International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 609-626, October-2023

https://doi.org/10.47001/IRJIET/2023.710081

Stress Monitoring and Relieving Application for IT Professionals

[1]M. S. D. Perera, [2]S. M. D. A. R Jayathilake, [3]J. D. Ranasinghe, [4]S. V. Bartholomeusz, [5]H. M. Samadhi Chathuranga [6]Samitha Vidhanaarachchi, [7]Thilanga Jayarathne, [8]Arosha Dasanayaka

[1,2,3,4] Undergraduate, Department of Computer Science and Software Engineering, Faculty of Computing, Sri Lanka Institute of Information and Technology, Colombo, Sri Lanka

[5]Lecturer, Department of Information Technology, Faculty of Computing, Sri Lanka Institute of Information and Technology, Colombo, Sri Lanka

[6]Lecturer, Department of Computer Science and Software Engineering, Faculty of Computing, Sri Lanka Institute of Information and Technology, Colombo, Sri Lanka

[7]CEO at Xinotech Technology Services Inc., Colombo, Sri Lanka

[8]Counselling Psychologist at Mind Heals, Kurunegala, Sri Lanka

Abstract - **The proposed solution presented in this report aims to address the critical issue of stress detection and management among IT professionals. Our innovative approach leverages machine learning and consists of four key components, three of which actively monitor and analyze an individual's stress levels through their keystroke dynamics, heart rate variability (HRV) via an external mouse (utilizing cost-effective IOT devices), and facial expressions captured by a webcam. The fourth component focuses on providing tailored recommendations and suggestions to help users reduce their detected stress levels. Stress is a prevalent concern among IT professionals, with potential long-term repercussions on both physical and mental health. Recognizing the urgency of addressing this problem, our system facilitates early stress detection and offers practical strategies to mitigate and maintain stress at manageable levels. The ultimate goal is to enhance the overall work experience, minimize health complications, and boost productivity among IT professionals who utilize our user-friendly approach, which integrates seamlessly with their everyday tools and equipment. This holistic solution holds the promise of a healthier, happier, and more productive workforce in the IT industry. Furthermore, our system is designed to be scalable and adaptable to various IT environments, allowing organizations to tailor it to their specific needs and preferences. It can be seamlessly integrated into existing IT infrastructure, making it a cost-effective and efficient solution for companies seeking to prioritize the well-being of their IT professionals.**

Keywords: Aggregation, Classification model, IT Professionals, Keystroke Dynamics, Machine learning, Predictions, Program, Script, Stress Detection

I. INTRODUCTION

Stress is a pervasive and critical health issue today, with far-reaching consequences, including burnout, heart disease, and even premature death. Researchers and experts have explored various avenues to detect and address stress, often relying on innovative technologies and approaches. A survey [1] conducted by Paniker et. Al. discusses several machine learning techniques to detect stress. This literature review synthesizes information from four distinct components related to stress and emotion detection and management, aiming to provide a comprehensive understanding of the landscape.

In today's dynamic and demanding world, stress has become an omnipresent challenge affecting individuals' well-being, quality of life, and overall productivity. The shift towards remote work, particularly in the IT industry, has introduced new stressors, such as isolation, potentially leading to heightened stress levels and related health issues [2].

Stress, as a complex phenomenon, can lead to a wide range of adverse consequences, from mental health issues to severe cases of depression and even suicide. Traditional stress detection relies on symptom-based assessments that are often subjective and require expert intervention. However, recent advancements in technology, particularly in machine learning, offer promising avenues for stress detection and management. A review paper [3] written by Gedam et. Al. shows how stress can be detected successfully and accurately using wearable sensors and machine learning techniques.

Several approaches to stress detection have been explored by researchers around the globe, including facial expression recognition [4], speech analysis [5], and wearable devices [3]. While these methods offer valuable insights, they often focus on single physiological factors and lack a comprehensive approach. This limitation is especially relevant in the context

ISSN (online): 2581-3048
Volume 7, Issue 10, pp 609-626, October-2023
https://doi.org/10.47001/IRJIET/2023.710081

of IT professionals, where a more integrated solution is needed.

One emerging area of research focuses on leveraging facial dynamics for stress assessment. Facial expressions convey a wealth of emotional information, and dynamic hierarchical attention mechanisms have been proposed to allocate attention to different facial features based on their contribution to stress levels [6]. This innovative approach holds the potential to revolutionize stress monitoring and make it more effective and reliable.

Furthermore, there is a growing need for personalized stress management strategies and proactive interventions. Machine learning techniques have shown promise in this regard, offering the potential for systematic analysis of physiological data and real-time insights to help individuals manage stress more effectively. A literature review [7] written by Mittal et. Al., discuss how ML can be used for stress management in educational and workplace environments. The IT industry presents a unique set of challenges related to stress and emotion management. Long hours, tight deadlines, and continuous learning demands can lead to stress and emotional burnout among IT professionals.

In summary, stress is a pervasive issue with significant implications for health and well-being. While various approaches to stress detection and management exist, there is a need for integrated and personalized solutions, particularly in high-stress environments like the IT industry. Advances in machine learning offer promising avenues for addressing this challenge and improving the overall quality of life for individuals dealing with stress.

II. LITERATURE SURVEY

All research projects face their own set of constraints and issues that will need attention in the future. These could stem from factors like time constraints, limited access to necessary resources, technological advancements post-research, and evolving trends. Below are some of the literatures where certain limitations that should be addressed were identified.

In the referenced paper [8], Priyangka John and their team present an application that employs brain wave analysis to detect stress levels and aids in alleviating stress through the incorporation of music or audio waves. Initially, stress detection is performed by analyzing the user's brain waves with the aid of a headgear, and the application subsequently aids in reducing the detected stress level using Binaural beats and Solfeggio Frequency.

A potential limitation identified with this approach lies in the accessibility of the headgear, as not every individual may have access to it. While the application approach is functional, its effectiveness is confined to audio waves and music for stress relief, which somewhat limits its functionality in this regard.

In contrast, the system outlined in this paper employs a multi-faceted approach, detecting stress through keystroke dynamics, mouse interactions with HRV dynamics, and emotions through facial dynamics. The stress predictions from these various sources are then aggregated and fed into a reinforcement learning-based activity recommendation system. This system provides personalized stress-relieving activity recommendations to users, offering a more comprehensive and advanced version of the Relax App application discussed earlier.

A work [9] conducted by Kalansooriya et al. developed a social media application named “xīnlǐ,” which is designed to assist people in improving their mental health. Their solution primarily focuses on emotions. For better accuracy, they have designed a machine learning-based multimodal architecture to identify one of seven basic emotions: happiness, surprise, contempt, sadness, fear, disgust, and anger. In their multimodal architecture, they have incorporated a CNN-based emotional predictor using user facial expressions, an LSTM (Long Short-Term Memory)-based algorithm to predict emotions from the semantics (meaning) of the text contained in user video logs, a CNN-based emotional predictor using speech in user video logs, and, lastly, they have aggregated these models with a neural network. Once the emotion is identified, they provide emotion alleviation activity recommendations with a Q-learning-based recommendation engine.

Comparing their solution with ours, unlike theirs, our solution is designed to run as a background application on the computer, so users will not have to spend extra time to benefit from our application, except for the activity attempting part. We also propose to use a multimodal architecture not only to detect emotions but also to detect stress without interrupting the user's workflow. The "xīnlǐ" app requires user video logs to detect users’ emotions, meaning they have to upload videos of themselves daily. Our app will not require any action from users to detect stress and emotion. Additionally, our app will measure stress levels and emotions of the users every one-minute cycle, providing near-real-time data. The RS (Recommendation System) they used is a personalized RS developed with Q-learning, an RL (Reinforcement Learning) technique.

Our RS is an upgraded version of this system, which enhances their system architecture by utilizing DRL (Deep Reinforcement Learning) techniques. Our RS employs a DQN (Deep Q-Network) to approximate the action-value function,

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 609-626, October-2023

https://doi.org/10.47001/IRJIET/2023.710081

making it more scalable than their solution. One noteworthy improvement is that our system also recommends a collection of activities (slate recommendations) for a given state, rather than just one activity. This is an open research area, as discussed in the literature survey section, because current RL algorithms cannot handle this problem [10]. Additionally, our proposed system is designed to provide activity recommendations for both stress relief and negative emotion alleviation purposes; hence, it has two objectives. Therefore, we use MORL (Multi-Objective Reinforcement Learning) techniques, as discussed in the methodology section.

Furthermore, the research [11] done by Can and his team, presents a valuable exploration into automatic stress level detection using physiological signals from wrist-worn devices. Their work focuses on the applicability of such a system in real-life settings, acknowledging the challenges posed by unrestricted movements and the resultant artifacts. A key aspect of their contribution lies in the development of novel artifact detection and removal strategies tailored to specific sensors, supported by scientifically validated performance metrics. Furthermore, they extract features from heart activity, skin conductance, and accelerometer signals and employ machine learning algorithms for stress level classification. This aspect is particularly relevant to our research gap as it aligns with our emphasis on multifaceted stress detection.

