PAPER • **OPEN ACCESS**

Stress Detection using Machine Learning and Deep Learning

To cite this article: Z. Zainudin *et al* 2021 *J. Phys.: Conf. Ser.* **1997** 012019

View the article online for updates and enhancements.

Journal of Physics: Conference Series **1997** (2021) 012019 doi:10.1088/1742-6596/1997/1/012019

Stress Detection using Machine Learning and Deep Learning

Z. Zainudin[1], S. Hasan[1], S.M. Shamsuddin[1] and S. Argawal[2]

[1]School of Computing, Faculty Engineering, Universiti Teknologi Malaysia, Malaysia
[2]Information Technology Department, Indian Institute of Information Technology, Allahabad Prayagraj, India

zanariah86@gmail.com, shafaatunnur@utm.my

Abstract. Stress is a normal phenomenon in today's world, and it causes people to respond to a variety of factors, resulting in physiological and behavioural changes. If we keep stress in our minds for too long, it will have an effect on our bodies. Many health conditions associated with stress can be avoided if stress is detected sooner. When a person is stressed, a pattern can be detected using various bio-signals such as thermal, electrical, impedance, acoustic, optical, and so on, and stress levels can be identified using these bio-signals. This paper uses a dataset that was obtained using an Internet of Things (IOT) sensor, which led to the collection of information about a real-life situation involving a person's mental health. To obtain a pattern for stress detection, data from sensors such as the Galvanic Skin Response Sensor (GSR) and the Electrocardiogram (ECG) were collected. The dataset will then be categorised using Multilayer Perceptron (MLP), Decision Tree (DT), K-Nearest Neighbour (KNN), Support Vector Machine (SVM), and Deep Learning algorithms (DL). Accuracy, precision, recall, and F1-Score are used to assess the data's performance. Finally, Decision Tree (DT) had the best performance where DT have accuracy 95%, precision 96%, recall 96% and F1-score 96% among all machine learning classifiers.

1. Introduction

Stress is something that concerns our lives. There are many variables in our day-to-day life that are tension. Human environments, like worksite, home, or society, may somehow inflict stress on a person. According to Palmer [1], "Stress is defined as a complex psychological and behavioural condition when the person's demands are imbalanced and the way demands are met."

Also, the American Institute of Stress found that 80% of workers experience stress in their everyday work and need support in managing stress. Based on Ahuja and Banga [2], study recorded major suicide cases among students aged 15-29 due to stress. There are 8934 cases recorded in 2015, and study was inspired to identify stress in early stages. These figures and stress effects on people, which has been the leading cause of many diseases like hypertension, sleep deprivation, and others. Stress that cannot be adequately treated can lead to serious cases where one person committed suicide. This is vital to identify and control stress before it becomes severe. Many researchers investigate stress detection in many fields. This paper will elaborate on stress identification based on five conditions using data obtained using IoT sensors. Early detection can help track tension, and different machine learning and deep learning approaches have been explored and compared.

2. Related Research Works

Many studies are being conducted to identify tension or depressed individuals. Table 1 shows some previous related research work focused on the stress detection scheme, where some researchers use the public dataset and some researchers collect their own dataset [3-12].

Journal of Physics: Conference Series **1997** (2021) 012019 doi:10.1088/1742-6596/1997/1/012019

Table 1. Previous Related Research Works.

