An Efficient Machine Learning Framework for Stress Prediction via Sensor Integrated Keyboard Data

P. B. PANKAJAVALLI1, G. S. KARTHICK1, AND R. SAKTHIVEL2

1Department of Computer Science, Bharathiar University, Coimbatore 641046, India
2Department of Applied Mathematics, Bharathiar University, Coimbatore 641046, India

Corresponding author: P. B. Pankajavalli (pankajavalli@buc.edu.in)

The work of P. B. Pankajavalli and G. S. Karthick were supported by the Department of Science and Technology—Interdisciplinary Cyber-Physical System (DST-ICPS), New Delhi, India, under Grant DST/ICPS/CPS-Individual/2018/193.

ABSTRACT Today’s sedentary life leads to a plethora of lifestyle-related illnesses. This has led to the quest to predict diseases before they occur. In the past, research on stress prediction was carried out conventionally in a laboratory-based environment. However, recent studies are focusing on developing non-invasive ways to predict stress with the help of wearable devices. Generally, the models developed for stress prediction do not provide accurate results because the stress patterns are highly subjective and vary from person to person. Therefore, person-dependent models may achieve higher accuracies. These models, however, have to be trained with collected data over a comparatively longer period of time. In this paper, an Adaptive Neuro-Fuzzy Inference System aided Fire Works Grey Wolf Optimization (ANFIS-FWGWO) classification algorithm has been proposed for stress prediction. In particular, the proposed machine learning framework has been implemented to predict computer users’ stress by using a sensor-integrated keyboard data. Various physiological parametric data were acquired during two different phases for the experimentation, and the received data was analyzed using an efficient machine learning framework. Specifically, the proposed framework encompasses various techniques such as data preprocessing for data smoothing and the Least Absolute Shrinkage and Selection Operator (LASSO) feature selection algorithm for identifying the important features on the data set. From the experimental analysis, it is concluded that the ANFIS-FWGWO classification algorithm discriminates the stress subjects with a high degree of accuracy when compared with existing classification algorithms.

INDEX TERMS Machine learning framework, stress prediction, fire works algorithm, grey wolf optimization, feature selection, classification.

I. INTRODUCTION Stress is considered as a psychological hyperarousal and physiological state in human beings. The World Health Organization (WHO) has found that stress is a massive epidemic of the twenty-first century. Generally, stress may be triggered by the discrepancy between the situational demands and the individual’s inability to leverage complicated situations. The state of difference challenges the complicated homeostasis or equilibrium of the human and maybe reinstated via a sequence of unique adaptive response systems organized by the Central Nervous System (CNS). This can be collectively stated as a stress response system. The incidence of stress can be due to diverse factors like work burden, grief, traumatic event, etc. Extreme stress minimizes the efficiency of work and leads to numerous illnesses and negative emotions. Continuous stress damages several internal organs, and this results in various disorders. These disorders lead to epithelial, gastrointestinal, musculoskeletal, cardiovascular, and mental diseases [1]. These diseases are usually aggravated by stress. The context of stress has necessitated the study of several diseases and their evolution in everyday life. There is an important shift in the paradigm due to the recognition of the mental state of an individual, which has
a significant impact on the well-being and functioning of every cell in the human body. Over a period, there was an elevation in interest among the research community in the early identification and diagnosis of stress-related disease [2]. The traditional stress identification approaches detect the stress and the stress-causing events that are highly complicated and influence the quality of human life. Hence, accurate analysis of the stress-related study is vital in minimizing the risk factors related to stress. The emergence of the Internet of Things (IoT) and wearable technologies allow researchers to develop smart systems for stress prediction [3].

The technology endorses the sensory equipment connected with the internet for obtaining the data. The data obtained from IoT applications are processed by Artificial Intelligence (AI) and learning-based approaches [4]. The advancement of sensory equipment and technology has provided the possibility of developing a better stress forecasting system. In particular, the biological signals acquired from the devices are not adequate, and also, the predicting algorithms are not efficient for accurate prediction [5]. The main aim of this research article is to detect stress with the assistance of physiological signals, which can be measured by the keyboard. The physiological signals such as force applied on keys, pulse rate, Galvanic Skin Response (GSR), and skin temperature of the computer user are sensed through the keyboard along with some key demographic parameters. The information is interpreted and classified by an effective feature selection and classification algorithm for stress prediction. Moreover, the significant features can be selected by Fast Correlation-Based Filter (FCBF) [6], Relief Feature Selection (RFS) [7], and Least Absolute Shrinkage and Selection Operator (LASSO) [8]. The efficient features are retrieved by the LASSO feature selection approach.

On the other hand, traditional machine learning algorithms suffer from parameter optimization and data overfitting issues. Recently, the ANFIS approach is found to be suitable for accurate prediction of stress due to its strong generalization and fast learning abilities. However, the ANFIS model updates the parameters using either the deterministic or probabilistic technique, which requires high computational power and fails to converge during local optimum. In this paper, motivated by the above discussion, we propose an Adaptive Neuro-Fuzzy Inference System aided Fire Works Grey Wolf Optimization (ANFIS-FWGWO) classification algorithm. The main contributions of this paper are summarized as follows:

1. An ANFIS structure is utilized to manage the problem complexity and to avoid erroneous stress prediction in the proposed algorithm.
2. Specifically, the ANFIS parameters can be efficiently optimized through hybridized optimization approach.
3. The FW, with its exploitation ability and GWO with its exploration ability, enhances the ANFIS model’s performance.
4. The proposed ANFIS-FWGWO with features selected by LASSO had attained the best accuracy in classification and forecasted stress effectively.