One significant dimension of their work is the empirical testing of their system in a real-life scenario involving participants in an algorithmic programming summer camp. The extensive data collection over nine days from 21 participants provides valuable insights into the practicality and effectiveness of wearable devices for stress detection in dynamic contexts. This parallels our objective of assessing the real-world utility of our stress management system.

Moreover, their research addresses several pressing research problems, including the comparative evaluation of stress detection model performance across different wearable devices, the impact of interpolation, aggregation window sizes, and artifact detection thresholds, as well as the discriminative effect of each sensor modality. These aspects contribute to a broader understanding of the challenges and opportunities in utilizing wearable technology for stress management, complementing our approach of integrating multiple stress detection methods.

Additionally, their investigation into the performance of person-specific and general models aligns with our goal of personalizing stress management recommendations. The comparison between personalized and general models could offer insights into the effectiveness of tailored interventions, which is a key aspect of our research. In summary, the work by [12] provides valuable insights into the practical challenges and potential solutions in real-life stress detection scenarios using wrist-worn devices, offering pertinent comparisons and empirical results that augment our research focus.

“A smart stress reduction system”- a research [13] done by Can and his team is focused on stress detection primarily using heart rate (HR) monitoring via an external mouse, shedding light on physiological responses to stress. However, this research predominantly emphasized HR-based stress detection, leaving room for a more comprehensive approach. Our study addresses this gap by integrating multiple stress detection methods, including HR monitoring, keystroke dynamics, and facial emotion analysis, to capture a holistic view of stress encompassing both physiological and behavioral markers.

While they introduced stress reduction methods, their exploration of context-awareness was limited. In contrast, our research extends the concept by introducing a sophisticated context analyzer. By identifying stress through various modalities and incorporating context, our system provides personalized and timely stress reduction recommendations. For instance, if heightened stress levels are detected during a demanding task, our system can recommend context-specific stress-reduction activities, such as mindfulness exercises or short breaks, to address the user's immediate needs.

A distinctive aspect of our research is the inclusion of activity recommendation in the stress and emotion management process. While previous work primarily focused on stress detection, our study goes beyond by suggesting activities to mitigate stress and alleviate emotions effectively. Leveraging multiple stress detection methods and context analysis, we offer personalized activity recommendations. For example, based on stress cues from HR data and facial emotion analysis during work sessions, our system may recommend tailored stress-reduction and emotion alleviation activities, like a brief walk or relaxation exercises, to address the user's stress within their specific context. Our research bridges existing gaps in stress management by integrating multiple stress detection methods, enhancing context-awareness, and introducing personalized activity recommendations. This comprehensive approach facilitates a deeper understanding of stress and provides practical interventions for effective stress reduction. In the subsequent sections, we detail our methodology, present experimental results, and discuss the implications of our findings, emphasizing the multifaceted nature of our stress management system.

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 609-626, October-2023

https://doi.org/10.47001/IRJIET/2023.710081

III. PROBLEM DEFINITION

Engaging in prolonged computer screen work poses a significant threat to individual well-being, particularly within the IT sector. The prevalence of stress among IT professionals is a well-documented concern, stemming from factors such as the increasing adoption of remote work, which can lead to isolation, as well as the demanding workloads that necessitate extended periods in front of a screen. Despite the IT industry's reputation for flexibility, it presents both advantages and drawbacks. In this context, occupational stress has emerged as a substantial challenge, with an escalating number of IT professionals grappling with heightened stress levels.

As emphasized earlier, whether arising from the isolation of remote work setups or the relentless workloads, addressing stress in a timely manner is imperative. The primary research problem addressed by this comprehensive project is to determine how to accurately detect users' stress levels and emotions without causing disruption to their day-to-day work. This will be achieved by leveraging the commonplace equipment they use in their daily tasks and subsequently assisting in alleviating the detected stress levels and emotions.

Addressing this escalating concern promptly is imperative. Whether stemming from the isolation of remote work or the relentless workloads, it is crucial to develop a solution that detects and alleviates stress without disrupting the user's daily workflow. To this end, this comprehensive project seeks to answer the following primary research problem:“How to detect users' stress levels and emotions without interrupting or impacting their day-to-day work, utilizing the simple equipment they use every day, and assist in relieving the detected stress levels and alleviate the detected emotion?”

To systematically address this research problem, four sub-research problems have been delineated, each tackled by individual components:

- How to identify stress levels from keyboard dynamics?
- How to identify stress levels from an external mouse equipped with an HR sensor?
- How to identify emotion from facial features using a web camera?
- How to relieve stress and alleviate emotion through computer-based activities and how to recommend them to users?

The forthcoming sections of this report will expound upon the approach, its properties, functionality, and implementation in detail.

IV. METHODOLOGY

As discussed in the above sections the overall project is a combination of 4 main components where each of them has its own contribution towards the overall project.

4.1 Data Collection

The collection of data was conducted via various platforms such as Kaggle, WhatsApp, Microsoft Teams, Zoom, Microsoft/Google Forms etc. When considering stress related discussions and brainstorming sessions were conducted via Microsoft Teams, Zoom, WhatsApp. Data relating to views and ideas on certain aspects of stress and the project were acquired via Microsoft / Google Forms.

4.1.1 Keystroke Dynamics-based Stress Detection Component

The Keystroke Dynamics-based Stress Detection Component of this research relies on a meticulously curated dataset comprising keystroke, mouse movement, and application usage data from two individuals engaged in computer-based tasks. This dataset is augmented with crucial physiological and affective metrics, including fatigue levels, Photographic Affect Meter (PAM) values, perceived stress levels, energy levels, and overall pleasantness. Notably, these metrics are recorded at regular intervals ranging from 5 to 30 minutes, providing a temporal resolution conducive to capturing dynamic shifts in the participants' emotional states.

Each folder within the dataset comprises approximately 14,000 rows of meticulously recorded data, offering a comprehensive view of the participants' computer usage behavior over extended periods.Prior to analysis, the raw dataset underwent an extensive preprocessing phase to enhance its suitability for subsequent modeling. This involved rigorous quality checks, including outlier detection and removal, normalization of feature scales, and addressing any missing or erroneous data points. Additionally, temporal alignment techniques were applied to synchronize keystroke dynamics, mouse movement, and application usage data with the corresponding physiological and affective metrics.

4.1.2 Heart Rate Variability-based Stress Detection

Our pursuit of accurate stress detection and prediction begins with the utilization of a meticulously curated dataset sourced from the SWELL knowledge work (SWELL-KW) dataset. This dataset provides a comprehensive collection of measurements of heart rate variability (HRV) indices, coupled with corresponding stress levels. The SWELL-KW dataset, a product of experiments conducted at the Institute for Computing and Information Sciences at Radboud University,

ISSN (online): 2581-3048
Volume 7, Issue 10, pp 609-626, October-2023
https://doi.org/10.47001/IRJIET/2023.710081

captures the multifaceted nature of stress responses during knowledge work activities. The experiments involved 25 subjects engaged in typical office work scenarios, including tasks like report writing, presentations, email correspondence, and information retrieval. These subjects were exposed to common workplace stressors such as unexpected email notifications and time pressure.

In addition to HRV indices, the SWELL-KW dataset encompasses a wide array of data modalities, including computer logging, facial expressions, body postures, ECG signals, and skin conductance. This holistic approach ensures a comprehensive understanding of stress responses within real-world work contexts.

4.1.3 Facial Dynamics-based Emotion Detection

At the core of our research lies the Fer2013 dataset, a comprehensive repository of grayscale facial images designed to encapsulate a wide array of emotional states. Each of these images is standardized to a size of 48x48 pixels through an automated registration process, ensuring uniformity and centralization. Boasting an extensive collection of approximately 30,000 images, this dataset presents a diverse spectrum of facial expressions, spanning across seven distinct emotion categories: Angry, Disgust, Fear, Happy, Sad, Surprise, and Neutral.

It's worth noting that the distribution of these emotions within the dataset exhibits some variation. Specifically, the Disgust category is represented by a more modest count of 600 images, while the remaining labels feature a more substantial presence, each accounting for nearly 5,000 samples. The Fer2013 dataset forms the indispensable foundation upon which we build, enabling the training, validation, and evaluation of stress detection models that hinge on the dynamics of facial expressions.

4.1.4 Reinforcement Learning-based Recommendation System Component

To create a stress-relieving and negative-emotion coping activity pool, there were three primary methods of data collection:

1. Gathered activity preferences from potential users via Google Form - an online questionnaire.
2. Collected emotion alleviation activities dataset from researchers who had conducted the 'Xilini' [9] research.
3. Created an activity pool with the consultation of our external supervisor, who is a clinical psychologist.