Ref	Title	Dataset	Result
[3]	Stress Detection with Machine Learning and Deep Learning using Multimodal Physiological Data	Public dataset WESAD dataset	Achieved accuracy 84.32% and 95.21% using RF, DT, AdaBoost, KNN, LDA, SVM and DL
[4]	Stress Detection through Speech Analysis using Machine Learning	Public Dataset Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) dataset	CNN- Achieved accuracy 94.26%-94.3%
[5]	Introducing WESAD, a Multimodal Dataset for Wearable Stress and Affect Detection	Public dataset WESAD dataset	Accuracy of 80% (three class) and 93% (two class) was achieved using RF, DT, AdaBoost, KNN, LDA, and SVM
[6]	A Machine Learning Approach for Stress Detection using a Wireless Physical Activity Tracker	Private Dataset Collected own dataset using FITBIT device and analysis using ANOVA	AIC- 782.8842 (Logit model) AIC- 781.6256 (Probit model) AIC-786.8999 (Complementary Log-Log model) *lower AIC, the better of model
[7]	Machine Learning and IoT for prediction and detection of stress	Private Dataset Collected own dataset and classified using Python	LR-66% SVM-68%
[8]	Machine Learning-based signal processing using physiological signals for stress detection.	Private Dataset Collected own dataset based on heart rate, EMG, GSR hand and foot data, respiration and classified using WEKA	KNN classifier- Achieved accuracy 92.06% SVM- Achieved accuracy 96.82%
[9]	Stress detection using wearable physiological sensors	Private Dataset Collected own dataset from BN-PPGED	SVM- Achieved accuracy 82%
[10]	Emotion Recognition Based on Multichannel Physiological Signals with Comprehensive Nonlinear Processing	Private Dataset Collected own dataset based on the ECG,GSR,EMG	KPCA reduce the features and GBDT for classifier- Achieved accuracy 93.42%
[11]	Emotion Recognition by Heart Rate Variability	Public Dataset MAHNOB dataset	SVM- Achieved accuracy 48.5%
[12]	Classification of Physiological Signals for Emotions Recognition using IOT	Private Dataset-SAID Dataset Collected own dataset using ECG and GSR SAID Dataset	ANN- Achieved Mean accuracy 75.8% and standard deviation of accuracy 11.38%

Abbreviations
RF=Random Forest, SVM=Support Vector Machine, KNN= k-Nearest Neighbour, DT=Decision Tree, AdaBoost =Adaptive Boosting, LDA= Linear Discriminant Analysis, DL=Deep Learning, LR= Logistic Regression, CNN=Convolutional Neural Network, AIC= Akaike information criterion ANN=Artificial Neural Network, KPCA=Kernel Principal Component Analysis, GBDT=Gradient Boosting Decision Tree ()

3. Research Methodology

The research methodology used to conduct the analysis for this paper is detailed below. The paper's primary contribution is the identification of stress using machine learning and deep learning. The flow diagram below illustrates the proposed work on stress detection using machine learning and deep learning. This can be summarised in five steps.

Figure 1. Flow diagram of stress detection.

3.1. Dataset preparations

There are three method to collect data such as interview/questionnaire, sensor measuring method and collection of social media. This paper used dataset comes sensors were it been collected at Indian Institute of Information Technology (IITA). The dataset was gathered using a sensor included in the MySignals Healthcare Toolkit. MySignals is a forum for medical device and e-Health application creation. The MySignals toolkit includes an Arduino Uno board and a variety of sensor ports. The sensors were attached to the MySignals Hardware package (which includes an Arduino) and programmed using the Arduino SDK.

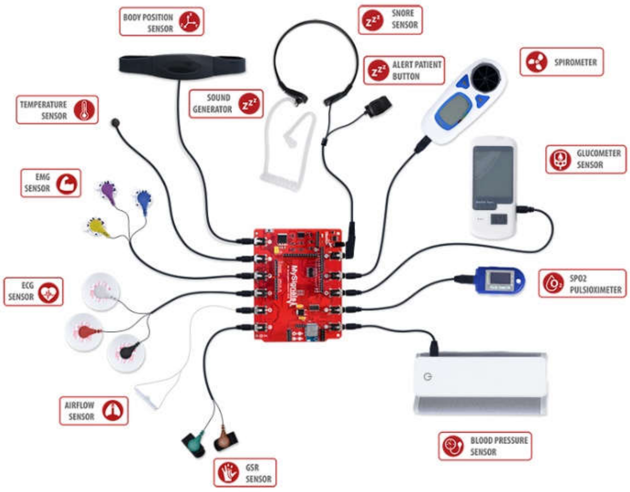

Figure 2. MySignals toolkit.

3.2. Dataset Acquisitions

The Galvanic Skin Response (GSR) and Electrocardiogram (ECG) sensors were used to collect data from 252 participants, a combination of male and female students ranging in age from 20 to 22 years. The tests took place in closed and quit locations. Each participant is required to watch 18 videos from a list of YouTube videos ranging in length from 2 to 5 minutes. Throughout the video playback, MYSignal toolkits were used to record Galvanic Skin Response (GSR) and Electrocardiogram (ECG) sensors. Following that, participants were given a response form to complete about their emotional rate at the end of each video session. After pre-processing the raw data from MYSignal, the Mean, Median, Standard deviation, Minimum reading, Maximum reading, Max Ratio, and Min Ratio are extracted to obtain the best features. Next after data collection and pre-processing, the processed data was analysed using a machine learning classifier to predict the users' mental states and to comprehend their physiological characteristics under various conditions. This dataset is referred to as the Stress Analysis using IOT Device Data Set (SAID).