II. RELATED WORKS

A. OPTIMIZATION APPROACHES

The Adaptive Neuro-Fuzzy Inference System (ANFIS) is an Artificial Intelligence (AI) based approach that mimics the thinking of humans to resolve erroneous issues. It is based on the fuzzy inference model, and it can acquire the benefits of the fuzzy system and neural network. The inference system is related to the fuzzy rule set, and it has the learning ability to solve the non-linear functions [9]. Seydi Ghomsheh et al. [10] proposed a particle swarm optimization (PSO) based technique for optimizing the parameters of ANFIS. The training system concentrated on applying quantum behaving particle swarm optimization (QPSO) for ANFIS parameter setting. The consequent parameters are determined by the least square estimate (LSE) that is incorporated with the parameters altered by the QPSO technique. Hasanipanah et al. [11] established an effective ANFIS with an optimization technique for monitoring the drought, where the monitoring system utilized the genetic-based ANFIS, ant colony optimization-based ANFIS, and PSO-based ANFIS.

Grey wolf optimization (GWO) is a meta-heuristic technique that mimics the grey wolves’ behaviors. For effective ANFIS parameter learning, the GWO algorithm is adopted in [12]. Attallah et al. [13] developed a predictive model for analyzing the electroencephalogram (EEG) signals where the electrodes are placed on a different position on the head. The ANFIS parameter is included in the model, which is adjusted and optimized by incorporating GWO in the model. In [14], the authors modeled GWO based ANFIS technique to manage resources, namely soil, and water. This integrated approach was more effective and accurate than the ANN technique. El-Hasnony et al. [15] proposed a framework for the prediction of time, energy, and cost. The GWO is utilized for tuning the weights and parameters of ANFIS at the initial stage. Moreover, numerous optimization strategies are incorporated in training the ANFIS framework. In the research illustrated in [16], the PSO algorithm has been used for selecting the optimal features from the physiological dataset for predicting the stress level automatically.

A firework algorithm (FA) is established by observing the explosion of fireworks, and it is mainly used in optimization problems that have complicated functions. The process of searching has mimicked the explosion and the sparks diversity [17]. Li et al. [18] hybridized the firework and differential evolution algorithm for dynamic clustering and classification. The diversity and limited accuracy are due to the concurrent structure that is caused by the algorithms with rigid optimization. Xue et al. [19] established an evolutionary computation (EC) technique for the classification and clustering problem. The issues in the classification are directly
solved by the firework algorithm, and a linear equation is constructed based on the knowledge gained during the training phase, and the objective function is optimized by the evolutionary computation (EC). The foremost advantage in FA is the principle of self-adaptation, and it is utilized in building an optimized system with tuning parameters. Barraza et al. [20] hybridized GWO and FA, whereas it acquired the best features for solving the issues and attained better overall performance. The hybrid approaches the best exploration and exploitation capability in solving the issues.

**B. FEATURE SELECTION AND CLASSIFICATION APPROACHES**

Many of the learning techniques have faced issues in determining the optimized subset of features in order to improve the classification performance. In particular, the redundant or inappropriate features in the dataset can be removed by the feature selection approach. The classification performance is enriched by the selection of the best features. The fast correlation-based filter (FCBF) [21] has attained high accuracy and efficiency in the process of classification. FCBF removes the features with values that are close to zero linear correlation, in which the correlation threshold value 0.5 is used during the experimentation. The features will be selected only if the value is greater than the specified threshold. It assists in removing the redundant feature from the dataset. The relief feature selection (RFS) [22] algorithm is less complex, and thus it is simple to interpret. The feature extraction by multi labeled has grabbed attention and also shown significant progress in the selection of features. The least absolute shrinkage and selection operator (LASSO) [23] is a common high-dimensional data analysis technique that can perform feature selection and regularization simultaneously. LASSO can accomplish the selection of variables and regularization that can enhance the interpretation process and accuracy of prediction.

The linear regression (LR) model is a supervised learning technique in machine learning that mainly performs the regression [24]. It mainly focuses on forecasting the values based on the independent features. LR is utilized for finding the relationship between the target class and predictors. The regression framework is utilized in heart disease identification, and it plays a significant role in diversity. The outliers in the system have a massive impact on the boundaries and regression, which is considered a drawback of LR. The decision tree (DT) is projected to forecast the product for the recommendation system. The values acquired from the DT are simple and easy to prepare. The small alteration in the system can make a massive impact on the optimized decision tree [25]. Naïve Bayes (NB) [26] is a learning technique for multi-domain and large-scale information in the healthcare domain, which is utilized in cancer prediction. Naïve Bayes is independent and also faces the issue of zero frequency.

Random Forest (RF) [27] performs classification by generating trees and iteratively eliminates irrelevant features. The generation of a huge number of trees makes the algorithm ineffectual and too slow for a real-time forecasting system. Support vector machine (SVM) [28] effectively classifies the homogeneous data and forecast the series of observation. This classifier leverages the homogeneity in a similar class of data and exploits the divergence among diverse classes, and an effective classification is attained. The SVM classifier is ineffective when it is applied on a large scale of data and massive noise. Artificial Neural Network (ANN) [29] has been incorporated with fuzzy logic and genetic algorithm, which can be applied to various classification problems in order to achieve high accuracy.

**C. STRESS FORECASTING SYSTEM AND STRESS RELATED ISSUES**

Stress is stimulated by worries, pressure and has no control over the situation outcomes and consequences. The stress associated with work and mental health is considered a rising concern. The forecasting of stress is necessary at the preliminary stage, and it has an impact on regular life, whereas it causes cardiovascular disease, mental and musculoskeletal disorders [30]. The availability and emergence of digital technology allow the development of mobile phone-based applications and wearable devices, which in turn collect biological signals from humans for stress prediction. The advancement of prediction approaches, namely Machine Learning (ML) and Artificial Intelligence (AI), has played a significant role in the prediction of stress based on biological signals [31]. The wearable device helps predict and forecast the status of mental health and mood [32], [33]. In [34], various machine learning algorithms have been applied for analyzing the patient’s medical history to determine anxiety and suicidal tendencies. Shatte et al. [35] focused on applying machine learning algorithms to predict several mental health issues such as Alzheimer’s disease, schizophrenia, anxiety and depression. In the research study presented in [36], the linguistic features and keystroke have been analyzed for predicting the cognitive stress and physical stress state. Most of the existing algorithms have some limitations in collecting and analyzing biological signals. Therefore, an effective stress prediction model is needed for analyzing the biological signals effectively.

