent Neural Network Model with Varying Number of Layers (With and Without Adaptive Normalization)

Normalization	Variable	Num of Layer	Num Hidden Units	Initial Learn Rate	Dropout Value	MSE	New MSE
With Adaptive	Num of Layer	1	30	0.001	0.1	0.028384	0.001016
		2	30	0.001	0.1	0.031887	0.001081
		3	30	0.001	0.1	0.031283	0.001051
Without Adaptive	Num of Layer	1	30	0.001	0.1	0.029081	0.012331
		2	30	0.001	0.1	0.031122	0.013447
		3	30	0.001	0.1	0.033178	0.014248

In Table (3), the details of the change in the number of layers are presented. The results show that considering a single layer with adaptive normalization improves the weighted loss function evaluation metric by 6% and 3% compared to 2 and 3 layers, respectively. As a result, it can be concluded that using a single layer provides better performance in detecting the entry and exit of attendees, which is generally compared in

Figure (6). In Figure (6), (7), (8) and (9), "AN" stands for Adaptive Normalization, and "New MSE" refers to the Weighted Mean Squared Error. The horizontal axis represents MSE, AN MSE, New MSE, and AN New MSE, while the vertical axis indicates the error values.



	1	2	3
MSE	0.029080775	0.031121589	0.033178408
AN MSE	0.028383932	0.031887105	0.03128347
New MSE	0.01233111	0.013447394	0.014247749
AN New MSE	0.001015554	0.001080634	0.001051171

Figure 6: Comparison of MSE and Weighted MSE under Adaptive and Non-Adaptive Normalization with Varying Number of LSTM Layers

Considering the results obtained in Figure (6) and the optimal performance of the network with a single layer, the performance of the proposed LSTM neural network model was further examined by altering other parameters (such as the number of hidden units, learning rate, and dropout rate) in order to enhance the network's detection capability.
In the next section, the performance of the network is evaluated by changing the number of hidden units.

5.3.2.2 Evaluation of the Proposed LSTM Recurrent Neural Network Model with Changes in the Number of Hidden units

In section 5-3-2-1, it was observed that considering a single layer along with adaptive normalization resulted in better network performance. In this section, the network's performance is evaluated by considering the previously optimized conditions (number of layers = 1) and keeping the other parameters fixed (learning rate = 0.001 and dropout = 0.1). However, the number of hidden units was changed to 30, 60, and 90, with and without adaptive normalization. Table (4) presents the results of this evaluation.

Table 4: Evaluation of the Proposed LSTM Recurrent Neural Network Model with Changes in the Number of Hidden units (With and Without Adaptive Normalization)

Normalization	Variable	Num of Layers	Num Hidden Units	Initial Learn Rate	Dropout Value	MSE	New MSE
With Adaptive	Num Hidden Units	1	30	0.001	0.1	0.028384	0.001016
		1	60	0.001	0.1	0.028008	0.000991
		1	90	0.001	0.1	0.027233	0.000980
Without Adaptive	Num Hidden Units	1	30	0.001	0.1	0.029081	0.012331
		1	60	0.001	0.1	0.029000	0.012267
		1	90	0.001	0.1	0.030960	0.012913

As shown in Table (4), by considering a single layer and a hidden units count of 90, the weighted loss function criterion improved in the adaptive normalization mode. This improvement is 0.1% compared to a hidden units count of 60 and 0.3% compared to 30 hidden units. A general comparison of these parameters is illustrated in Figure (7).