Figure 3. Experiments Setup for each participant.

3.3. *Classification Algorithm*

The classification algorithms is method to detect stress level in SAID dataset which is been categorized into four classes as 0, 1, 2 and 3 as 'Relax', 'Stressed', 'Partially Stressed' and 'Happy' respectively as illustrated in Figure 4.

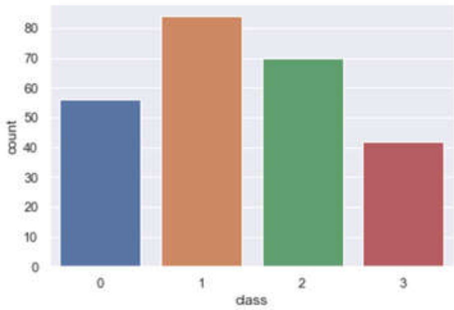

Figure 4. Count of each class for SAID dataset.

Classification algorithm been used in this paper are Multilayer Perceptron (MLP), Decision Tree (DT), K- Nearest Neighbour (KNN), Support Vector Machine (SVM) and Deep Learning (DL). The dataset is split into two parts: 70% for training and 30% for research. The following subsection will discuss the machine learning algorithms used in this experiment and how their parameters were set for each classifier.

3.3.1. Multilayer Perceptron (MLP). MLP is the popular and mostly used in most research area. MLP has input and will be transmitted inside MLP layer called as hidden layer in one direction to be classified as output. There are no loops, thus it will not affect the output of each neurons. The parameter setting for MLP is summarized below using Table 2.

Table 2. MLP Parameter Setting.

Parameter	Parameter Setting
hidden_layer_sizes	100
activation	relu
learning_rate_init	0.001
momentum	0.9
solver	adam

3.3.2. Decision Tree (DT). Decision tree builds classification models in the form of a tree structure. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes based on their class dataset [3]. The parameter setting for DT is summarized below using Table 3.

Table 3. Decision Tree Parameter Setting.

Parameter	Parameter Setting
criterion	gini
min_samples_leaf	1
min_samples_split	2

3.3.3. K- Nearest Neighbour (KNN). KNN is a class membership where it will group the dataset based on their classes whether it belong to group a or group b. KNN works by allocated data based on the nearest neighbours which one is its k closest neighbours (k is a positive number and a small number). If k = 1, then the data will be allotted to the group a or b based on closest neighbour [2]. Table 4 shows parameter setting for KNN.

Table 4. K- Nearest Neighbour Parameter Setting.

Parameter	Parameter Setting
n_neighbor	5
weights	uniform
leaf_size	30
metric	minkowsk

3.3.4. Support Vector Machine (SVM). SVM works upon the ideal hyper plane and still effective in high dimensional spaces. In 2-Dimensional data, SVM will try to classify based on dataset classes [2]. Table 5 shows parameter setting for SVM.

Table 5. SVM Parameter Setting.

Parameter	Parameter Setting
C	1.0
kernel	rbf
degree	3
decision_function_shape	one-vs-rest ('ovr')

3.3.5. Deep Learning (DL). Deep learning has many layers of the processing units for the input of the dataset. Each layer has massive sub-layers of hidden layers. This algorithm is not only for supervised but also applicable for unsupervised classification problem [3]. The deep learning layer architecture are show in Figure 5 below:

```
Layer (type)                 Output Shape              Param #
=================================================================
dense_402 (Dense)            (None, 14)                210
_________________________________________________________________
dropout_58 (Dropout)         (None, 14)                0
_________________________________________________________________
dense_403 (Dense)            (None, 100)               1500
```

```
_________________________________________________________________
dropout_59 (Dropout)         (None, 100)               0
_________________________________________________________________
dense_404 (Dense)            (None, 4)                 404
=================================================================
Total params: 2,114
Trainable params: 2,114
Non-trainable params: 0
_________________________________________________________________
```

Figure 5. Deep learning layer architecture.