**III. SYSTEM METHODOLOGY**

The main intention of developing an efficient machine learning framework is to predict the stress of computer users, by processing the data generated by a sensor-integrated keyboard. The developed framework acquires the data from sensor integrated keyboard and then preprocesses the collected data with other supporting demographic features to standardize the dataset. The framework makes use of three feature selection algorithms to determine the prominent features and examines the three different set features obtained by feature selection algorithms using machine learning classifiers. The developed framework can be perceived from the workflow illustrated schematically in Fig. 1. Firstly, the data acquisition using sensor integrated keyboard is
elaborated. Secondly, the preprocessing techniques such as data smoothing and outlier detection are briefed. Finally, LASSO was used for feature selection and subsequently stress prediction is performed by the proposed ANFIS-FWGWO.

**A. INSTRUMENTATION AND DATA ACQUISITION**

The developed sensor integrated keyboard comprises two major units: (1) Sensor Integrated Keyboard and (2) Associated Apparatus. Initially, sensor integrated keyboard included twelve force sensors that were fixed on the keys. The force sensor keyboard measures the amount of force applied on keys by the system user. The associated apparatus consisted of a microcontroller, pulse sensor, GSR sensor, infrared temperature sensor, communication interface, and user interface. The primary parameters such as force applied on keys, pulse rate, GSR, and skin temperature of computer user will be acquired from the sensors and transmitted to the microcontroller and stored. In the research study [36], the mean absolute percent error (MAPE) was used to determine the error rate of the sensor measurements. From the analysis, the acceptable error rate of GSR and force sensor can be $\pm 10\%$ and $\pm 5\%$ for pulse sensor and skin temperature sensor. Further, various secondary parameters, including age, gender, systolic blood pressure, diastolic blood pressure, cortisol level, environmental factor, workload level, and so on, were considered and collected from the computer users via questionnaires. The acquired data were analyzed using an effective machine learning framework for predicting the user’s stress. Table 1 illustrates the description of the attributes in dataset. The primary and secondary data collected during the experimentation process was analyzed using statistical tools and Matlab. The primary data acquisition was carried out as two experimental phases:

(i) **Base Phase:** The data was acquired for 10 minutes in which users were allowed to perform non-constrained tasks during the base phase.

(ii) **Stress Phase:** The data was acquired for 15 minutes in which users were allowed to perform highly time-constrained and complex tasks.

Each phase of the acquired dataset contained set of pulse rates, skin temperature, GSR, and force applied on keyboard keys. These acquired primary data were combined with secondary data to examine the deviations in the parameters, which can be endorsed to determine the stress of computer users.

**B. DATA SMOOTHING AND OUTLIER DETECTION**

The outliers present in the dataset may impede the classification performance, and so the data points located far away

---

**Table 1.** Description of the features in dataset.

| S.No | Features     | Features Description                                      | Feature Type |
|------|--------------|-----------------------------------------------------------|--------------|
| 1    | KF1 – KF12   | Force applied on keyboard keys (from twelve keys)         | Numeric      |
| 2    | PR           | Pulse rate                                               | Numeric      |
| 3    | GSR          | Galvanic Skin Response                                   | Numeric      |
| 4    | ST           | Skin Temperature                                         | Numeric      |
| 5    | Age          | Age in years                                             | Numeric      |
| 6    | Gender       | Gender type                                              | Binary       |
| 7    | SBP          | Systolic Blood Pressure                                  | Numeric      |
| 8    | DBP          | Diastolic Blood Pressure                                 | Numeric      |
| 9    | CL           | Cortisol Level                                           | Nominal      |
| 10   | EF           | Environmental Factor                                     | Nominal      |
| 11   | WL           | Workload Level                                           | Nominal      |
| 12   | HT           | Hypertension                                             | Nominal      |
| 13   | FA           | Frequent Anxiety                                         | Nominal      |
| 14   | CD           | Concurrent Depression                                    | Nominal      |
| 15   | MT           | Muscle Tension                                           | Nominal      |
from the mean value of the corresponding random variable directly affect the performance of the data classification. To discard the isolated data points, it is necessary to measure the distance between the isolated data point and the mean value of the corresponding random variable. The threshold value is calculated by a multiplier of standard deviation. During experimentation, the outlier detection technique depicted in Algorithm 1 was applied on the base and stress phase data to obtain a significant conclusion by removing the isolated data points.

Algorithm 1 Pseudocode for Outlier Detection

Input: D: Dataset,
Output: R: Outliers Removed Dataset
Begin
D ← Impute the Dataset
M: Number of Column
N: Number of Samples
μ : Mean Value of the Column Vector
σ : Standard Deviation of the Column Vector
t: Threshold
temp: Temporary Variable

for i = 1, 2, ..., M and j = 1, 2, ..., N;
Initialize N ← 1000
Determine the μ and σ for every column vector
Evaluate the condition for each column vector V
if (||Vij − µ|| ≤ 4 × σ)
temp ← Vij + 1
end if
if (temp > t)
R ← temp
else
Remove the data point as an outlier
end if
end for

The acquired time-series data collected in two variants, as base phase and stress phase, may suffer from noise and hardware interference during data generation. Thus, it is necessary to remove the difference in the data by applying suitable smoothing techniques. The single moving average technique is applied to the stress dataset to avoid the small variations and to exhibit meaningful information from the data. The moving average technique estimates the running average over the data points of window size ‘N’ and this process is continued subsequently on all data points of a feature. Moreover, the smaller window size cannot remove the noise from the data, and a larger window size leads to the loss of vital information. Based on the trial-and-error process, the window size of N = 60 was used in this experiment.