By utilizing these methods, we were able to effectively gather the necessary data to create a comprehensive activity pool that will help users cope with stress and negative emotions**.**

4.2 Approach

The system architecture, illustrated below, embodies the core of this research project. The primary objective is to develop a machine learning and reinforcement learning-based application for stress identification and reduction, tailored specifically for IT workers. The system aggregates user data from multiple sources, including keyboard dynamics, heart rate dynamics, and facial dynamics, as depicted in the diagram. Facial dynamics are captured using a camera that captures images of the user's face. Subsequently, a reinforcement learning method is devised to learn and offer individualized stress-relieving activities based on the user's stress level. These activities are meticulously customized to align with the user's preferences and needs, offering a highly efficient intervention for stress release.

To ensure precise stress detection and individualized stress reduction recommendations, the application's performance is assessed through thorough user testing and cross validation. The overall goal of the study project is to offer a more efficient method of stress management that is accurate, individualized, and catered to specific needs.

The proposed solution aims to create a stress monitoring and alleviation application tailored for IT Professionals. The system will comprise four main components:

- Detecting stress through keystroke dynamics.
- Detecting stress via an external mouse using an HRV sensor.
- Detecting stress through facial analysis using a webcam.
- Providing recommendations and suggestions for users to undertake to decrease their stress levels.

The initial three components will assess user stress levels based on keyboard, mouse, and webcam data. The fourth component will utilize the prediction data generated by the first three components to offer suggestions for small stress-relieving activities through an integrated desktop application.The overall system flow is as follows:

- Users perform their regular work tasks.
- The components monitor keyboard and mouse data, which is then processed by their respective machine learning models to predict the user's stress level.
- The first two components generate prediction outputs in 20-minute intervals and collect prediction data from both the mouse and keyboard.

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 609-626, October-2023

https://doi.org/10.47001/IRJIET/2023.710081

- The camera will also be prompted every 20 mins and take a snap of the current state of the person to send to the machine learning model.
- After an hour (60 mins) has elapsed, the prediction data (three data sets from both the keyboard and mouse) collected in 20-minute intervals throughout the hour are analyzed to determine the user's current stress level.
- This output is subsequently transmitted to the final component, the recommendation system. It evaluates the received data, accesses its behavioral database and activity pool, and suggests stress-relieving activities for the user to help them reduce their stress levels.

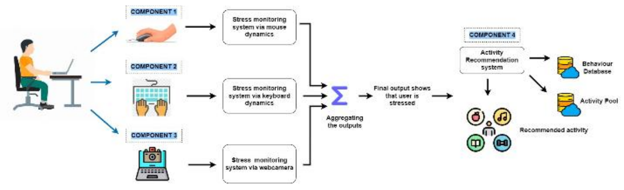

Figure 1: Overall System Architecture

The initial three components will assess user stress levels based on keyboard, mouse, and webcam data. The fourth component will utilize the prediction data generated by the first three components to offer suggestions for small stress-relieving activities through an integrated desktop application. The overall system flow is as follows:

4.2.1 Keystroke Dynamic Based Stress Detection Component

Keystroke dynamic based stress detection itself is a novel approach when it comes to the stress detection domain. The components predictions are derived from the keystroke dynamic related data that is fed to the machine learning model that is built into the system.

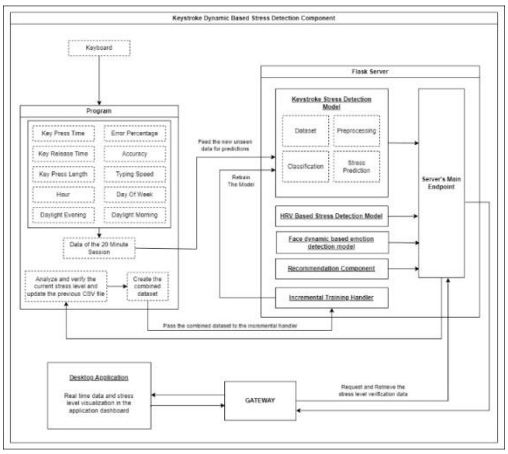

Figure 2: Keystroke Dynamic Based Stress Detection Component Workflow

First, a previously created dataset containing the keypress dynamics and stress levels is used in training the machine learning model. This data is analyzed, preprocessed, split and fed to a machine learning model. The model is trained based on the data that was fed into it. After the training process, the trained model is hosted in a Flask server, and it is used to predict the current stress level of the user based on their keystroke dynamic data that's fed to it. The model that was trained will perform as a base model, and as the system is being used, the base model will be retrained using new datasets utilizing an incremental learning-based approach while functioning as the base model for predictions.

The approach includes a program developed using Python language which will detect and record the keystroke dynamic related data and send the gathered data to the Flask based server. From the server the trained machine learning model that is hosted in it will acquire the gathered data for generating predictions. The program is also responsible for acquiring stress level data that was derived on the same timeframe from the specific database instance and analyze and prepare that data for the incremental learning-based approach for base model retraining.

The prediction in this approach is generated using a machine learning model (Random Forest). The initial model was trained from a currently existing dataset which contains the keypress time, release time, keypress length and stress level.

The dataset contains 12414 rows of keystroke dynamic related data and another dataset with approximately 14500 rows of keystroke dynamic related data. Both datasets were utilized in training the base models during the model selection phase of this project. During this phase the holdout validation technique was utilized to obtain a performance estimation on the model which would assist in the model selection. Before proceeding to train the models, the dataset was loaded into the Google Colab notebook and then the data was preprocessed and applied feature engineering techniques and tested with various approaches of these preprocessing techniques. The process included label encoding, one-hot encoding techniques, date time filtering techniques and data balancing techniques. During the data preprocessing phase, new features were derived from the dataset. They are Daylight_Evening, Daylight_Morning, Hour, Day_Of_Week.

During the model selection phase of this component various machine learning models were trained to assess the accuracy and the prediction rate and further identify what model would best suit the current scenario. The dataset contains time related data however the target variable is a categorical variable. Therefore, both possibilities were tested

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048
Volume 7, Issue 10, pp 609-626, October-2023
https://doi.org/10.47001/IRJIET/2023.710081

using two time series related models and two classification-based models.

The Auto Regressive Integrated Moving Average (ARIMA) model and the Long Short-Term Memory Model (LSTM) models were trained with the current dataset. The ARIMA Model was trained, however the prediction on the test set was not accurate since it was only able to predict only one state.

Then the LSTM model was trained and was tested using the test split and it generated quite good results when trained only with the stress level variable it was able to predict the upcoming stress value, but the model was not able to predict the proper stress level when it was trained with all the features. It was also only able to predict one state. Therefore, both time series related models were removed from the selections.

The SVM model training got an accuracy of 0.595479 before the data balancing techniques were used. However, since SVMs require balanced datasets to be trained the training was not accurate and the model could only predict one state.

Then the model was retrained after using an oversampling method on the dataset. This produced a balanced-out dataset which resulted in increasing the accuracy to a level of 0.777722.

Then the Random Forest Model was trained. Which first got an accuracy of 0.593117 without the feature engineering techniques but after the feature engineering was applied the model's accuracy increased up to 0.784413 and was able to predict the stress levels on the test set up to a good standard. During further tests the dataset was further preprocessed using data balancing techniques such as oversampling. The accuracy level increased up to 0.835934.

Since both SVM and Random Forest models showed promising results, the model was chosen from the best accuracy and F1 score. Hence, the Random Forest model was chosen to perform as the base model for the component. As discussed earlier 2 datasets were used during the model selection phase,

- Dataset 1: Had 12414 rows of keystroke dynamics related data.
- Dataset 2: Had 14818 rows of keystroke dynamics related data.

Both datasets were used in the testing phase and the accuracy results are as follows. The results for dataset 1 are shown in the following pictures.

```
[13] # Evaluate the model
     accuracy = accuracy_score(y_test, y_pred)

     # Calculate precision
     precision = precision_score(y_test, y_pred, average='weighted')

     # Calculate recall
     recall = recall_score(y_test, y_pred, average='weighted')

     # Calculate F1 score
     f1 = f1_score(y_test, y_pred, average='weighted')

     # Print the evaluation metrics
     print("Accuracy:", accuracy)
     print("Precision:", precision)
     print("Recall:", recall)
     print("F1 Score:", f1)

     Accuracy: 0.833266210229561
     Precision: 0.8217277667403249
     Recall: 0.833266210229561
     F1 Score: 0.819544478097879
```

Figure 1: Accuracy of the model without oversampling (DS-1)