3.4. Performance Evaluation

Several performance evaluation metrics are identified to be used to evaluate the performance of the stress detection model [13]. These metrics are accuracy, precision, recall and F1-Score are shown below:

$$Accuracy = \frac{True\ Positive+True\ Negative}{True\ Positive+True\ Negative+False\ Positive+False\ Negative} \tag{1}$$

$$Precision = \frac{True\ Positive}{True\ Positive+False\ Positive} \tag{2}$$

$$Recall = \frac{True\ Positive}{True\ Positive+False\ Positive} \tag{3}$$

$$F1\text{-}Scorel = 2x\ \frac{Precision*Recall}{TPrecision+Recall} \tag{4}$$

4. Experimental results and Discussions

The main goal of this paper is to detect stress level in SAID dataset. The dataset has been splitted into training dataset contain 70% and testing dataset contain 30% as shown in Figure 6. In this experiments Multilayer Perceptron (MLP), Decision Tree (DT), K- Nearest Neighbour (KNN), Support Vector Machine (SVM) and Deep Learning (DL) are been used and been detailed up in Table 2. Table 6 also shows that the accuracy has reached up 79% until 96% for SAID dataset. Based on the result from Table 6 below, Support Vector Machine (SVM) had the overall worst performance where SVM have accuracy 79%, precision 81%, recall 75% and F1-score 77%, whereas Decision Tree (DT) had the best performance where DT have accuracy 95%, precision 96%, recall 96% and F1-score 96% among all machine learning classifiers.

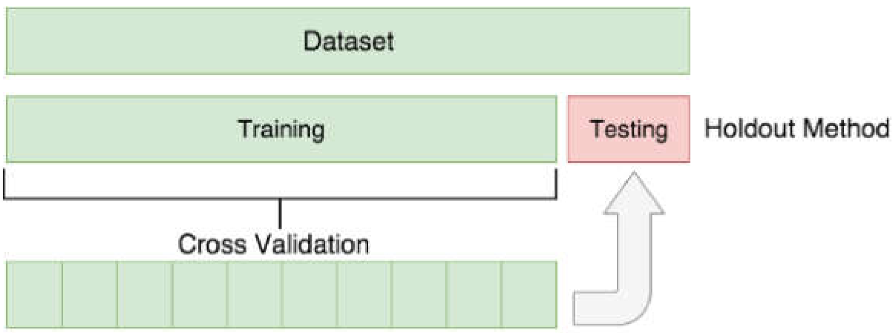

Figure 6. Training/Testing and Cross Validation use in these experiments to overcome overfitting problem.

Table 6. Experimental Result.

Classifier	Accuracy	Precision	Recall	F1-Score
Support Vector Machine (SVM)	79%	81%	75%	77%
K- Nearest Neighbour (KNN)	82%	79%	78%	78%
Multilayer Perceptron (MLP)	86%	84%	87%	85%
Deep Learning (DL)	91%	92%	91%	91%
Decision Tree (DT)	95%	96%	96%	96%

In can be concluded that using Decision Tree give the best result compared to other machine learning techniques. Decision Tree have several advantages such as the output are easy to read and assign specific values to each problem, decision path and the outcome of the output. Based on the learning curve it also show that using Decision Tree are suitable for this dataset where there is no underfitting and overfitting cases when training the model using Decision Tree algorithm. From the previous study, Tiwari et.al, 2019 [12], using Artificial Neural Network as the classifier and getting result Mean of Accuracy 73.58% and Standard Deviation of ANN model accuracy is 11.38%. From this, we can make a conclusion that using other machine learning from our experiments are better compared to previous studies. By choose a better classifier could improve the efficiency when training the model.