The single moving average can be represented by mathematical expression as

\[
\text{Single Moving Average} = \frac{x_P + x_{P-1} + \ldots + x_{P-N+1}}{N} \quad (1)
\]

where \(x_P + x_{P-1} + \ldots + x_{P-N+1}\) denotes the moving data points, and \(N\) is the total size of the data points. The data smoothing technique is applied to acquired data during the base phase and stress phase. The moving average technique removes the susceptible noise and interference from the acquired data and explores all the data points to lie on the smooth curve without dispersion. The comparison between the raw data and the preprocessed data of single keying force, pulse rate, GSR, and skin temperature is demonstrated with 60 sample data points in Fig. 2, in which the black dotted line indicates the raw data and the red dotted line indicates the preprocessed data.

C. FEATURE SELECTION

Feature selection algorithms identify and select the prominent features from the entire feature space. They can be considered as an imperative process that reduces the degradation of classifiers’ performance due to inappropriate features in the dataset and also improves the performance of the classifier in terms of a high degree of accuracy and reduces the processing time. In this work, three feature selection algorithms are used and elaborated below: The Fast Correlation-Based Filter (FCBF) is a feature selection technique that uses the sequential searching strategy [21]. As the nominal features cannot be applied directly to the feature selection and classification process, they are transformed into numerical features by using the label encoding operation before the feature selection and classification process. Initially, the FCFB technique includes the entire feature space and then determines the symmetric uncertainty to identify the dependencies among the set of features and evaluates how the target output is influenced by the features. Then, the most relevant features are selected based on the backward sequential searching strategy. It is necessary to standardize the data because a higher coefficient of the features will increase the cost function values. Therefore, the Standard Scaler standardization is applied on the data and then the LASSO feature selection algorithm is applied. Additionally, it also optimizes the values of the cost function. The Least Absolute Shrinkage and Selection Operator (LASSO) identify the relevant features by updating the coefficient of features, where the features are removed from the feature space only when the feature coefficient becomes zero [23]. Therefore, the features with greater coefficient values are added to the relevant features, and the remaining will be removed. Relief feature selection (RFS) uses the theory of attribute-based learning, which assigns weights to each feature depending upon the significant ratio, and that weight discriminates the features among the target classes [22]. In the relief feature selection algorithm, the features are ranked based on their weights, and the features with exceeding user-defined weights are considered as significant features which have a strong influence on target classes. Initially, the nearest hit (nearest attribute of identical class) and nearest miss (nearest attribute of contradictory class) can be determined for every feature, and the weights are updated.
based on the nearest miss and hit differentiation ratio. The feature weight will be high only when the feature is able to differentiate it among the instances of different classes and attain a similar weight for the same class instances. It was noted that the feature set selected by LASSO provided better performance when compared to the other two feature selection algorithms [23]. Hence in this work, features selected by LASSO are considered for further experimental analysis.

D. ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM-AIDED FIREWORKS GREY WOLF OPTIMIZATION (ANFIS-FWGWO) CLASSIFICATION ALGORITHM

Grey wolf optimization (GWO) holds the exploitation capability, but it is weak in neglecting the local optimum and premature convergence. The Fire Works (FW) algorithm holds the exploration ability, but it does not have exploitation capability. In this paper, the firework algorithm with exploration ability and the GWO algorithm with exploitation ability are combined to attain the global optimal solution. The parameters of ANFIS, GWO, and FW are initialized, and the strategy of selection is utilized to update the search agents. In the strategy of selection, if there are numerous other location information around $\mathbf{A}_p$, the probability of selection will be minimized to keep the divergence of the subsequent generation.

The whole searching process is led by the alpha, and the three best locations are acquired as alpha, beta, and delta. The positions are updated as follows,

$$\mathbf{O}_\alpha = \left| \mathbf{N}_1 \mathbf{A}_\alpha - \mathbf{A} \right|, \quad \mathbf{O}_\beta = \left| \mathbf{N}_2 \mathbf{A}_\beta - \mathbf{A} \right|,$$

$$\mathbf{O}_\delta = \left| \mathbf{N}_3 \mathbf{A}_\delta - \mathbf{A} \right|,$$

$$\mathbf{A}_1 = \mathbf{A}_\alpha - \mathbf{X}_1, \quad \mathbf{A}_2 = \mathbf{A}_\beta - \mathbf{X}_2, \quad \mathbf{A}_3 = \mathbf{A}_\delta - \mathbf{X}_3,$$

$$\mathbf{A}_{(t+1)} = \frac{\mathbf{A}_1 + \mathbf{A}_2 + \mathbf{A}_3}{3}.$$  \hspace{1cm} (2)

where the positions of the alpha, beta, and delta are signified as $\mathbf{A}_\alpha, \mathbf{A}_\beta, \mathbf{A}_\delta$; the distance among the search agents are signified as $\mathbf{O}_\alpha, \mathbf{O}_\beta, \mathbf{O}_\delta$; coefficient vectors are signified as $\mathbf{N}_1, \mathbf{N}_2, \mathbf{N}_3$ and the direction towards the length of the step is signified as $\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3$. The search agents after and before updating are signified as $\mathbf{A}_{(t+1)}$ and $\mathbf{A}$, respectively.

When the value of $A_\alpha$ is altered, the values of the coefficient are updated. The exploitation and exploration abilities are balanced by the equilibrium coefficient. The position of the wolf is updated by $A_\alpha$, and the best fitness value has been altered. When the value of the current position is nearer to the optimal solution, the value of coefficient $c$ is updated by changing the strategies of search; the process is given in the following equation;

$$c = 0.9 \times \left( 1 - \cos \left( \frac{\pi}{2} \frac{t}{\text{Maxim}_{\text{itera}}} \right) \right)$$  \hspace{1cm} (4)

where $c$ is the adaptive coefficient value, the count of the iteration is signified by $t$, and the maximum count of the iteration is signified by $\text{Maxim}_{\text{itera}}$. 

