Philippine Copyright 2025
By Johana Benolirao, Christaline Calubayan, Larry Miguel Cueva, Deseree Quiray and the
College of Computer and Information Sciences
Polytechnic University of the Philippines

All rights reserved. Portions of this manuscript may be reproduced with proper referencing and due acknowledgements of the authors.

APPLICATION OF A HYBRIDIZED LSTM-SVM IN THE DETECTION OF ARTIFACTS IN ELECTRODERMAL ACTIVITY SIGNALS FOR STRESS DETECTION

A Thesis

Presented to the Faculty of the

College of Computer and Information Science

Polytechnic University of the Philippines

Sta. Mesa, Manila

In Partial Fulfillment of the Requirements for the Degree

Bachelor of Science in Computer Science

Benolirao, Johana G.

Calubayan, Christaline B.

Cueva, Larry Miguel R.

Quiray, Deseree O.

December 2024

APPROVAL SHEET

The thesis **"Application of a Hybridized LSTM-SVM in the Detection of Artifacts in Electrodermal Activity Signals for Stress Detection"** submitted and presented by **Benolirao, Johana G., Calubayan, Christaline B., Cueva, Larry Miguel R., and Quiray, Deseree O.,** in partial fulfilment of the requirements for the degree of Bachelor of Science in Computer Science has been

Recommended for Approval and Acceptance: Date:

_____ _____

Montaigne G. Molejon, MSIT
Faculty-In-charge

Approved by the Thesis Defense Panel:

_____ _____

Aleta C. Fabregas, DIT

_____ _____

Montaigne G. Molejon, MSIT

_____ _____

Ma. Esperanza H. Reyes, MGM

Accepted for the Department of Computer Science

_____ _____

Montaigne G. Molejon, MSIT
Department Chairperson

Accepted for the College of Computer and Information Sciences

_____ _____

Melvin C. Roxas, MSGITS
Dean of the College

ACKNOWLEDGEMENTS

We wish to thank those who contributed to the success of this research work: First and foremost, we express our deepest gratitude and praise to Almighty God for his blessings, wisdom, and strength that guided us throughout the completion of this research.

We also express our gratitude to the Department of Science and Technology-Computing and Archiving Research Environment (DOST-COARE) for providing the necessary computing resources, which greatly contributed to the data processing and analysis of this research. Special thanks to our professors and panelists for their knowledge, encouragement, and assistance.

To our families and friends, thank you for unwavering support, love, and encouragement. Your motivation and belief in us have been a great source of strength throughout this journey.

Lastly, we acknowledge the hard work, dedication, and teamwork of each member of our research group. Without each other's efforts and contributions, the successful completion of this study would not have been possible.

With gratitude and humility, we thank everyone who has been a part of this journey.

Initials

J.G.B.
C.B.C.
L.M.R.C.
D.O.Q.

CERTIFICATION OF ORIGINALITY

This is to certify that the research work presented in this thesis, APPLICATION OF A HYBRIDIZED LSTM-SVM IN THE DETECTION OF ARTIFACTS IN ELECTRODERMAL ACTIVITY SIGNALS FOR STRESS DETECTION for the degree Bachelor of Science in Computer Science at the Polytechnic University of the Philippines embodies the result of original and scholarly work carried out by the undersigned. This thesis does not contain words or ideas taken from the published sources or written works that have been accepted as the basis for the award of the degree from any other higher education institution, except where proper referencing and acknowledgement of work were made.

Johana G. Benolirao
Researcher

Christaline B. Calubayan
Researcher

Larry Miguel R. Cueva
Researcher

Deseree O. Quiray
Researcher

_____
Date Signed

ABSTRACT

Title	:	Application of Hybridized LSTM-SVM in the Detection of Artifacts in Electrodermal Activity Signals for Stress Detection
Researchers	:	Cueva, Larry Miguel R., Benolirao, Johana G., Calubayan, Christaline B., Quiray, Deseree O.
Degree	:	Bachelor of Science in Computer Science
Institution	:	Polytechnic University of the Philippines - Mabini Campus
Year	:	2024
Adviser	:	Montaigne G. Molejon

Stress is a condition that can cause a severe influence on individuals, which may affect their physical and psychological health. In order to address this, utilization of wearable sensors to measure stress levels has been a trend, one example is the usage of electrodermal activity (EDA) signals to identify psychophysiological reactions of the body. However, artifacts or "noises", usually obtained along with genuine reaction during data gathering, frequently interfere in proper stress detection. This work introduces hybridized Long Short Term Memory (LSTM) and Support Vector Machine (SVM) in order to detect artifacts and properly treat it before using the signal for detecting stress levels. The researchers used a hybridized LSTM-SVM model to overcome limitations in traditional machine learning techniques for time series data, particularly EDA signals. By integrating both LSTM (for high-level features) and SVM (for traditional lower-level features), the model is expected to outperform both deep learning models, which typically use only high-level features, and conventional ML models, which rely solely on low-level features. The study replicated previous studies' models to be compared to the LSTM-SVM models, which are Logistic Regression, Random Forest, Gradient Boosted Tree, Support Vector Machine, and LSMT-CNN based on significant performance measures such as accuracy, ROC-AUC, precision, recall, and F1-Score. The model has