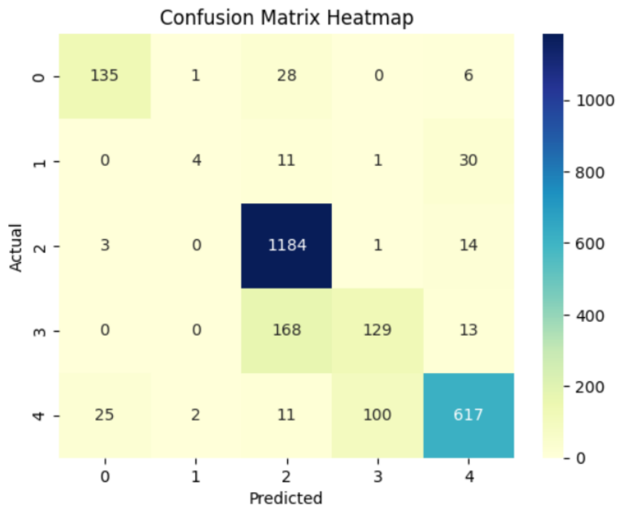

Figure 2: Confusion matrix without oversampling (DS-1)

```
[21] # Evaluate the model
     accuracy = accuracy_score(y_test, y_pred)

     # Calculate precision
     precision = precision_score(y_test, y_pred, average='weighted')

     # Calculate recall
     recall = recall_score(y_test, y_pred, average='weighted')

     # Calculate F1 score
     f1 = f1_score(y_test, y_pred, average='weighted')

     # Print the evaluation metrics
     print("Accuracy:", accuracy)
     print("Precision:", precision)
     print("Recall:", recall)
     print("F1 Score:", f1)

     Accuracy: 0.8359335681932561
     Precision: 0.8487462576667454
     Recall: 0.8359335681932561
     F1 Score: 0.8363306604335913
```

Figure 3: Model accuracy after oversampling (DS-1)

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 609-626, October-2023

https://doi.org/10.47001/IRJIET/2023.710081

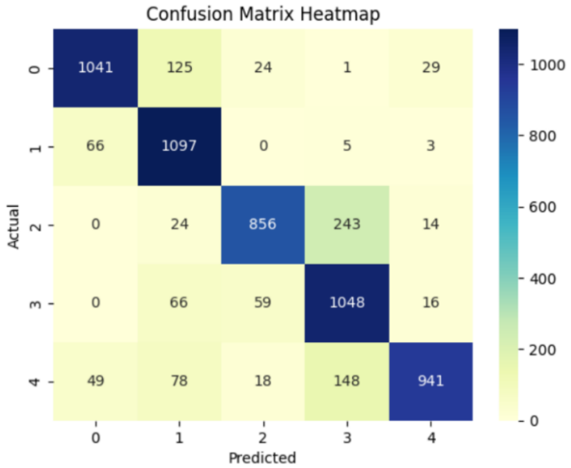

Figure 4: Confusion Matrix after oversampling (DS-1)

After testing with dataset 1 for further confirmation the same procedure was followed with dataset 2 as well. The results for dataset 2 are shown in the following pictures.

```
[31] # Evaluate the model
     accuracy = accuracy_score(y_test, y_pred)

     # Calculate precision
     precision = precision_score(y_test, y_pred, average='weighted')

     # Calculate recall
     recall = recall_score(y_test, y_pred, average='weighted')

     # Calculate F1 score
     f1 = f1_score(y_test, y_pred, average='weighted')

     # Print the evaluation metrics
     print("Accuracy:", accuracy)
     print("Precision:", precision)
     print("Recall:", recall)
     print("F1 Score:", f1)

     Accuracy: 0.7844129554655871
     Precision: 0.7837299408279815
     Recall: 0.7844129554655871
     F1 Score: 0.7829207752779439
```

Figure 5: Model accuracy without oversampling (DS-2)

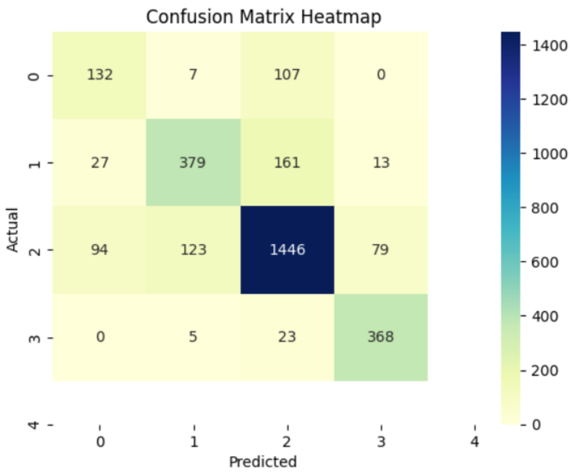

Figure 6: Confusion Matrix without oversampling (DS-2)

```
# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)

# Calculate precision
precision = precision_score(y_test, y_pred, average='weighted')

# Calculate recall
recall = recall_score(y_test, y_pred, average='weighted')

# Calculate F1 score
f1 = f1_score(y_test, y_pred, average='weighted')

# Print the evaluation metrics
print("Accuracy:", accuracy)
print("Precision:", precision)
print("Recall:", recall)
print("F1 Score:", f1)

Accuracy: 0.8230670287265971
Precision: 0.8251101737852043
Recall: 0.8230670287265971
F1 Score: 0.8180109558892035
```

Figure 7: Model accuracy after oversampling (DS-2)

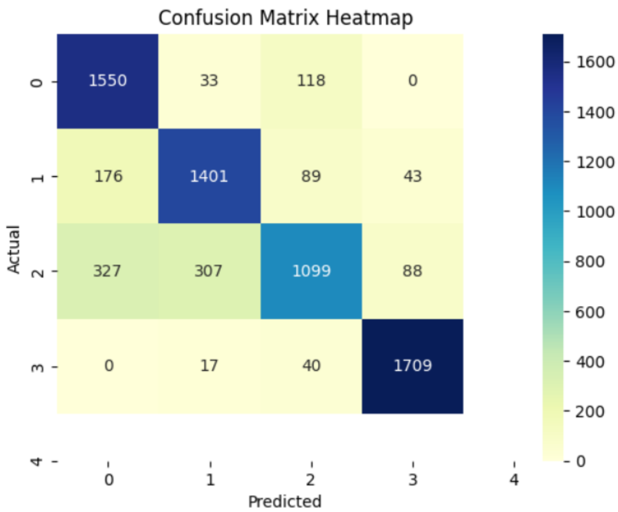

Figure 8: Confusion Matrix after oversampling (DS-2)

TABLE I: Model Test Final Results

Model Name	Accuracy	F1 Score
Support Vector Machine	0.777722	0.775083
Random Forest	0.835934	0.836331

After the various tests that were taken to find the best feature set and the finalized dataset the final model training dataset consisted of Key_Press_Length, Stress_Level, Hour, Day_Of_Week, Daylight_Evening, Daylight_Morning as the feature set for the stress level predictions and the Table 2 displays the accuracy and F1 Score that was obtained with the use of the finalized dataset.

The base random forest model was then hosted in the Flask server allowing it to read unseen data sent to it and predict stress levels and communicate the predicted stress level to the user.

As mentioned before the unseen data relating to the keystroke dynamics is generated via a python program which

ISSN (online): 2581-3048
Volume 7, Issue 10, pp 609-626, October-2023
https://doi.org/10.47001/IRJIET/2023.710081

can capture and derive the following dynamics from the user's session.

- Key Press time (PT), Key Release time (RT) Key Press Length (KPL), Error Percentage (EP), the Accuracy (A), Words Per Minute (WPM), Hour (H), Daylight Evening (DE), Daylight Morning (DM), Day of week (DOW).

To ensure the utmost privacy of users' data, the program implements a precautionary measure by replacing alphanumeric key labels with randomized symbols. This safeguards the actual content from any unintended exposure.

The metrics are culled from 20-minute sessions when the user interacts with the system. Each session generates a comma-separated value (CSV) file, meticulously logging all pertinent keypress dynamic data. Following each 20-minute cycle, this compiled data is dispatched to the Flask server. Simultaneously, the program resets its variables, readying itself for new session data. The CSV file, pivotal for model retraining, is persistently stored on the client's local machine. However, it's slated for deletion within 24 hours post the completion of model retraining, providing an additional layer of protection for users' data privacy.

The Flask server dutifully retrieves the sent 20-minute session data, funneling it to the hosted Random Forest Model for analysis. Post scrutiny, the model generates the prediction. This forecast is then routed back to the main server endpoint, making its way through the gateway to the application. Here, the stress level data is aggregated and presented. Subsequently, the compiled stress level returns to the server, readying the persistent CSV file for model retraining. Given that it excludes the Stress Level variable, the aggregated stress level is appended to the CSV file. This amalgamation with the initial dataset lays the foundation for an impeccable dataset, poised for incremental learning in bolstering the base model.

A background task, scheduled every 24 hours, triggers an API call to the server, initiating the model retraining process. The procedure commences with a comprehensive test of the model's current accuracy, meticulously recorded. After retraining, the accuracy undergoes another round of evaluation. These two accuracy values are then juxtaposed. The newly retrained model is only embraced as the base model by the system if its accuracy equals or surpasses that of the extant base model. Should the system favor the new model, it promptly supplants the current base model, seamlessly taking on the mantle.