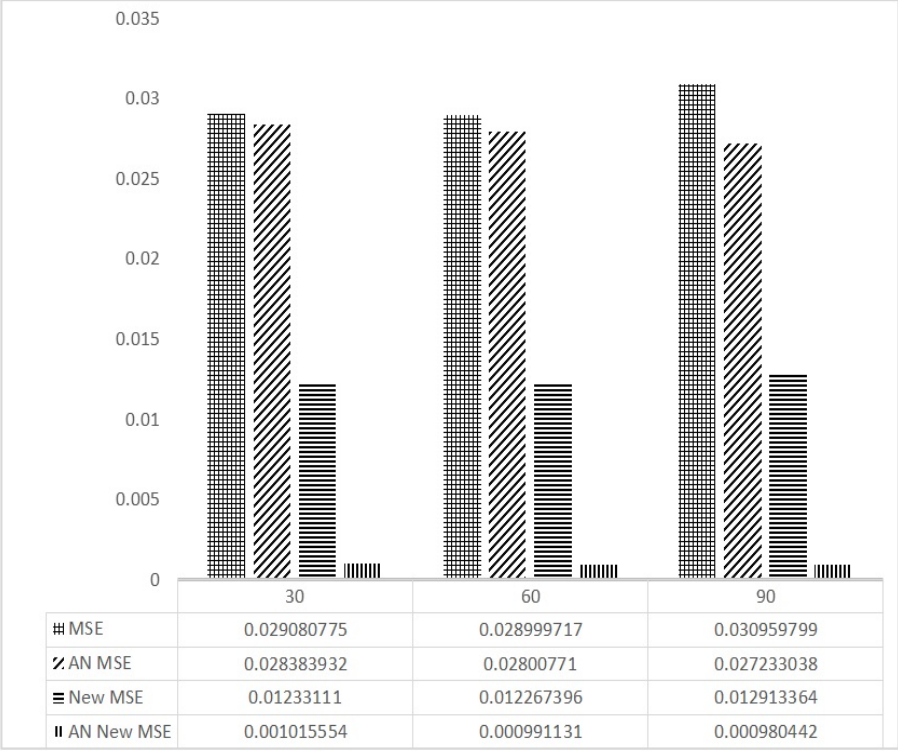


Figure 7: Comparison of MSE and Weighted MSE in Adaptive and Non-Adaptive Normalization Modes with Changes in the Number of Hidden units

Based on the results obtained from Figure (7) and the optimal performance of the network with a single layer and 90 hidden units, the performance of the proposed LSTM neural network model was further examined by altering other parameters (learning rate and dropout rate) in order to enhance the network's detection capability. The next

section evaluates the network's performance with changes in the learning rate.

5.3.2.3 Evaluation of the Proposed LSTM Neural Network Model with Changes in the Learning Rate

Based on the results obtained in the previous sections, when the number of network layers is set to one and the number of hidden units is 90, the network's performance with the given dataset, under adaptive normalization, leads to a further reduction in detection errors. Now, in this section, the impact of the learning rate (with variable values of 0.001, 0.01, and 0.02) on the optimal network detection for determining the number of attendees and reducing anxiety is examined. Table (5) presents the results of this evaluation.

Table 5: Evaluation of the Proposed LSTM Recurrent Neural Network Model with Changes in the Learning Rate (With and Without Adaptive Normalization)

Normalization	Variable	Num of Layers	Num Hidden Units	Initial Learn Rate	Dropout Value	MSE	New MSE
With Adaptive	Initial Learn Rate	1	90	0.001	0.1	0.027233	0.000980
		1	90	0.010	0.1	0.028016	0.000908
		1	90	0.020	0.1	0.030403	0.000995
Without Adaptive	Initial Learn Rate	1	60	0.001	0.1	0.030960	0.012913
		1	60	0.010	0.1	0.025769	0.010955
		1	60	0.020	0.1	0.031147	0.013046

Table (5) shows the details of the changes in the learning rate values. As the results indicate, by considering a single layer, 90 hidden units, and a learning rate of 0.01, the weighted loss function criterion improved in the adaptive normalization mode compared to the mode without adaptive normalization. The results of the learning rate variations are presented in Figure (8). Based on the provided results, it can be stated that setting the learning rate to 0.01 leads to better network performance. In contrast, a learning rate of 0.01 improves network detection by 0.7% compared to a learning rate of 0.001, and by 0.9% compared to a learning rate of 0.02.

Based on the results obtained from Figure (8) and the optimal performance of the network with a single layer, 90 hidden units, and a learning rate from Figure (8)of 0.01, the performance of the proposed LSTM neural network model was further evaluated by modifying the final parameter dropout rate in order to enhance the network's detection capability. In the next section, the network's performance is evaluated by changing the dropout rate.