FIGURE 2. Comparison between raw data vs. pre-processed data.
The subsequent generation of firework will be selected from the spark locations. The best value is retained for the next iteration. The probability selection and distance of \( a_p \) is given as

\[
R(a_p) = \sum_{q \in k} d(a_p, a_q) = \sum_{q \in k} \| a_p - a_q \| \quad (5)
\]

\[
c(a_p) = \frac{R(a_p)}{\sum_{q \in k} R(a_{qp})}. \quad (6)
\]

The generated populations are passed into the ANFIS phase. ANFIS utilizes fuzzy logic to transform inputs of extremely correlated neural network elements and also to transform the connection information as the desired output. The ANFIS is composed of five layers which are fuzzification, normalization, product, summation, and defuzzification. The framework of ANFIS has adaptable and fixed nodes. The proficiency of the network primarily relies on adaptable parameters, and the learning guidelines alter the settings of the specific parameter, which in turn minimizes the incidence of error among the actual output and desired output.

To elucidate the framework of ANFIS, two fuzzy inference-based rules are considered:

**Rule 1:** If \( a \) is \( L_1 \) and \( b \) is \( M_1 \), then \( f_1 \)

\[
= x_1 a + y_1 b + r_1
\]

**Rule 2:** If \( a \) is \( L_2 \) and \( b \) is \( M_2 \), then \( f_2 \)

\[
= x_2 a + y_2 b + r_2
\]

where the inputs are signified as \( a \) and \( b \) for the fuzzy sets \( L_1, L_2, M_1, M_2 \), and defuzzification layer parameters are \( x_1, x_2, y_1, y_2, \) and the output value is signified as \( f_1 \) or \( f_2 \). In the IF case, the parameters are denoted as a premise or precedent parameter, and in the THEN case, the parameters are determined as a significant parameter. Layer 1 holds the principle parameter. Layer 4 holds the resultant portion of nodes which can be adaptable in the Layer 2 and 3 nodes with the product and normalization, respectively. The adaptive member function of the fuzzification layer is node \( p \), and for the fuzzy set or linguistic label, any parameterized function act as a membership function. The Gaussian membership function is decided by using two parameters \((n, \sigma)\), and the output information is given as;

\[
b_p^{(1)} = \text{Gaussian} (a : n, \sigma) = e^{-\frac{1}{2}(\frac{a - n}{\sigma})^2} \quad (7)
\]

While the parameters in the Gaussian membership function are standardized by width \( \sigma \) and center \( n \), the Gaussian parameters are pointed as antecedent or premise parameters.

Every node in this layer responds to a fuzzy rule with a single Sugeno style. The input information is collected in this layer from the relevant neurons in the fuzzification and defines the strength of the fuzzy rule they denote. As an outcome, from the consequence layer, the output information is gathered as,

\[
b_p^{(2)} = \prod_{p=1}^{k} a_{qp}^{(2)} \quad (8)
\]

where the input information from fuzzification layer (q) to product layer (p) is denoted as \( a_{qp}^{(2)} \), and the output information is denoted as \( a_p^{(2)} \) for every neuron in the product layer.

The feedback information from all the previous layer neurons is gathered by the nodes in the normalization layer, and the weighted firing power of the rule is measured in this layer. In the normalization layer, information is determined as,

\[
b_p^{(3)} = \frac{a_p^{(2)}}{\sum_{q=1}^{g} a_{qp}^{(5)}} = \bar{f}_p, \quad (9)
\]

where \( a_{qp}^{(5)} \) signifies the input generated and received based on the neuron \( q \) from product to normalization layer \( p \). The normalization layer output information is denoted by \( b_p^{(3)} \).

The node in this layer is adaptable and modifiable and also measures the relevant weights. The specific rule for measuring the weight is given as

\[
b_p^{(4)} = a_{qp}^{(4)} [k_{q0} + k_{q1} + k_{q2}] = \tilde{\mu}_p[k_{q0} + k_{q1} + k_{q2}], \quad (10)
\]

where the input of this layer is denoted by \( a_{qp}^{(4)} \) and \( k_{q0}, k_{q1}, k_{q2} \) are considered as a consequential parameter of node \( p \). The defuzzification layer output information is denoted by \( b_p^{(6)} \).

The overall output of the ANFIS model is acquired by the summation layer that is the aggregated information of the entire previous layer’s output information.

\[
b_p^{(5)} \leftarrow \sum_{p=1}^{g} a_p^{(5)} = \sum_{p=1}^{g} \bar{f}_p[k_{q0} + k_{q1} + k_{q2}] \quad (11)
\]

where the input of this layer is denoted by \( a_{qp}^{(4)} \) and the summation layer output information is denoted by \( b_p^{(5)} \).

In FWGWO, the location of every particle reproduces a full set of parameters for the ANFIS system. Once the process is completed, and the maximum iteration is reached, the process is terminated. The performance of the classification is investigated with the assistance of classification parameters. The working procedure of ANFIS-FWGWO is described in Algorithm 2.

**IV. EXPERIMENTAL RESULTS AND DISCUSSIONS**

The experiment has been carried out to evaluate the performance and efficiency of the proposed ANFIS-FWGWO classification algorithm based on the features selected by LASSO. The primary sensor data is analyzed in Section A. The result of LASSO feature selection algorithms is discussed in Section B. The k-fold validation procedure and performance evaluation metrics are taken into account for analyzing the classification algorithm, which is elaborated in Section C, and the performance of various classification algorithms are reported in Section D.