This iterative model retraining, integral to the system's ongoing use, culminates in an escalating accuracy for the base model. This, in turn, refines the precision of stress level predictions, personalized to each user. Consequently, the system adeptly forecasts user stress levels, harnessing their keystroke dynamics, all the while refining the predictive model in stride, without impinging on users' tasks.

4.2.2 HRV Based Stress Detection Using an External Mouse Component

During the project, we carried out tests to identify the most suitable heart rate sensor for accurately measuring stress levels. Through evaluations of the sensor performance and accuracy, we aimed to identify a device that is cost effective and capable of providing reliable stress level measurements for developers and other users.The sensors we evaluated are listed below.

- KY-039 heartbeat sensor is designed to detect the pulse from the human finger by using a phototransistor to detect the amount of light passing through the finger. However, this sensor has limitations since it has limited accuracy and reliability as it is prone to interference from external light sources.
- MAX30100 heart rate and pulse oximetry sensor are a compact model that not only measures heart rate but also estimates oxygen saturation (SpO2) levels in the blood. It uses PPG signals to measure changes in blood volume. The main issue faced with this sensor is that it is limited to basic health metrics and may not be suitable for in depth HRV analysis. It may also be still affected by motion artifacts to some extent.
- GY-MAX30102 sensor is an upgraded version of the MAX30100 sensor and is designed to perform enhanced performance and accuracy with a larger dynamic range for measuring changes in blood volume. Despite the upgrade, this sensor may require careful calibration for optimal performance and it is still susceptible to motion artifacts in certain scenarios.
- MAX30102 sensor is an advanced optical sensor module specifically designed for health monitoring applications. It offers improved sensitivity, better signal to noise ratio and enhanced motion compensation features.

The node MCU development boards we tried out are listed below:

- ESP8266 microcontroller is known for its cost-effectiveness and simplicity, making it suitable for basic Wi-Fi-enabled projects. However, it has limited processing power and memory compared to more

ISSN (online): 2581-3048
Volume 7, Issue 10, pp 609-626, October-2023
https://doi.org/10.47001/IRJIET/2023.710081

advanced MCUs. It may face challenges when handling complex tasks.

- ESP32 offers more advanced features, improved processing power, and additional interfaces, making it a suitable choice for more complex applications that require both Wi-Fi and Bluetooth connectivity, as well as greater processing capabilities. Therefore, ESP32 is more optimal for a stress detection system.

The development of mouse prototypes with heart rate detection and data visualization and display unit is a new novelty as this represents an innovative approach to stress monitoring and health awareness. Furthermore, the table in the following, displays an evaluation of the sensors that were tested for, in this context.

TABLE II: Sensor Analysis through Testing Codes

Sensor Model	Price (LKR)	Accuracy (%)	Sample Rate (Hz)	Power Consumption (mA)	Signal-to-Noice Ration (SNR)	Additional Notes
KY-039	165.00	85	50	10	30 dB	Compact design
MAX30100	550.00	90	100	8	35 dB	Low power usage
GY-MAX30102	590.00	92	120	9	38 dB	High SNR
MAX30102	590.00	95	80	7	40 dB	Best performance

First Prototype (MAX30100 and ESP8266): During the initial prototype, we combined the MAX30100 sensor with the ESP8266 microcontroller along with an old display unit. In previous studies, the combination of this sensor and microcontroller have been successfully used in a pulse oximeter.

The purpose of connecting the MAX30100 sensor is to measure the heart rate and the blood oxygen saturation level values from the user's fingers. Meanwhile the ESP8266 microcontroller would process the sensor data and control the display unit to visualize the collected data.

Second Prototype (MAX30102 and ESP32): In the second prototype, we incorporated the MAX30102 sensor which is more advanced and the ESP32 microcontroller. The same old display unit was used during this prototype as well.

The MAX30102 sensor has better accuracy and reliability in measuring the heat rate variability and the SpO2 values. Since the ESP32 microcontroller is more powerful, it enabled faster processing of the collected data. The combination of these devices can be used for heart rate and pulse monitoring effectively.

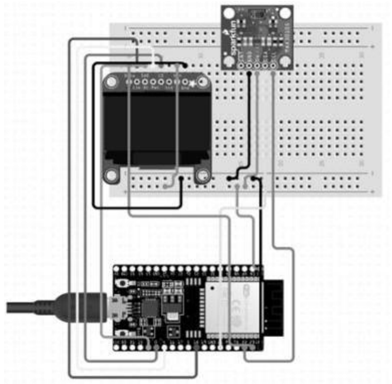

Figure 119: Circuit Diagram

To begin with, a previously crafted dataset which contains measurements of the heart rate variability (HRV) indices and corresponding stress levels computed from the multimodal SWELL knowledge work (SKELL – KW) is used as the foundation to train our machine learning model. This dataset undergoes analysis and preprocessing. Then, it is divided into segments and inputted into the machine learning model. The models training process involves exposing it to this dataset, enabling it to learn the underlying patterns and associations. While the base model continues to generate stress level predictions based on the input HRV and Spo2 values, it concurrently undergoes incremental learning and integrates new data sets into the model's existing knowledge.

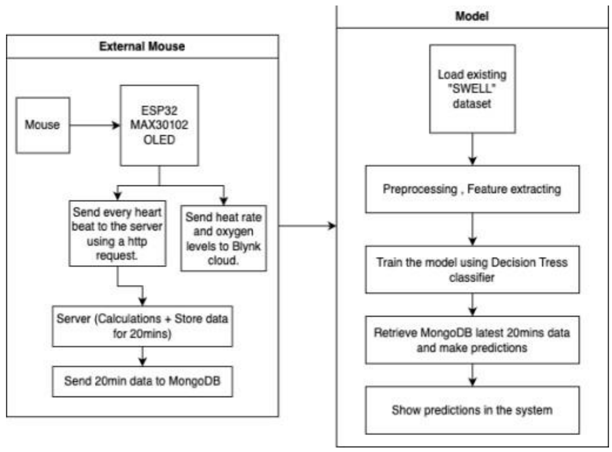

Figure 1210: Stress Detection via HRV Sensor in Mouse Component Workflow

The methodology section outlines the systematic approach used to analyze users' HRV and SpO2 values through the computer mouse and use for stress detection. Machine learning algorithms have been frequently used for user stress detection in human computer interactions. This involved data preprocessing, balancing, feature selection, and utilizing the Decision Tree Classifier as the primary model due to its interpretability and classification capabilities. By using decision trees, prominent features that influence stress can be

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 609-626, October-2023

https://doi.org/10.47001/IRJIET/2023.710081

identified and the results obtained can be used to narrow down the approach to reducing stress as proved through a study which has been conducted before. The feature selection process aimed to identify key parameters for the model's predictive accuracy. Subsequent subsections detail each step, revealing the strategies employed to derive insights from the data.

4.2.3 Facial Dynamic Based Emotion Detection Component

The dataset used for this component played a pivotal role in the assessment of stress levels, covering various modalities including facial dynamics, heart rate, and keyboard dynamics. A significant component of this research hinged on the Fer2013 dataset. This dataset, containing grayscale facial images standardized to 48x48 pixels, housed approximately 30,000 images. It encompassed a diverse spectrum of facial expressions, including seven distinct emotional categories: Angry, Disgust, Fear, Happy, Sad, Surprise, and Neutral. While some variation existed in the distribution of these emotions within the dataset, it provided a robust foundation for the training, validation, and evaluation of stress detection models centered on facial dynamics.

In addressing the core task of stress detection, two meticulously developed model architectures were employed: Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs). Both models were enriched with hierarchical attention mechanisms, dynamically allocating weights to facial features based on their significance. The CNN-based model effectively captured critical facial features using its hierarchical attention mechanisms, while the ViT-based model focused on discerning the importance of unique facial regions in stress detection. This process commenced with comprehensive dataset preprocessing and refinement, ensuring data readiness for training. A dedicated 'Generate_data' class facilitated the specification of data paths, establishing a structured foundation for subsequent model development.

A critical aspect of this work was the implementation of a neural network model designed for emotion recognition. This model leveraged convolutional neural networks (CNNs) enhanced with spatial attention mechanisms. These mechanisms played a crucial role in assigning weights to input image features, discerning their relevance for the task at hand. The integration of spatial attention further enhanced the model's capacity for precise feature extraction, contributing to accurate emotion recognition.

The methodology also encompassed the exploration of a ViT model for emotion recognition, adding a novel dimension to the research. This model architecture leveraged a transformer-based approach, initially designed for natural language processing, and applied it to image classification. The ViT treated images as sequences of patches, processing them through multiple transformer layers. This innovative approach allowed the model to effectively capture complex dependencies and relationships between patches, ultimately leading to accurate emotion recognition.

In summary, the methodology presented a comprehensive approach to stress detection, encompassing data collection, dataset description, system architecture, component architecture, tool and library usage, and detailed model architectures for both CNN and ViT models. Each step was intricately designed to address the unique challenges of stress detection among IT professionals, culminating in a holistic framework for effective stress management.