5.3.2.4 Evaluation of the Proposed LSTM Neural Network Model with Changes in the Dropout Rate

In neural networks, there is a risk of overfitting due to the large number of layers. To prevent this, dropout is used. In this approach, during each training iteration, a certain number of nodes are randomly excluded from the training process with a

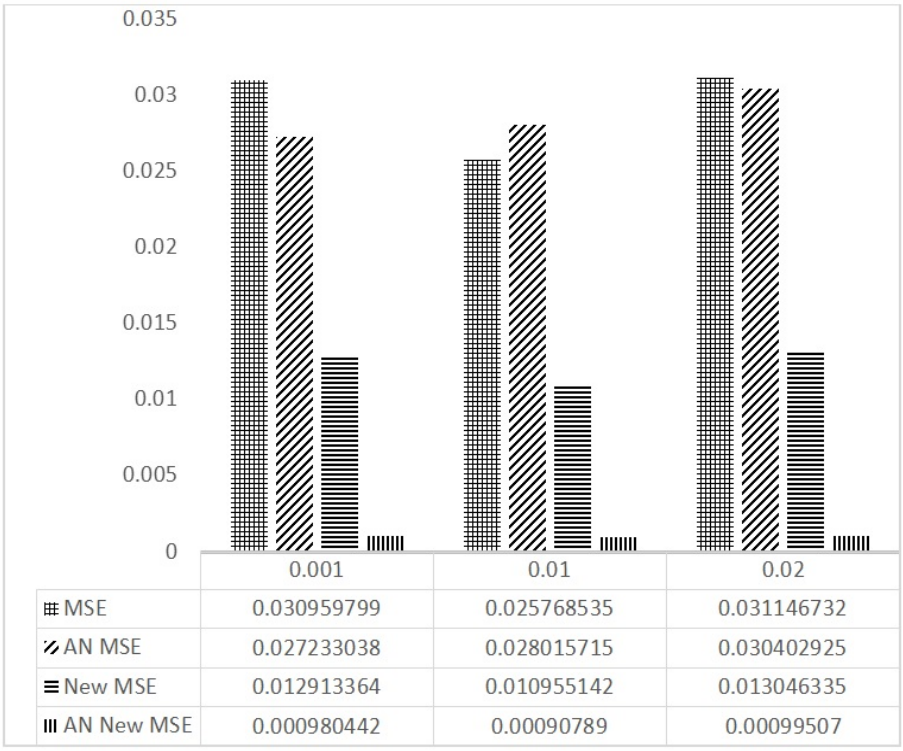


Figure 8: Comparison of MSE and Weighted MSE in Adaptive and Non-Adaptive Normalization Modes with Changes in the Learning Rate

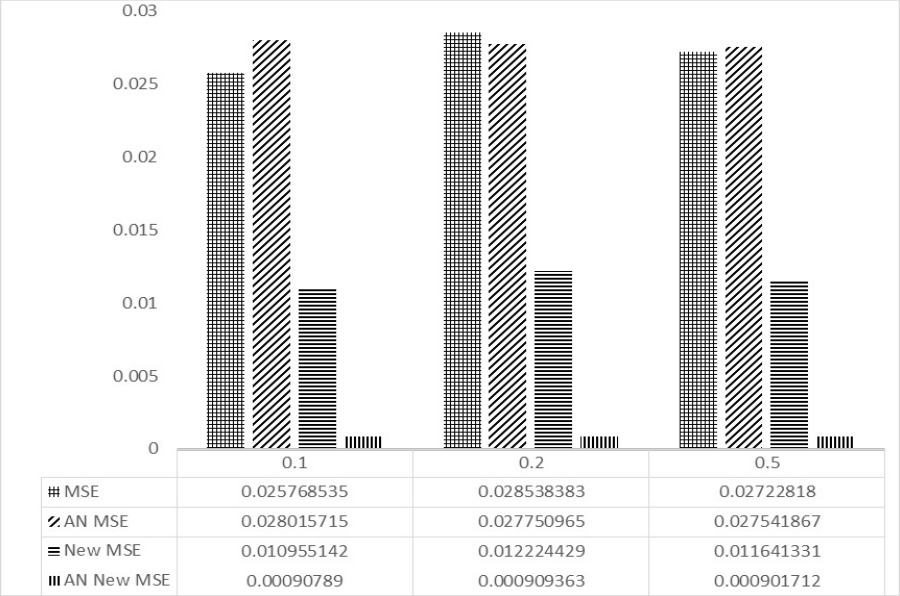
specified probability. This significantly prevents nodes from mimicking each other's behavior and reduces the risk of overfitting. For this reason, various values of this parameter were also evaluated in this study. Based on the results obtained in the previous sections, when the number of network layers is set to one, the number of hidden units is 90, and the learning rate is 0.01, the network performance with the given dataset is optimized under adaptive normalization. Now, in this section, the impact of the dropout rate (with variable values of 0.1, 0.2, and 0.5) is examined. Table (6) presents the results of this evaluation.