an accuracy of 0.8953, ROC-AUC of 0.7836, precision of 0.8867, recall of 0.8953, and F1-score of 0.8840. While competitive, it has room for development compared to modern techniques such as LSTM-CNN. Optimization of hyperparameters, testing on various datasets, enabling real-time applications, and investigating adaptive learning are among the recommendations. The model shows promise for stress detection and artifact removal in wearable devices, potentially leading to advances in biological signal processing.

Keywords: Electrodermal activity signals, artifact detection, machine learning, LSTM-SVM, signal processing, stress detection

TABLE OF CONTENTS

		Page
Title Page		i
Certification and Approval Sheet		ii
Acknowledgments		iii
Certification of Originality		iv
Abstract		v
Table of Contents		vii
List of Tables		ix
List of Figures		x
1	**The Problem and Its Setting**	
	Introduction	1
	Theoretical Framework	9
	Conceptual Framework	11
	Statement of the Problem	12
	Null Hypothesis	13
	Scope and Delimitation	13
	Significance of the Study	13
	Definition of Terms	14
2	**Review of Literature and Studies**	
	Electrodermal Activity Artifact Detection Using Machine Learning Techniques	18
	Limitations of Machine Learning and Deep Learning Techniques in EDA Artifact Detection	21
	Advantages of LSTM	26
	Hybridizing LSTM and SVM	30
	Stress Detection Using Electrodermal Activity Signals	42
	Synthesis of the Study	50
3	**Methodology**	
	Research Design	52
	Sources of Data	52
	Research Instrument	54
	System Architecture	56
	Procedures	58
	Ethical Considerations	59

Data Analysis 60

4 **Results and Discussion**
Optimal Gamma and C Hyperparameters 65
Performance of C and Gamma Hyperparameter Tuning for LSTM-SVM 75
LSTM-SVM Model Compared to Other Existing Artifact Detection Models 76
Gradient Boosted Tree (Hossain et. al., 2022) 76
Linear Regression (Hossain et. al., 2022) 95
Support Vector Machine (Hossain et. al., 2022) 101
Linear Regression (Taylor et. al., 2015) 111
Random Forest (Taylor et. al., 2015) 118
Support Vector Machine (Taylor et. al., 2015) 126
LSTM-CNN (Llanes-Jurado et. al., 2023) 138
Performance of the LSTM-SVM Model Compared to Other Existing Models 140

5 **Summary of Findings, Conclusions, and Recommendations**
Summary of Findings 143
Conclusions 145
Recommendations 147

6 **References** 149

7 **Appendices**
Appendix 1. Research Instruments 156
Appendix 2. Raw Data 158
Appendix 3. Correspondence 159
Appendix 4. Ethical Clearance 161
Appendix 5. Certification of Originality Check 162
Appendix 6. Certificate of Editing 163
Appendix 7. Biographical Statements 164
Appendix 8. Implementation Report 166
Appendix 9. Screen Layout and Description 170
Appendix 10. Thesis Revision matrixes 172

LIST OF FIGURES

Number	Title	Page
1	**Components of EDA: Phasic (Skin Conductance Response) and Tonic (Skin Conductance Level)**	3
2	**Skin conductance response relative to time**	4
3	**Example of Noisy and Clean EDA signal from the study of Hossain et. al. (2022)**	5
4	**Theoretical Framework**	9
5	**Conceptual Framework.**	11
6	**Some guidelines developed to aid human annotators in the data labeling process from the study of Gashi et. al. (2020)**	20
7	**Internal mechanism of LSTM cell**	29
8	**Full LSTM model**	29
9	**Cimino & Dell'Orletta (2016) hybridized LSTM-SVM architecture**	35
10	**Typical LSTM network**	37
11	**LSTM-XGBoost architecture**	38
12	**EDA Signal (SC) decomposition/deconvolution from Sanchez-Reolid et al. (2022)**	45
13	**Flowchart of feature extraction from EDA signals from Sanchez-Reolid et al. (2019)**	47
14	**System Architecture**	56
15	**System Workflow**	56
16	**EDA signal with detected artifacts (top left) along with smoothed and corrected signal**	142