4.2.4 Activity Pool Development and Recommendation System Component

There are mainly two parts under this component. The first one is to develop a computer-based activity pool which helps users to alleviate emotions and relieve stress. The second one is to develop the activity recommendation system.

Earlier days, recommendation problem was seen as a prediction problem. Thus, there were several ways to develop recommendation systems using machine learning technologies are collaborative filtering, content-based filtering and hybrid approaches which utilize both collaborative filtering and content-based filtering [11]. But it now widely considered as a sequential decision making problem. Reinforcement learning is the de facto machanism in machine learning to solve sequential decision making problems [11].

However, it is difficult to employ reinforcement learning algorithms for a practical recommendation problem as they do not scale well. To overcome this problem several approaches such as function approximation techniques can be employed with reinforcement learning algorithms. One such function approximation method is to use a neural network. Using deep learning techniques such as neural networks with reinforcement learning is called deep reinforcement learning [15].

We empolyed Deep Q learning which is a greedy policiy reinforcement learning algorithm which uses a CNN to approximate Q values, to develop our recommendation system. Our recommendation system had to requirements which are to alleviate emotions and relive stress. However, these reinforcement learning algorithms are designed to optimize one long term objective only. To make our system optimize both the requirements we employed MORL.

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 609-626, October-2023

https://doi.org/10.47001/IRJIET/2023.710081

While there are several approaches to apply MORL techniques, if the objectives do not clash, single policy approaches can be used to optimize both the objectives at the same time. The weighted sum approach is one such single policy approach that can be used if the weight of the objectives is known [17]. Our recommendation system we know alleviate emotions and relieve stress need to be equally optimized and they do not clash each other. Hence the weighted sum MORL approach was used in our recommendation system.

Also, there was a requirement in our system to recommend multiple activities. Q learning alone cannot handle this problem as Q learning only gives action which has the highest Q value at for the given state. Considering actions as activities, it can only recommend one activity. SlateQ is a algorithm developed by Google researchers which is capable of recommending slates of items with Q learning [16].

We employed Deep Q learning, MORL techniques and SlateQ to develop our recommendation system.

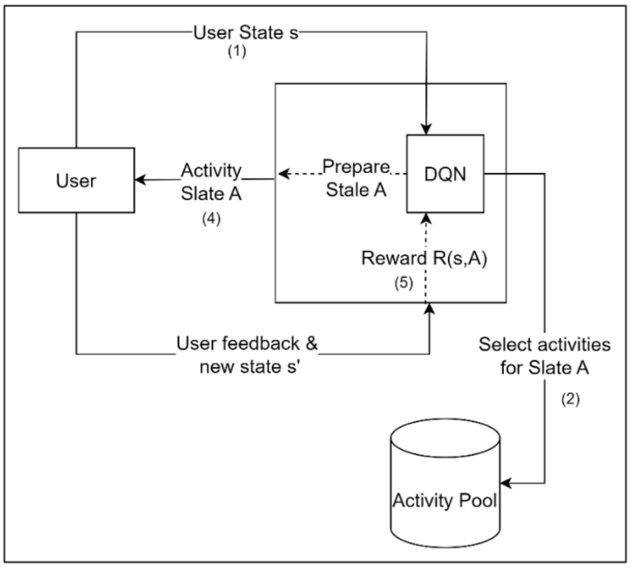

Figure 1311: Workflow of the Recommendation Component

As the first step, identifies stress leves, emotions, user interaction information will be sent to our recommendation engine from the client application as the user state. Based on the provided state, our recommendation engine finds the most suitable 4 activities as recommendations. SlateQ algorithm is used to find the 4 most suitable activities, Q value of selected activity a in state s can be represented as follows,

$$Q(s,a) \leftarrow \alpha(r + \max_{A' \in A} \sum_{j \in A'} P(j|s',A')Q(s',j)) + (1-\alpha)Q(s,a) \qquad -(1)$$

Where, α is the learning rate, r is immediate reward received after attempting activity a in state s, s' is next state, P(j|s',A') is probability of selecting activity j from next slate A'.By applying weighted sum MORL approach to optimize two objectives the Q value can be formulated as below,

$$Q(s,a) = W_e Q_e(s,a) + W_s Q_s(s,a) \qquad -(2)$$

$$Q_e(s,a) \leftarrow \alpha(r_e + \max_{A' \in A} \sum_{j \in A'} P(j|s',A')Q_e(s',j)) + (1-\alpha)Q_e(s,a)$$

$$Q_s(s,a) \leftarrow \alpha(r_s + \max_{A' \in A} \sum_{j \in A'} P(j|s',A')Q_s(s',j)) + (1-\alpha)Q_s(s,a)$$

here W_e and W_s are weights for emotion alleviation and stress relieving objectives. $Q_e$and Q_s are Q values calculated separately for emotion alleviation and stress relieving objectives for respectfully.

Once the 4 activties are chosen, they will be sent to the client application and the user will choose one of the activties. Based on the user's feedback DQN is trained.

V. RESULTS AND DISCUSSION

5.1 Keystroke Dynamic Based Stress Detection Component

When discussing the results of the tests that were carried out to ensure that the component is functioning as expected as explained in the above sections the tests were carried out on both backend and frontend functionalities of the component.

The keystroke dynamic-based stress detection component presents a significant advancement in stress assessment, revolutionizing monitoring, and management. By capturing users' natural keyboard interactions and leveraging machine learning, it offers a non-invasive, user-centric solution.

Unlike conventional methods requiring specialized devices or controlled stress induction, this approach utilizes everyday keyboard usage. This unobtrusive data collection process allows users to continue tasks without altering behavior.

The component gathers a comprehensive set of keystroke dynamics—keypress length, press/release times, accuracy, error rate, typing speed, and contextual factors—resulting in a rich dataset. This holistic approach enhances stress prediction accuracy, capturing multiple dimensions of user behavior indicative of stress levels.

The Random Forest-based model, as detailed in the methodology, attains impressive accuracy (0.84) and F1 score

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 609-626, October-2023

https://doi.org/10.47001/IRJIET/2023.710081

(0.84). Its adaptability stands out; frequent retraining with new data refines predictions, ensuring relevance over time.

User privacy is central; no sensitive data or personal info is involved. Alphanumeric keys are converted to random symbols, safeguarding against information leaks. Users aren't inconvenienced, using keyboards as usual.

Seamless integration into daily life is facilitated by background operation within a Flask server. Regular acquisition of keystroke dynamics ensures consistent data input without disruption. This adaptability positions it as a versatile tool for stress assessment.

In summary, this component offers an innovative, practical solution to stress assessment. Its unobtrusiveness, comprehensive data collection, adaptive learning, and seamless integration make it a groundbreaking tool. By eliminating the need for specialized hardware or conscious user engagement, it heralds a new era in stress detection research.

5.2 HRV Based Stress Detection Using an External Mouse Component

The table below presents a thorough evaluation of the stress detection model's classification performance. Precision, recall, and F1-score metrics are employed to gauge the model's proficiency in accurately classifying instances and distinguishing between various stress levels.

	precision	recall	f1-score	support
0	0.76	0.72	0.74	43743
1	0.70	0.66	0.68	43819
2	0.98	0.99	0.98	44015
3	0.71	0.79	0.75	43895
accuracy			0.79	175472
macro avg	0.79	0.79	0.79	175472
weighted avg	0.79	0.79	0.79	175472

Figure 1412: Classifier Accuracy

The results clearly demonstrate an impressive overall accuracy of 79% for the model. This indicates its effectiveness in predicting stress levels with precision based on physiological data collected by the developed stress detection system. The weighted average F1-score of 0.79 signifies a well-balanced trade-off between precision and recall, affirming the model's consistent performance across diverse stress categories.

In real-world scenarios, the model exhibits a noteworthy macro average F1-score of 0.79, highlighting its ability to maintain consistent performance across different stress level categories. This resilience is crucial for ensuring dependable stress assessment under various circumstances. Additionally, the high macro average precision and recall of 0.79 underscore the model's capacity to minimize false positives and false negatives, a critical aspect in preventing unnecessary stress interventions and providing timely support to individuals experiencing elevated stress levels.

The evaluation of classification performance vividly showcases the promising capabilities of the developed stress detection model. While some variations in precision and recall among stress levels are observed, the model's overall accuracy and reliability in predictions underscore its potential utility in unobtrusive stress monitoring. These findings exemplify the model’s adeptness in handling diverse stress levels encountered in real-life situations.

The decision to utilize the mouse as the primary human-computer interaction device for our stress detection mechanism was founded on several key considerations. Through a survey, it was established that young individuals with prior technology experience perceive the mouse as more accurate and user-friendly compared to touch pads and touchscreens. This heightened accuracy is essential for capturing subtle physiological signals like heart rate variability. Furthermore, the mouse's capacity to detect users' task completion difficulty through interactions solidifies its suitability as an indicator device for stress detection. Additionally, the widespread use and familiarity of the mouse among users enhance their comfort and confidence in interacting with the system.