Table 6: Evaluation of the Proposed LSTM Recurrent Neural Network Model with Changes in the Dropout Rate (With and Without Adaptive Normalization)

Normalization	Variable	Num of Layers	Num Hidden Units	Initial Learn Rate	Dropout Value	MSE	New MSE
With Adaptive	Dropout Value	1	90	0.01	0.1	0.028016	0.000908
		1	90	0.01	0.2	0.027751	0.000909
		1	90	0.01	0.5	0.027542	0.000902
Without Adaptive	Dropout Value	1	60	0.01	0.1	0.025769	0.010955
		1	60	0.01	0.2	0.028538	0.012224
		1	60	0.01	0.5	0.027228	0.011641

Table (6) shows the details of the changes in the dropout parameter values. As the

results indicate, by considering a single layer, 90 hidden units, a learning rate of 0.01, and a dropout rate of 0.5, the weighted loss function criterion improved in the adaptive normalization mode compared to the mode without adaptive normalization. A general comparison of these parameters is illustrated in Figure (9).

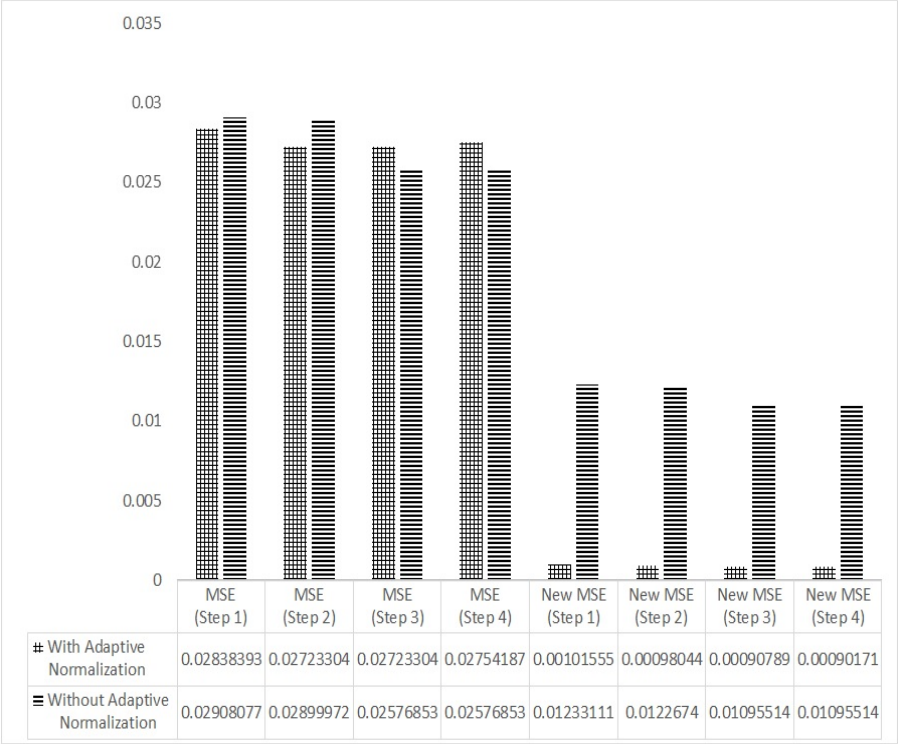


	0.1	0.2	0.5
MSE	0.025768535	0.028538383	0.02722818
AN MSE	0.028015715	0.027750965	0.027541867
New MSE	0.010955142	0.012224429	0.011641331
AN New MSE	0.00090789	0.000909363	0.000901712

Figure 9: Comparison of MSE and Weighted MSE in Adaptive and Non-Adaptive Normalization Modes with Changes in the Dropout Parameter