**A. ANALYSIS OF PRIMARY SENSOR DATA**

A significant difference has been recognized between the data acquired during the base phase and stress phase. The skin conductance level is increased during the stress phase, where
Algorithm 2: Pseudocode for ANFIS-FWGWO

Input: Size of the Population, Number of Iterations, Learning Factors
Output: Optimized Stress Prediction Model
Initialization
Calculate the objective function of every search agent in the iteration
$A_\alpha$ = The first best search agent
$A_\beta$ = The second best search agent
$A_\delta$ = The third best search agent
while (t < maximum iteration)
for $p = 1:N$ // ANFIS training
Generate $b^{(1)}_p \leftarrow \text{Gaussian} (a : n, \sigma) = e^{-\frac{1}{2}(\frac{a-n}{\sigma})^2}$
Calculate $b^{(2)}_p \leftarrow \prod_{k=1}^{p} a^{(2)}_q$
Calculate $b^{(3)}_p \leftarrow \frac{\mu^{(3)}_p}{\sum_{q=1}^{g} \mu^{(3)}_q} = \hat{\mu}_p$
Calculate $b^{(4)}_p \leftarrow a^{(4)}_p [k_{p0} + k_{p1} + k_{p2}] = \mu_p [k_{p0} + k_{p1} + k_{p2}]$
Calculate $b^{(5)}_p \leftarrow \sum_{p=1}^{g} a^{(5)}_p = \sum_{p=1}^{g} \mu_p [k_{p0} + k_{p1} + k_{p2}]$
Input the set of parameters to FWGWO for each search
Update the current position
Update the values
Calculate the objective function
Update the search agent values
if the first solution is changed, then
Update the adaptive coefficient value
end if
if k \geq T and rand() > c then
   For every search $a_p$ do
      Explosion spark generation
   End for
Selection of new search agent
$t = t+1$
k = 0
end if
$k = k+1, t = t+1$
end while
Return $A_\alpha$
End

The significant variation in GSR data was identified from the base phase to the stress phase. The GSR value ranged between $916 \pm 472$ during the stress phase and $686 \pm 270$ during the base phase. The pulse rate and skin temperature are absolutely unbalanced from the base phase to the stress phase, which indicates the elevation in pulse rate and skin temperature during the stress phase. In the stress phase, the pulse rate is increased, and the time is reduced between a pulse and the subsequent pulse. The pulse rate ranged between $100 \pm 9$ during the stress phase and $75 \pm 10$ during the base phase. Similarly, the skin temperature ranged between $86 \pm 12$ during the stress phase. Meanwhile, the keying force value ranged between $650 \pm 350$ during the stress phase and $200 \pm 75$ during the base phase. Hence, significant variations between the keying forces were observed while comparing the data acquired during both phases.

B. ANALYSIS OF LASSO FEATURES SELECTION ALGORITHM

LASSO feature selection algorithm identifies the best features by calculating and updating the coefficient of features. LASSO considers a feature with a greater coefficient important. This plays a predominant role in predicting the target class accurately. Table 2 depicts the list of features ($n = 18$) selected by LASSO based on the scores, and Fig. 3 represents the ranking of the selected features. From the results, it is shown that the most prominent features for predicting stress are keying forces applied on keyboard keys, skin temperature, GSR, and pulse rate.

C. K-FOLD CROSS-VALIDATION AND PERFORMANCE EVALUATION METRICS

In this experimentation, the k-fold cross-validation technique is used for result validation where the dataset is split into k identical parts, and k-1 parts of the dataset are used for classifier training, and the remaining dataset is used for classifier testing. Here, 10-fold cross-validation is used in which 90% of the dataset is used for training, and 10% of the dataset is used for testing. Finally, results are determined by taking the mean of the results obtained on each iteration.

In this work, the performance of the classifiers is evaluated using various metrics, which include Accuracy, Sensitivity, Specificity, F1-Score, AUC-Score, and Mathew Correlation.
TABLE 2. Features selected and their scores by LASSO algorithm.

| S.No | Features            | Score | S.No | Features            | Score |
|------|---------------------|-------|------|---------------------|-------|
| 1    | Keying Forces 1     | 0.22  | 10   | Keying Forces 10    | 0.236 |
| 2    | Keying Forces 2     | 0.209 | 11   | Keying Forces 11    | 0.183 |
| 3    | Keying Forces 3     | 0.207 | 12   | Keying Forces 12    | 0.177 |
| 4    | Keying Forces 4     | 0.195 | 13   | Pulse Rate          | 0.107 |
| 5    | Keying Forces 5     | 0.87  | 14   | Skin Temperature    | 0.104 |
| 6    | Keying Forces 6     | 0.308 | 15   | GSR                 | 0.091 |
| 7    | Keying Forces 7     | 0.307 | 16   | Blood Pressure      | 0.084 |
| 8    | Keying Forces 8     | 0.291 | 17   | Serum Cortisol      | 0.081 |
| 9    | Keying Forces 9     | 0.247 | 18   | Workload Level      | 0.079 |

FIGURE 4. Confusion matrix.

Coefficient (MCC) [37]. The performance metrics are evaluated using the confusion matrix depicted in Fig.4.

Based on the confusion matrix, True Negative (TN) represents that the subject is not under stress and the classifier also correctly predicts that the subject is not under stress. True Positive (TP) denotes that the subject with stress, and the classifier also correctly predicts the subject is under stress. False Positive (FP) is also referred to as Type-1 error, specifies that the subject is not under stress, but the classifier incorrectly predicts that the subject is under stress. False Negative (FN) is also referred to as Type-2 error, specifies that the subject is under stress, but the classifier incorrectly predicts that the subject is not under stress [37].