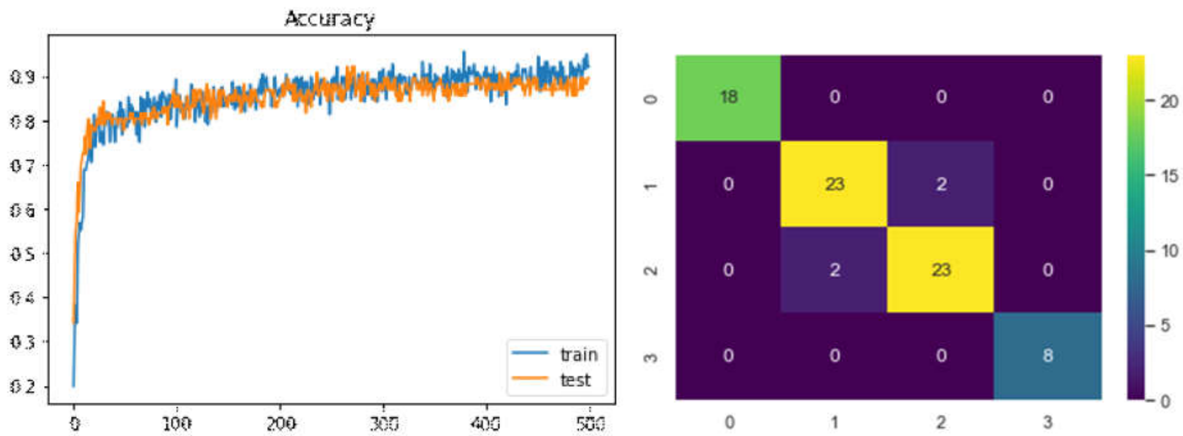

Figure 7. Learning curve for training (Cross Validation=10) and confusion matrix using Decision Tree.

5. Conclusion

In this paper, it can conclude that using suitable classifier will get better result in accuracy, precision, recall and F1-Score. From experiments results, it shows some significant value where DT achieved the best resulting on accuracy 95%, precision 96%, recall 96% and F1-score 96%. The results prove that the using DT has a competitive performance compared to the others classifiers for detecting stress and non-stress and classifying stress levels. Further work can be done by using more classifier and applied 10-fold cross validation.

Acknowledgment

This work is supported by Ministry of Education (MOE), Malaysia, Universiti Teknologi Malaysia (UTM), Malaysia and “ASEAN- India Science & Technology Development Fund (AISTDF)”, SERB, Sanction letter no. – IMRC/AISTDF/R&D/P-6/2017. This paper is financially supported by University Grant No. 04G48. The authors would like to express their deepest gratitude to all researchers in IITA Allahabad for their support in providing the SAID datasets to ensure the success of this research, as well as School of Computing, Faculty of Engineering for their continuous support in making this research possible.

Journal of Physics: Conference Series **1997** (2021) 012019 doi:10.1088/1742-6596/1997/1/012019

References

[1] Palmer S 1989 *The Health and Safety Practitioner*, **8** 16-18
[2] Ahuja R and Banga A 2019 *International Conference on Pervasive Computing Advances and Applications – PerCAA 2019* **125** 349-353
[3] Bobade P and Vani M 2020 *Proceedings of the Second International Conference on Inventive Research in Computing Applications (ICIRCA-2020)* 51-58
[4] Vaikole S Mulajkar S More A Jayaswal P and Dhas S 2020 *International Journal of Creative Research Thoughts (IJCRT)* **8** 5 2239-44
[5] Schmidt P Reiss A Duerichen R Marberger C and Laerhoven K V 2018 *Proceedings of the 20th ACM International Conference on Multimodal Interaction* 400-408
[6] Padmaja B Rama Prasad V V and Sunitha K V N 2018 *International Journal of Machine Learning and Computing* **8** 1 33-38
[7] Pandey P S 2017 *2017 17th International Conference on Computational Science and Its Applications (ICCSA)*
[8] Ghaderi A Frounchi J and Farnam A 2015 *22nd Iranian Conference on Biomedical Engineering (ICBME)* 93-98
[9] Sandulescu V Andrews S Ellis D Bellotto N and Mozos O M 2015 *International Work-Conference on the Interplay Between Natural and Artificial Computation* 526-532
[10] Zhang X Xu C Xue W Hu J He Y and Gao M 2018 *Sensors 2018* **18** 11 3886
[11] Ferdinando H Ye L Seppänen T and Alasaarela E 2014 *Australian Journal of Basic and Applied Sciences*. **8** 14 50-55
[12] Tiwari S Agarwal S Muhammad Syafrullah and Adiyarta K 2019 *Proc. EECSI 2019 - Bandung, Indonesia* **6** 107-111
[13] Zainudin Z Shamsuddin S M and Hasan S 2019 *The International Conference on Advanced Machine Learning Technologies and Applications (AMLTA2019)* **921** 43-51