In the various evaluations presented in Figure (9), it was determined that in the case of adaptive normalization with a dropout rate of 0.5, the best result was achieved in the network's performance. Additionally, in this case, the network's performance with adaptive normalization showed an improvement of 91.7% compared to the mode without adaptive normalization. Therefore, it can be stated that one of the most important and influential layers in the neural network is the dropout layer, which, along with the newly introduced evaluation metric, has contributed to the high efficiency of the network. Based on the results obtained, by considering the number of layers = 1, hidden units = 90, learning rate = 0.01, and dropout = 0.5, the weighted loss function criterion improved in the "Adaptive Normalization" mode compared to the "Without Adaptive Normalization" mode. A comparison of MSE and weighted MSE, considering the optimal parameter values for the LSTM recurrent neural network, is shown in Figure (10).

In Figure (10), "New MSE" refers to the weighted loss function, which is compared with MSE for all optimal parameter values. In Step 1, the optimal number of layers (equal to 1) is considered; in Step 2, the optimal number of hidden units (equal to 90); in Step 3, the optimal learning rate (equal to 0.01); and in Step 4, the optimal dropout rate (equal to 0.5). As shown in Figure (10), the weighted MSE criterion, with adaptive normalization, has decreased relative to the MSE in all steps.



	MSE (Step 1)	MSE (Step 2)	MSE (Step 3)	MSE (Step 4)	New MSE (Step 1)	New MSE (Step 2)	New MSE (Step 3)	New MSE (Step 4)
With Adaptive Normalization	0.02838393	0.02723304	0.02723304	0.02754187	0.00101555	0.00098044	0.00090789	0.00090171
Without Adaptive Normalization	0.02908077	0.02899972	0.02576853	0.02576853	0.01233111	0.0122674	0.01095514	0.01095514

Figure 10: Comparison of MSE and Weighted MSE in Adaptive and Non-Adaptive Normalization Modes with the Optimal Parameter Values for the Network

Conclusion and Future Work

In this study, we examined PSA as one of the most common types of social anxiety and attempted to automate its treatment process using recurrent neural network-based approaches. Given the time-consuming process of traditional therapy by psychologists and the challenges of rapid access to healthcare services, a model based on an adaptive LSTM recursive neural network with the new weighted loss function was proposed. This proposed model was able to provide treatment suggestions as well as a psychologist. Experimental results showed that the application of adaptive normalization along with proper adjustment of network parameters has significantly improved the performance of the proposed model compared to the version without adaptive normalization. Furthermore, the mean square error of the adaptive LSTM model with the proposed weighted loss function was 0.000902, which is almost in line with the psychologist's opinion. Therefore, the proposed model can facilitate and accelerate the patient's treatment in accordance with the psychologist's opinion, which justifies and even recommends its use. These findings demonstrate the high potential of adaptive LSTM-based recurrent neural network models in simulating psychological assessment processes and facilitating the treatment of disorders such as PSA. In the future, the proposed model can be applied to the treatment of other psychological disorders, and by integrating it with more advanced neural network architectures, the accuracy and efficiency of the model can be further enhanced.

References

[1] A. Frantz and K. Grosenbacher, "A mixed method study on the effectiveness of using virtual reality to improve adolescent public speaking," *J. Student Res.*, vol. 10, no. 4, 2021. [Online]. Available: `https://www.jsr.org/hs/index.php/path/article/view/2098`

[2] J. Salkevicius, A. Miskinyte, and L. Navickas, "Cloud based virtual reality exposure therapy service for public speaking anxiety," *Information*, vol. 10, no. 2, p. 62, 2019, doi: 10.3390/info10020062.

[3] M. Wrzesien, J.-M. Burkhardt, M. Alcañiz, and C. Botella, "How technology influences the therapeutic process: A comparative field evaluation of augmented reality and in vivo exposure therapy for phobia of small animals," in *Proc. 13th IFIP TC 13 Int. Conf. Human-Computer Interaction (INTERACT)*, Lisbon, Portugal, 2011, pp. 523–540, doi: 10.1007/978-3-642-23774-4_43.

[4] S. Kahlon, P. Lindner, and T. Nordgreen, "Gamified virtual reality exposure therapy for adolescents with public speaking anxiety: A four-armed randomized controlled trial," *Front. Virtual Reality*, vol. 4, p. 1240778, 2023, doi: 10.3389/frvir.2023.1240778.