Moreover, developers favor the mouse due to its seamless integration into programming environments and libraries. This familiarity streamlines the implementation of our stress detection mechanism, potentially reducing development time. The mouse's precise control over graphical elements and support for complex interactions further affirm its suitability for meeting our system's requirements. In sum, the selection of the mouse as the primary input device ensures accuracy, user satisfaction, developer convenience, and effective interaction with the stress detection mechanism.

5.3 Facial Dynamic Based Emotion Detection Component

A comprehensive evaluation of the results obtained from the experiments conducted with both the Convolutional Neural Network (CNN) model and the Vision Transformer (ViT) for emotion recognition was undertaken.

The initial implementation of the Convolutional Neural Network (CNN) model yielded an accuracy ranging from the low 60s to a maximum of 66% in the emotion recognition task. However, a pivotal improvement was achieved by

ISSN (online): 2581-3048
Volume 7, Issue 10, pp 609-626, October-2023
https://doi.org/10.47001/IRJIET/2023.710081

implementing data augmentation techniques to artificially expand the training dataset. This approach led to a notable enhancement, resulting in an approximate 70% accuracy.

The confusion matrix and micro-averaged metrics provided valuable insights into the model's performance. The micro-averaged precision, recall, and F1-score were approximately 0.5795, indicating an overall balanced performance across all classes. Additionally, the macro-averaged metrics demonstrated a well-distributed performance with a macro-averaged F1-score of approximately 0.5579. This suggests that the model exhibits consistent effectiveness across the different emotion categories.

TABLE III: Confusion Matrix for CNN Model

Actual	**Predicted**						
	Angry	Disgust	Fear	Happy	Sad	Surprise	Neutral
Angry	500	50	100	30	50	50	60
Disgust	40	420	80	30	40	40	50
Fear	80	90	400	60	70	100	80
Happy	30	20	50	700	50	30	20
Sad	40	40	80	50	500	80	50
Surprise	60	50	100	40	70	600	50
Neutral	80	40	80	20	30	50	700

The ViT transformer model also demonstrated promising results. With an accuracy of approximately 80.12%, it outperformed the CNN model in terms of overall recognition. The confusion matrix illustrated a robust performance in classifying emotions, particularly excelling in categories such as "Angry," "Disgust," and "Fear."

TABLE IV: Confusion Matrix for ViT Model

Actual	**Predicted**						
	Angry	Disgust	Fear	Happy	Sad	Surprise	Neutral
Angry	590	20	50	10	20	20	30
Disgust	10	510	20	10	10	10	20
Fear	20	30	540	20	30	40	30
Happy	5	5	15	600	10	5	5
Sad	10	10	20	10	550	20	15
Surprise	15	10	30	10	20	590	15
Neutral	25	10	20	5	10	15	640

Macro-averaged precision, recall, and accuracy were calculated to be approximately 0.8007, 0.8180, and 0.8012 respectively. These metrics indicate a strong overall performance across all emotion categories, further emphasizing the effectiveness of the ViT model in this task.

The comparative analysis between the CNN and ViT models unveiled valuable insights into their respective efficiencies. While the CNN model demonstrated commendable performance, especially after the implementation of data augmentation, the ViT model showcased superior accuracy and consistency across all emotion categories. This highlights the potential superiority of the ViT model in practical applications of emotion recognition.

To validate the real-world applicability of our trained models, we implemented real-time emotion detection using a webcam feed. This demonstration showcased the models' ability to accurately recognize emotions in real-time, paving the way for applications in stress monitoring and interventions.

5.4 Activity Pool Development and Recommendation System Component

Prior to implementing our recommendation system (RS) in a live production environment where real users will engage with our reinforcement learning (RL) model, it is crucial to conduct a thorough evaluation. One significant challenge in this process is the limited availability of versatile simulation platforms designed for sequential recommender scenarios. In response to this challenge, Google has developed RecSim [31], which serves as a configurable simulation platform tailored for reinforcement learning recommender systems.

For assessing the performance of our RS, we harnessed the capabilities of RecSim. This framework allowed us to simulate user interactions and assess the system's efficacy in delivering recommendations. We set up the environment using RecSim's predefined classes, including the implementation of a document model, user model, and user choice model to replicate real-world conditions and user behaviors for the recommended slate.

In our simulated environment, each user was given a fixed time budget, with each selected activity in the recommended slate consuming 1 unit of time. The experiment concluded when the time budget was exhausted. In our simulations, each agent was tasked with selecting 4 documents (activities) to recommend from a pool of 20 candidate

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 609-626, October-2023

https://doi.org/10.47001/IRJIET/2023.710081

documents. We executed both simulations within the Google Colab runtime environment and recorded statistics using Tensorboard.

The graph below depicts the average rewards per episode over the course of time steps. At the conclusion of the experiment, the average reward for our RS with the SlateQ-utilizing agent was 159, while for the full slate Q-learning agent (DQN agent), it amounted to 154.9.

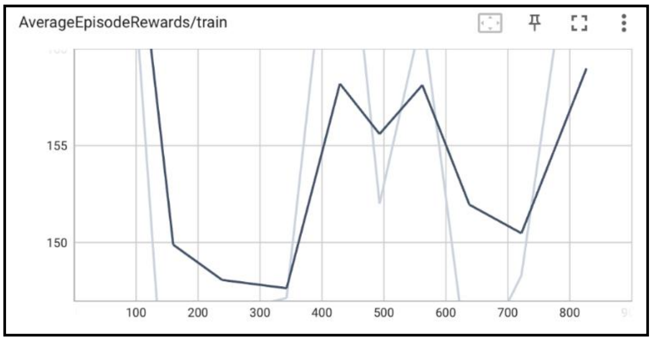

Figure 1513: Average episodic reward with full SlateQ

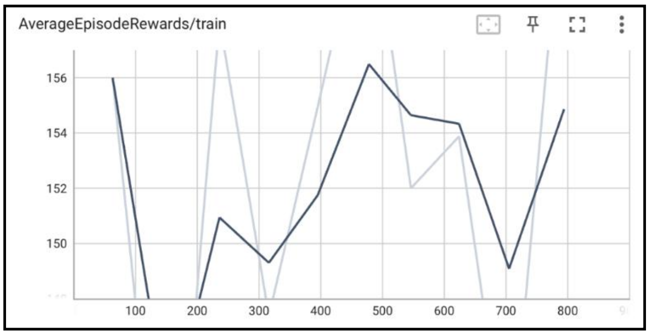

Figure 1614: Average episodic reward with the proposed agent

VI. CONCLUSION

In the ever-evolving landscape of the IT industry, marked by the proliferation of remote work arrangements and the relentless demands of the digital age, the well-being of IT professionals has emerged as an increasingly critical concern. Prolonged computer screen work, often necessitated by the demands of the job, has cast a shadow on the work-life balance of individuals in this sector. Stress, an all-too-common companion in the lives of IT professionals, is a complex issue stemming from various sources, including the isolation associated with remote work and the weighty responsibilities of their roles.

This comprehensive research project has sought to address the pressing challenge of detecting and mitigating stress levels and emotions in IT professionals without disrupting their daily work routines. To tackle this formidable challenge, the project was divided into four distinct components, each aligned with a specific sub-research problem:

- Keystroke Dynamics-based Stress Detection: This component delved into the analysis of keystroke dynamics to identify stress levels, offering a non-intrusive means of stress detection that seamlessly integrates into users' everyday work. It leveraged advanced machine learning techniques, including deep neural networks, to precisely capture subtle changes in typing patterns associated with stress.
- Heart Rate Variability-based Stress Detection: The second component harnessed external mouse technology to analyze heart rate variability, providing an additional layer of insight into users' stress levels. By integrating the ESP32 microcontroller and MAX30102 heart rate sensor, it achieved real-time data collection and processing, offering precise stress assessments.
- Facial Dynamics-based Emotion Detection: Component three leveraged webcam technology to examine facial dynamics, allowing for the detection of users' emotions and further enriching the understanding of their mental states. Convolutional Neural Networks (CNNs) and Vision Transformer (ViT) models were employed for facial feature analysis, demonstrating the project's technical sophistication.
- Reinforcement Learning-based Recommendation System: The fourth component synthesized the results from the previous three components to generate personalized recommendations for stress relief, offering users actionable steps to alleviate stress and enhance well-being. Deep Reinforcement Learning (DRL) techniques were used to optimize user engagement and satisfaction, creating a dynamic recommendation system.

These components collectively constitute an innovative platform, seamlessly integrated into a user-friendly desktop application, designed to assist IT professionals in managing their stress and emotions while preserving their productivity and workflow.