1) ACCURACY
Accuracy is the ratio of correct stress predictions to the overall input samples, which can be interpreted as:

\[
Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \tag{12}
\]

2) SPECIFICITY
The specificity of the classification algorithm notifies the ratio of the classified non-stress subjects to the total number of non-stress subjects. In other words, a subject can be predicted as a stressed subject wherein the subject actually belonged to the non-stress category. Specificity can be calculated as:

\[
Specificity = \frac{TN}{TN + FP} \times 100 \tag{13}
\]

3) SENSITIVITY
The sensitivity of the classification algorithm shows the ratio of the classified stress subjects to the total number of stress subjects. In other words, a subject can be predicted as a stress subject where the subject actually belonged to the stress category. Sensitivity can be calculated as:

\[
Sensitivity = \frac{TP}{TP + FN} \times 100 \tag{14}
\]

4) F1-SCORE
The F1-Score of the classification algorithm is the weighted measure whose value ranges between 0 and 1, where the value 1 denotes the better performance of the classification algorithm and the value 0 denotes the poor performance of the classification algorithm.

\[
F1 - Score = \frac{2 \times (Precision \times Recall)}{Precision + Recall} \times 100 \tag{15}
\]

where recall is a measure that determines the capability of the model in identifying the true positives accurately, and precision is the ratio of accurately identified positives among total positives.

5) MCC
MCC of the classification algorithm is the determination of the correlation coefficient among the actual and predicted outcomes. The outcome of MCC is between \(-1\) and \(+1\), where \(-1\) denotes the wrong stress prediction, \(+1\) denotes the correct stress prediction, and 0 represents the random stress prediction. MCC will be calculated as:

\[
MCC = \frac{TP \times TN - FP + FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{16}
\]

Finally, the stress prediction ability of the classification algorithms will be analyzed using the Area Under Curve (AUC), which can be estimated by plotting the difference between the true positive rate and the false-positive rate.

D. PERFORMANCE ANALYSIS OF MACHINE LEARNING CLASSIFIERS WITH FEATURE SELECTED BY LASSO
The performance of machine learning classification algorithms with the features selected by LASSO is depicted in Table 3.

From the experimental result analysis, the LR classification algorithm achieves \{90%, 89%, 90%, 90%, 82%, 90%\} for \{Accuracy, Sensitivity, Specificity, F1-Score, MCC, AUC\} with respect to the tuning parameter \(C = 1\) and the variations in the performance under different tuning parameters is presented in Fig 5a. The k-NN classifier achieves the performance as \{83%, 88%, 87%, 81%, 66%\}.
83%} for {Accuracy, Sensitivity, Specificity, F1-Score, MCC, AUC} with respect to tuning parameter \( k = 1 \) and the negligible difference in the performance under various hyperparameters can be noted from Fig. 5b. The performance of NB classification algorithm is \{87%, 88%, 86%, 86%, 75%, 87%\} for {Accuracy, Sensitivity, Specificity, F1-Score, MCC, AUC} and it can be observed from Fig. 5c. The SVM (Kernel = RBF) classifier achieves its performance as \{85%, 88%, 83%, 85%, 72%, 86%\} for {Accuracy, Sensitivity, Specificity, F1-Score, MCC, AUC} with respect to tuning.
FIGURE 6. Performance of corresponding classifiers on features selected by LASSO with respect to (a) Accuracy, (b) Sensitivity, (c) Specificity, (d) F1-Score, (e) MCC and (f) AUC under certain tuning hyperparameter yields best results for LR, k-NN, NB, SVM-RBF, SVM-Linear, RF, ANN and ANFIS-FWGWO.

parameters $C = 100$, $g = 0.0001$ and the difference in the performance can be observed from Fig. 5d under tuning of hyperparameters.

The SVM (Kernel = Linear) classifier achieves its performance as {82%, 79%, 89%, 79%, 66%, 81%} for {Accuracy, Sensitivity, Specificity, F1-Score, MCC, AUC} with respect to tuning parameters $C = 100$, $g = 0.0001$ and the difference in the performance can be observed from Fig. 5e under tuning of hyperparameters. The RF classification algorithm produces the results as {79%, 70%, 86%, 75%, 59%, 78%} for {Accuracy, Sensitivity, Specificity, F1-Score, MCC, AUC} with respect to tuning parameter of ntree = 50 and the difference in the performance can be observed from Fig. 5f under tuning of hyperparameters. The performance of ANN classifier with 40 hidden neurons is {84%, 88%, 77%, 82%, 70%, 83%} for {Accuracy, Sensitivity, Specificity, F1-Score, MCC, AUC}. Further, the ANN classifier with 16 and 20 hidden neurons and the high-pitched variations in the performance under various hyperparameters can be observed from Fig. 5g.

In the research experiment, the proposed ANFIS-FWGWO classification algorithm has been employed for stress prediction. The LASSO feature selection algorithm extracted the most prominent features, which help for the prediction of stress. The proposed ANFIS-FWGWO algorithm improves the learning rate and reduces the computation complexity. In particular, the hybridization of FWGWO algorithm overcomes the exploitation and exploration issues and helps in parameter optimization.

The performance of the ANFIS-FWGWO classification algorithm produces a result as {94%, 90%, 97%, 93%, 87%, 94%} for {Accuracy, Sensitivity, Specificity, F1-Score, MCC, AUC} concerning 100 iterations and the difference in the performance can be observed from Fig. 5h under tuning of hyperparameters. From Fig. 5h, it can be seen that the performance of the proposed ANFIS-FWGWO
Classification algorithm is high when the number of iterations is reduced. Thus, it can be concluded that the performance of ANFIS-FWGWO algorithm with LASSO features was outstanding when compared with other classification algorithms, which can be seen from Fig. 6.

V. CONCLUSION

In this paper, an efficient machine learning framework for predicting stress in computer users was developed. We have proposed ANFIS-FWGWO classification algorithm in which demographic features and sensor integrated keyboard data are used for the evaluation of stress prediction. The proposed algorithm incorporates the exploration ability of GWO and the exploitation ability of FW. The proposed classification algorithm is compared with existing machine learning classification algorithms and evaluated using different metrics to showcase the significance of the proposed classification algorithm. From the obtained results, it can be concluded that the ANFIS-FWGWO classification algorithm outperforms the existing algorithms with an accuracy of 94% with respect to the features selected by LASSO algorithm. In the future, the proposed ANFIS-FWGWO classification algorithm can be evaluated against multi-class classification problems, and the performance can be analyzed by integrating it with cloud platforms.