[5] H. Zhou, Y. Fujimoto, M. Kanbara, and H. Kato, "Effectiveness of virtual reality playback in public speaking training," in *Companion Publ. 2020 Int. Conf. Multimodal Interaction (ICMI Companion)*, New York, NY, USA: ACM, 2020, pp. 462–466, doi: 10.1145/3395035.3425220.

[6] R. V. Boas, L. V. O. Lima, G. Zanini, and P. R. Henriques, "Artefact of augmented reality to support the treatment of specific phobias," in *Trends and Innovations in Information Systems and Technologies*, vol. 2, A. Rocha et al., Eds. Cham, Switzerland: Springer, 2020, pp. 181–187, doi: 10.1007/978-3-030-45691-7.

[7] A. Miloff, P. Lindner, W. Hamilton, L. Reuterskjöld, G. Andersson, and P. Carlbring, "Single-session gamified virtual reality exposure therapy for spider phobia vs. traditional exposure therapy: Study protocol for a randomized controlled non-inferiority trial," *Trials*, vol. 17, p. 60, 2016, doi: 10.1186/s13063-016-1171-1.

[8] S. Kahlon, P. Lindner, and T. Nordgreen, "Virtual reality exposure therapy for adolescents with fear of public speaking: A non-randomized feasibility and pilot study," *Child Adolesc. Psychiatry Ment. Health*, vol. 13, p. 47, 2019, doi: 10.1186/s13034-019-0307-y.

[9] B. O. Rothbaum, P. Anderson, E. Zimand, L. Hodges, D. Lang, and J. Wilson, "Virtual reality exposure therapy and standard (in vivo) exposure therapy in the treatment of fear of flying," *Behav. Ther.*, vol. 37, no. 1, pp. 80–90, 2006, doi: 10.1016/j.beth.2005.04.004.

[10] M. Krijn, P. M. G. Emmelkamp, R. P. Olafsson, and R. Biemond, "Virtual reality exposure therapy of anxiety disorders: A review," *Clin. Psychol. Rev.*, vol. 24, no. 3, pp. 259–281, 2004, doi: 10.1016/j.cpr.2004.04.001.

[11] F. Abdollahzadeh, S. R. Kamel Tabbakh Farizani, Y. Forghani, M. Niazi Torshiz, and H. Abdollahzadeh, "Comparing the effectiveness of cognitive behavioral therapy and augmented reality technology on physical factors and public speaking anxiety," *Health Psychol. J.* [in Persian], vol. 12, no. 46, pp. 149–170, 2023.

[12] G. Lampropoulos, E. Keramopoulos, and K. Diamantaras, "Enhancing the functionality of augmented reality using deep learning, semantic web and knowledge graphs: A review," *Vis. Inform.*, vol. 4, no. 1, pp. 32–42, 2020, doi: 10.1016/j.visinf.2020.01.001.

[13] T. Zhang, W. Zheng, Z. Cui, Y. Zong, and Y. Li, "Spatial–temporal recurrent neural network for emotion recognition," *IEEE Trans. Cybern.*, vol. 49, no. 2, pp. 839–847, Feb. 2019, doi: 10.1109/TCYB.2017.2788081.

[14] W. Hamat, T. Suzuki, K. Hashikura, and K. Yamada, "Anxiety expression using support vector machine," in *Proc. 2018 15th Int. Conf. Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, Chiang Rai, Thailand, 2018, pp. 421–424.

[15] A. Arsalan, M. Majid, and S. M. Anwar, "Electroencephalography based machine learning framework for anxiety classification," in *Proc. Int. Conf. Intelligent Technologies and Applications (INTAP)*, Bahawalpur, Pakistan, 2019, pp. 187–197, doi: 10.1007/978-981-15-5232-8_17.

[16] A. Javanbakht, L. Hinchey, K. Gorski, A. Ballard, L. Ritchie, and A. Amirsadri, "Unreal that feels real: Artificial intelligence-enhanced augmented reality for treating social and occupational dysfunction in post-traumatic stress disorder and anxiety disorders," *Eur. J. Psychotraumatol.*, vol. 15, no. 1, p. 2418248, 2024, doi: 10.1080/20008066.2024.2418248.