In conclusion, this research not only addresses the immediate challenges faced by IT professionals but also highlights the potential of technology-driven solutions to enhance mental and emotional well-being in a rapidly evolving work environment. As the IT industry continues to evolve, the insights and innovations generated by this project offer a promising path toward a healthier and more resilient workforce in the modern workplace. This technical sophistication combined with user-centric design positions this

ISSN (online): 2581-3048
Volume 7, Issue 10, pp 609-626, October-2023
https://doi.org/10.47001/IRJIET/2023.710081

research at the forefront of stress detection and management in the IT sector.

Furthermore, our Stress Monitoring and Relieving Application has broader implications beyond the IT industry. Stress is a pervasive issue in today's fast-paced world, affecting individuals across various professions and lifestyles. The multidimensional approach we have developed, combining physiological, behavioral, and machine learning-based components, can serve as a template for stress management applications in diverse settings. From healthcare professionals facing high-pressure situations to students coping with academic stress, the adaptability and effectiveness of our system can potentially benefit individuals from all walks of life.

Finally, our research represents a significant step forward in the development of technology-driven solutions for stress management and well-being enhancement. By integrating cutting-edge sensors, data analysis techniques, and reinforcement learning algorithms, we have created a holistic approach to stress monitoring and relief. The potential impact of this application is far-reaching, with the capacity to improve the quality of life for countless individuals, and contribute to a more resilient and productive workforce. As we continue to refine and expand upon this research, we anticipate even greater strides in the field of stress management and mental health support.

REFERENCES

[1] S. Panicker and P. Gayathri, " A survey of machine learning techniques in physiology based mental stress detection systems.," Biocybernetics and Biomedical Engineering, pp. 444-469, 2019.

[2] A. Rezvani and P. Khosravi, "Emotional intelligence: The key to mitigating stress and fostering trust among software developers working on information system projects," International Journal of Information Management, pp. 139-150, 2019.

[3] S. Gedam and S. Paul, "A review on mental stress detection using wearable sensors and machine learning techniques," IEEE Access, no. 9, pp. 84045-84066, 2021.

[4] D. Arasu, A. AzlanMohamed, N. Ruhaiyem, N. Annamalai, S. Lutfi and M. Qudah, "Human Stress Recognition from Facial Thermal-Based Signature: A Literature Survey," CMES, 2022.

[5] M. Alva, M. Nachamai and J. Paulose, "A comprehensive survey on features and methods for speech emotion detection," ICECCT, pp. 1-6, 2015.

[6] S. Greene, H. Thapliyal and A. Caban, "A survey of affective computing for stress detection: Evaluating technologies in stress detection for better health," EEE Consumer Electronics Magazine, vol. 5(4), pp. 44-56, 2016.

[7] S. Mittal, S. Mahendra, V. Sanap and P. Churi, "How can machine learning be used in stress management: A systematic literature review of applications in workplaces and education," International Journal of Information Management Data Insights, no. 100110, 2022.

[8] P. J. Jayaraj, M. Ghazali and A. Gaber, "Relax App: Designing Mobile Brain-Computer Interface App to Reduce Stress among Students," International Journal of Innovative Computing, 2021.

[9] S. Kalansooriya, A. Kaluarachchi, C. Weerawickrama, D. Nanayakkara, D. Kasthuriarachchi and D. Adeepa, "“xīnlǐ” The Social Media App to Replenish Mental Health with the Aid of an Egocentric Network," in R10-HTC, 2022.

[10] M. Afsar, T. Crump and B. Far, "Reinforcement learning based recommender systems: A survey," ACM Computing Surveys, pp. 1-38, 2022.

[11] Y. Can, N. Chalabianloo and C. Ersoy, "Continuous stress detection using wearable sensors in real life: Algorithmic programming contest case study," Sensors, 2019.

[12] N. C. D. E. C. E. Yekta Said Can, "Continuous Stress Detection Using Wearable Sensors in Real Life: Algorithmic Programming Contest Case Study," National Library of Medicine, 2019.

[13] Y. Can, H. Iles-Smith, N. Chalabianloo, D. Ekiz, J. Fernández-Álvarez, C. Repetto and C. Ersoy, "How to relax in stressful situations: a smart stress reduction system," Healthcare, vol. 8, no. 2, p. 100, 2020.

[14] C. Weerasinghe, "Stress Detection by Keystroke,App & Mouse Changes," [Online]. Available: https://www.kaggle.com/datasets/chaminduweerasinghe/stress-detection-by-keystrokeapp-mouse-changes.

[15] S. Liang and R. Srikant, "Why deep neural networks for function approximation," arXiv preprint, 2016.

[16] E. Ie, V. Jain, J. Wang, S. Narvekar, R. Agarwal, R. Wu and C. Boutilier, "SlateQ: A tractable decomposition for reinforcement learning with recommendation sets," 2019.

[17] C. Liu, X. Xu and D. Hu, "Multiobjective Reinforcement Learning: A Comprehensive Overview," in IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2015.

[18] "3 types of stress and what you can do to fight them," [Online]. Available: https://www.betterup.com/blog/types-of-stress. .

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 609-626, October-2023

https://doi.org/10.47001/IRJIET/2023.710081

[19] J. Hernandez, P. P. A. R. and M. C. , "Under Pressure: Sensing Stress of Computer Users," Association of Computing Machinery, p. 10, 2014.

[20] S.-H. LAU, "STRESS DETECTION FOR KEYSTROKE DYNAMICS," p. 232, 2018.

[21] A. A. M. S. Y. M. Lim, "Detecting Emotional Stress during Typing Task with," p. 10, 2014.

[22] Z.-L. L. W.-J. Y. J. D. H.-R. LV, "EMOTION RECOGNITION BASED ON PRESSURE SENSOR KEYBOARDS,," 2008.

[23] L. M. Vizer, "Detecting Cognitive and Physical Stress Through Typing Behavior," 2009.

[24] Paul H. Dietz, B. E. Jonathan Westhues and S. B. , "A Practical Pressure Sensitive Computer Keyboard," UIST, 2009.

[25] T. Yamauchi and K. X. , "Reading Emotion From Mouse Cursor Motions: Affective Computing Approach," Cognitive Science, 2017.

[26] H. Lv and W.-Y. W. , "Biologic verification based on pressure sensor keyboards and classifier fusion techniques," IEEE, 2006.

[27] D. Gunetti and C. P. , "Keystroke analysis of free text," ACM, 2005.

[28] L. Hai-Rong, L. Z.-L. Y. W.-J. and J. D. , "Emotion recognition based on pressure sensor keyboards," IEEE, 2008.

[29] G. Hytry, "IT's Stressful. Ask DevOps, They'll Know: Stress in the IT Sector [2022 Survey]," 2022. [Online]. Available: https://spacelift.io/blog/are-it-jobs-stressful.

[30] J. Fan, Z. Wang, Y. Xie and Z. Yang, "A theoretical analysis of deep Q-learning," Learning for dynamics and control, pp. 486-489.

[31] Ie, E., Hsu, C. W., Mladenov, M., Jain, V., Narvekar, S., Wang, J., ... & Boutilier, C. (2019). Recsim: A configurable simulation platform for recommender systems. arXiv preprint arXiv:1909.04847.

AUTHOR'S BIOGRAPHIES

Dulshan Perera

An undergraduate student specializing in Software Engineering at the Sri Lanka Institute of Information Technology in Colombo, Sri Lanka.

Amantha Jayathilake

An undergraduate student specializing in Software Engineering at the Sri Lanka Institute of Information Technology in Colombo, Sri Lanka.

Janudi Ranasinghe

An undergraduate student specializing in Data Science at the Sri Lanka Institute of Information Technology in Colombo, Sri Lanka.

Shehan Bartholomeusz

An undergraduate student specializing in Software Engineering at the Sri Lanka Institute of Information Technology in Colombo, Sri Lanka.

Samadhi Rathnayake

A lecturerin the Department of Information Technology at the Sri Lanka Institute of Information Technology in Colombo, Sri Lanka.

Samitha Vidhanaarachchi

A lecturerin the Department of Computer Science and Software Engineeringat the Sri Lanka Institute of Information Technology in Colombo, Sri Lanka.

Thilanga Jayarathne

The visionary Founder of Xinotech Technology Services, driving innovation and excellence in the tech industry.

International Research Journal of Innovations in Engineering and Technology(IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 609-626, October-2023

https://doi.org/10.47001/IRJIET/2023.710081

Arosha Dasanayaka

A compassionate Counseling Psychologist at Mind Heals, dedicated to supporting mental well-being.

Citation of this Article:

M. S. D. Perera, S. M. D. A. R Jayathilake, J. D. Ranasinghe, S. V. Bartholomeusz, H. M. Samadhi Chathuranga Samitha Vidhanaarachchi, Thilanga Jayarathne, Arosha Dasanayaka, "Stress Monitoring and Relieving Application for IT Professionals" Published in *International Research Journal of Innovations in Engineering and Technology - IRJIET*, Volume 7, Issue 10, pp 609-626, October 2023. Article DOI https://doi.org/10.47001/IRJIET/2023.710081

****