REFERENCES

[1] G. P. Chrousos, “Stress and disorders of the stress system,” Nature Rev. Endocrinol., vol. 5, no. 7, pp. 374–381, Jun. 2009.

[2] G. Sandra Hart and E. LowellStaveland, “Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research,” Adv. Psychol., vol. 52, pp. 139–183, Apr. 1998.

[3] P. B. Pankajavalli and G. S. Karthick, “A unified framework for stress forecasting using machine learning algorithms,” Proc. Adv. Comput. Intell. Inform., Int. Conf. Adv. Comput. Intell. Inform. (ICACII), R.Chillarige, S.Distefano, S.Rawat, Eds. Singapore: Springer, Dec. 2019, pp. 199–210.

[4] K. Venkataareshbabu, N. Nisheel, R. Sathivel, and K. Muralitharan, “Novel elegant fuzzy genetic algorithms in classification problems,” Soft Comput., vol. 23, no. 14, pp. 5583–5603, Jul. 2019.

[5] P. Verma and S. K. Sood, “A comprehensive framework for student stress monitoring in fog-cloud IoT environment: M-health perspective,” Med. Biol. Eng. Comput., vol. 57, no. 1, pp. 231–244, Jan. 2019.

[6] B. Senliol, G. Gulgezen, L. Yu, and Z. Catelepe, “Fast correlation based filter (FCBF) with a different search strategy,” in Proc. 23rd Int. Symp. Comput. Inf. Sci., Istanbul, Turkey, Oct. 2008, pp. 1–4.

[7] Q. Lou, Z. Deng, K.-S. Choi, H. Shen, J. Wang, and S. Wang, “Robust multi-label relief feature selection based on fuzzy margin co-optimization,” IEEE Trans. Emerg. Topics Comput. Intell., early access, Jan. 8, 2021, doi: 10.1109/TETCI.2020.3044679.

[8] Z. Wang, X. Xiao, and S. Rajasekar, “Novel and efficient randomized algorithms for feature selection,” Big Data Mining Anal., vol. 3, no. 3, pp. 208–224, Sep. 2020.

[9] S. Guillaume, “Designing fuzzy inference systems from data: An interpretability-oriented review,” IEEE Trans. Fuzzy Syst., vol. 9, no. 3, pp. 426–443, Jun. 2001.

[10] V. Seydi Gholmeh, M. Aliyari Shoorehdeli, and M. Teshnehab, “Training ANFIS structure with modified PSO algorithm,” in Proc. Mddit. Conf. Control Autom., Athens, Greece, Jun. 2007, pp. 1–6.

[11] M. Hasanipanah, H. B. Amnieh, and H. Arab, “Feasibility of PSO–ANFIS model to estimate rock fragmentation produced by mine blasting,” Neural Comput. Appl., vol. 30, pp. 1015–1024, Aug. 2018.

[12] E. Emray, H. M. Zawbaa, and C. Grosan, “ Experienced gray wolf optimization through reinforcement learning and neural networks,” IEEE Trans. Neural Netw. Learn. Syst., vol. 29, no. 3, pp. 681–694, Mar. 2018.

[13] O. Attallah, “An effective mental stress state detection and evaluation system using minimum number of frontal brain electrodes,” Diagnostics, vol. 10, no. 5, p. 292, May 2020.

[14] H. Kour, J. Manhas, and V. Sharma, “Evaluation of adaptive neuro-fuzzy inference system with artificial neural network and fuzzy logic in diagnosis of Alzheimer disease,” in Proc. 6th Int. Conf. Comput. Sustain. Global Develop. (INDIACom), New Delhi, India, Mar. 2019, pp. 1041–1046.
G. S. KARATHICK is currently a Senior Research Fellow under DST-ICPS Project and a full-time Ph.D. Research Scholar with the Department of Computer Science, Bharathiar University, Coimbatore, India. Previously, he worked as an Assistant Professor with the Department of Computer Science, Sri Ramakrishna College of Arts and Science, Coimbatore. He has qualified UGC-NET (computer science and applications), in June 2020. He has published research articles in journals, book chapters, and conferences. He has edited one book and filed an Indian patent. His areas of interest include machine learning algorithms.

R. SAKTHIYEL received the B.Sc., M.Sc., M.Phil., and Ph.D. degrees in mathematics from Bharathiar University, Coimbatore, India, in 1992, 1994, 1996, and 1999, respectively. He served as a Lecturer with the Department of Mathematics, Sri Krishna College of Engineering and Technology, Coimbatore, from 1999 to 2002. He joined as a Research Professor with the Department of Computer Science and Engineering, Pohang University of Science and Technology, Pohang, South Korea, from 2006 to 2008. From 2008 to 2013, he was a Professor with the Department of Mathematics, Sri Ramakrishna Institute of Technology, Coimbatore, India. He has published over 450 research articles in reputed Science Citation Index journals. He has published a book and a good number of book chapters in Springer. His current research interests include systems and control theory, optimization techniques, and nonlinear dynamics. He received the most coveted “Highly Cited Researcher” Award for the last four years, in 2017, 2018, 2019, 2020, consecutively from Clarivate Analytics, USA. He has been on the Editorial Board of international journals, including IEEE Access, Journal of the Franklin Institute, Neurocomputing, Advances in Difference Equations, Neural Processing Letters, and Journal of Electrical Engineering & Technology.

P. B. PANKAJAVALLI is currently an Assistant Professor with the Department of Computer Science, School of Computer Science and Engineering, Bharathiar University, Coimbatore, Tamil Nadu, India. She has around 16 years of experience in teaching. She is also carrying out funded projects in the domain sensor networks and the Internet of Things. She has published around 20 articles in journals and conference proceedings at the national and international level, four book chapters, oneedited book, and filed an Indian patent. Her areas of interests include ad-hoc networks, wireless sensor networks, and the Internet of Things.