[17] A. Zimmer *et al.*, "Effectiveness of a smartphone-based, augmented reality exposure app to reduce fear of spiders in real-life: A randomized controlled trial," *J. Anxiety Disord.*, vol. 82, p. 102442, 2021, doi: 10.1016/j.janxdis.2021.102442.

[18] T. Jurcik *et al.*, "The efficacy of augmented reality exposure therapy in the treatment of spider phobia—A randomized controlled trial," *Front. Psychol.*, vol. 15, p. 1214125, 2024, doi: 10.3389/fpsyg.2024.1214125.

[19] M. Palau-Batet, J. Bretón-López, J. Grimaldos, C. Suso-Ribera, D. Castilla, A. García-Palacios, and S. Quero, "Improving exposure therapy through projection-based augmented reality for the treatment of cockroach phobia: A feasibility, multiple-baseline, single-case study," *Appl. Sci.*, vol. 14, no. 20, p. 9581, 2024, doi: 10.3390/app14209581.

[20] M. C. Juan, M. Alcañiz, J. Calatrava, I. Zaragoza, R. Baños, and C. Botella, "An optical see-through augmented reality system for the treatment of phobia to small animals," in *Proc. 2nd Int. Conf. Virtual Reality (ICVR)*, Lecture Notes in Computer Science, vol. 4563, Beijing, China, 2007, pp. 651–659, doi: 10.1007/978-3-540-73335-5_70.

[21] F. Fatharany, R. R. Hariadi, D. Herumurti, and A. Yuniarti, "Augmented reality application for cockroach phobia therapy using everyday objects as marker substitute," in *Proc. 2016 Int. Conf. Information and Communication Technology and Systems (ICTS)*, Surabaya, Indonesia, 2016, pp. 49–52, doi: 10.1109/ICTS.2016.7910271.

[22] M. Z. Uddin, M. M. Hassan, A. Alsanad, and C. Savaglio, "A body sensor data fusion and deep recurrent neural network-based behavior recognition approach for robust healthcare," *Inf. Fusion*, vol. 55, pp. 105–115, 2020, doi: 10.1016/j.inffus.2019.08.004.

[23] O. Balan, G. Moise, A. Moldoveanu, M. Leordeanu, and F. Moldoveanu, "An investigation of various machine and deep learning techniques applied in automatic fear level detection and acrophobia virtual therapy," *Sensors*, vol. 20, no. 2, p. 496, 2020, doi: 10.3390/s20020496.

[24] L. Petrescu, C. Petrescu, A. Oprea, O. Mitruţ, G. Moise, A. Moldoveanu, and F. Moldoveanu, "Machine learning methods for fear classification based on physiological features," *Sensors*, vol. 21, no. 13, p. 4519, 2021, doi: 10.3390/s21134519.

[25] O. Balan, G. Moise, A. Moldoveanu, M. Leordeanu, and F. Moldoveanu, "Fear level classification based on emotional dimensions and machine learning techniques," *Sensors*, vol. 19, no. 7, p. 1738, 2019, doi: 10.3390/s19071738.

[26] A. Mostafa, M. I. Khalil, and H. M. Abbas, "Emotion recognition by facial features using recurrent neural networks," in *Proc. 2018 13th Int. Conf. Computer Engineering and Systems (ICCES)*, Cairo, Egypt, 2018, pp. 417–422, doi: 10.1109/ICCES.2018.8639182.

[27] L. Gonzalez-Carabarin, E. A. Castellanos-Alvarado, P. Castro-Garcia, and M. A. Garcia-Ramirez, "Machine learning for personalised stress detection: Inter-individual variability of EEG-ECG markers for acute-stress response," *Comput. Methods Programs Biomed.*, vol. 209, p. 106314, 2021, doi: 10.1016/j.cmpb.2021.106314.

[28] A. Al-Ezzi, N. Yahya, N. S. Kamel, I. Faye, K. Alsaih, and E. Gunaseli, "Severity assessment of social anxiety disorder using deep learning models on brain effective connectivity," *IEEE Access*, vol. 9, pp. 86899–86913, 2021, doi: 10.1109/ACCESS.2021.3089358.

[29] Y. Yu, X. Si, C. Hu, and J. Zhang, "A review of recurrent neural networks: LSTM cells and network architectures," *Neural Comput.*, vol. 31, no. 7, pp. 1235–1270, 2019, doi: 10.1162/neco_a_01199.

[30] N. Passalis, A. Tefas, J. Kanniainen, M. Gabbouj, and A. Iosifidis, "Deep adaptive input normalization for time series forecasting," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 9, pp. 3760–3765, Sep. 2020, doi: 10.1109/TNNLS.2019.2944933.

[31] Y. Deng, C. Han, J. Guo, and L. Sun, "Temporal and spatial nearest neighbor values based missing data imputation in wireless sensor networks," *Sensors*, vol. 21, no. 5, p. 1782, 2021, doi: 10.3390/s21051782.

[32] X. Liu, Y. Zhang, and Q. Zhang, "Comparison of EEMD-ARIMA, EEMD-BP and EEMD-SVM algorithms for predicting the hourly urban water consumption," *J. Hydroinform.*, vol. 24, no. 3, pp. 535–558, 2022, doi: 10.2166/hydro.2022.146.

[33] E. M. Bartholomay and D. Houlihan, "Public speaking anxiety scale: Preliminary psychometric data and scale validation," *Pers. Individ. Dif.*, vol. 94, pp. 211–215, 2016, doi: 10.1016/j.paid.2016.01.019.

[34] T. O. Hodson, T. M. Over, and S. S. Foks, "Mean squared error, deconstructed," *J. Adv. Model. Earth Syst.*, vol. 13, no. 12, p. e2021MS002681, 2021, doi: 10.1029/2021MS002681.

[35] B. T. Atmaja and M. Akagi, "Evaluation of error- and correlation-based loss functions for multitask learning dimensional speech emotion recognition," *J. Phys.: Conf. Ser.*, vol. 1896, no. 1, p. 012004, 2021, doi: 10.1088/1742-6596/1896/1/012004.

[36] A. Abdallah, A. Celik, M. M. Mansour, and A. M. Eltawil, "Deep learning-based frequency-selective channel estimation for hybrid mmWave MIMO systems," *IEEE Trans. Wireless Commun.*, vol. 21, no. 6, pp. 3804–3821, Jun. 2022, doi: 10.1109/TWC.2021.3124202.

[37] T. O. Hodson, "Root-mean-square error (RMSE) or mean absolute error (MAE): When to use them or not," *Geosci. Model Dev.*, vol. 15, no. 14, pp. 5481–5487, 2022, doi: 10.5194/gmd-15-5481-2022.

[38] T. Siddique and M. S. Mahmud, "Ensemble deep learning models for prediction and uncertainty quantification of ground magnetic perturbation," *Front. Astron. Space Sci.*, vol. 9, p. 1031407, 2022, doi: 10.3389/fspas.2022.1031407.

[39] J. Qi, J. Du, S. M. Siniscalchi, X. Ma, and C.-H. Lee, "On mean absolute error for deep neural network based vector-to-vector regression," *IEEE Signal Process. Lett.*, vol. 27, pp. 1485–1489, 2020, doi: 10.1109/LSP.2020.3016837.

[40] J. L. Rodgers and W. A. Nicewander, "Thirteen ways to look at the correlation coefficient," *Amer. Statist.*, vol. 42, no. 1, pp. 59–66, 1988, doi: 10.1080/00031305.1988.10475524.

Fatemeh Abdollahzadeh,
Department of Computer Engineering, Ma.C.,
Islamic Azad University, Mashhad, Iran;
e-mail: fa.abdollahzadeh@iau.ac.ir.

Seyed Reza Kamel Tabbakh Farizani (Corresponding author),
Department of Computer Engineering, Ma.C.,
Islamic Azad University, Mashhad, Iran;
e-mail: drkamel@iau.ac.ir.

Yahya Forghani,
Department of Computer Engineering, Ma.C.,
Islamic Azad University, Mashhad, Iran;
e-mail: yahyafor2000@iau.ac.ir.

Masood Niazi Torshiz,
Department of Computer Engineering, Ma.C.,
Islamic Azad University, Mashhad, Iran;
e-mail: Masood.niazi@iau.ac.ir.

Hasan Abdollahzadeh,
Department of Psychology,
Payame Noor University, Tehran, Iran;
e-mail: abdollahzadeh@pnu.ac.ir.

Received 24.05.2025 , Revised 12.08.2025, Accepted 08.06.2026 Available online 30.06.2026