LIST OF TABLES

Number	Title	Page
1	**Guidelines for Annotation of EDA Signals**	20
2	**Signal components in EDABE dataset**	53
3	**Confusion Matrix**	63
4	**Confusion Matrix with the Subjects.**	63
5	**Comparison of Models in terms of Accuracy, F1- Score, ROC-AUC, Precision and Recall**	64
6	**Confusion Matrix for C = 100 and Gamma = 0.1**	66
7	**Average Performance of the Test Subjects in C =100 and Gamma = 0.1**	67
8	**Confusion Matrix for C = 100 and Gamma = 0.5**	67
9	**Average Performance of the Test Subjects in C =100 and Gamma = 0.5**	68
10	**Confusion Matrix for C = 100 and Gamma = 1**	68
11	**Average Performance of the Test Subjects in C =100 and Gamma = 1**	69
12	**Confusion Matrix for C = 10 and Gamma = 0.1**	69
13	**Average Performance of the Test Subjects in C =10 and Gamma = 0.1**	70
14	**Confusion Matrix for C = 10 and Gamma = 0.5**	70
15	**Average Performance of the Test Subjects in C =10 and Gamma = 0.5**	71
16	**Confusion Matrix for C = 10 and Gamma = 1**	71
17	**Average Performance of the Test Subjects in C =10 and Gamma = 1**	72
18	**Confusion Matrix for C = 1 and Gamma = 0.1**	72
19	**Average Performance of the Test Subjects in C =1 and Gamma = 0.1**	73
20	**Confusion Matrix for C = 1 and Gamma = 0.5**	73
21	**Average Performance of the Test Subjects in C =1 and Gamma = 0.5**	74
22	**Confusion Matrix for C = 1 and Gamma = 1**	74
23	**Average Performance of the Test Subjects in C =1 and Gamma = 1**	75
24	**Performance of Every Model Compared to Tuned LSTM-SVM**	75
25	**Confusion Matrix of N Estimators = 600, Learning Rate = 0.01, and Max Depth = 5**	77
26	**Performance of N Estimators = 600, Learning Rate = 0.01, Max Depth = 5**	78
27	**Confusion Matrix of N Estimators = 600, Learning Rate = 0.1, and Max Depth = 5**	78
28	**Performance of N Estimators=600, Learning Rate=0.1, Max Depth=5)**	79

29 **Confusion Matrix of N Estimators = 600, Learning Rate = 0.1, and Max Depth = 3** 79
30 **Performance of N Estimators = 600, Learning Rate = 0.1, Max Depth= 3** 80
31 **Confusion Matrix of N Estimators = 600, Learning Rate = 0.01, and Max Depth = 3** 80
32 **Performance of N Estimators = 600, Learning Rate = 0.01, Max Depth = 3** 80
33 **Confusion Matrix of N Estimators = 600, Learning Rate = 0.01, and Max Depth = 10** 81
34 **Performance of N Estimators = 600, Learning Rate = 0.01, Max Depth = 10** 81
35 **Confusion Matrix of N Estimators = 600, Learning Rate = 0.1, and Max Depth = 10** 82
36 **Performance of N Estimators = 600, Learning Rate = 0.1, Max Depth = 10)** 82
37 **Confusion Matrix of N Estimators = 200, Learning Rate = 0.01, and Max Depth = 3** 83
38 **Performance of N Estimators = 200, Learning Rate = 0.01, Max Depth = 3** 83
39 **Confusion Matrix of N Estimators = 200, Learning Rate = 0.1, and Max Depth = 3** 84
40 **Performance of N Estimators = 200, Learning Rate = 0.1, Max Depth = 3** 85
41 **Confusion Matrix of N Estimators = 200, Learning Rate = 0.01, and Max Depth = 5** 85
42 **Performance of N Estimators = 200, Learning Rate = 0.01, Max Depth = 5** 86
43 **Confusion Matrix of N Estimators = 200, Learning Rate = 0.1, and Max Depth = 5** 86
44 **Performance of N Estimators = 200, Learning Rate = 0.1, Max Depth = 5** 87
45 **Confusion Matrix of N Estimators = 200, Learning Rate = 0.01, and Max Depth = 10** 87
46 **Performance of N Estimators = 200, Learning Rate = 0.01, Max Depth = 10** 88
47 **Confusion Matrix of N Estimators = 200, Learning Rate = 0.1, and Max Depth = 10** 88
48 **Performance of N Estimators = 200, Learning Rate = 0.1, Max Depth = 10** 88
49 **Confusion Matrix of N Estimators = 400, Learning Rate = 0.01, and Max Depth = 3** 89
50 **Performance N Estimators = 400, Learning Rate = 0.01, Max Depth = 3** 89

51	**Confusion Matrix of N Estimators = 400, Learning Rate = 0.1, and Max Depth = 3**	90
52	**Performance of N Estimators = 400, Learning Rate = 0.1, Max Depth = 3)**	90
53	**Confusion Matrix of N Estimators = 400, Learning Rate = 0.01, and Max Depth = 5**	91
54	**Performance of N Estimators = 400, Learning Rate = 0.01, Max Depth = 5**	91
55	**Confusion Matrix of N Estimators = 400, Learning Rate = 0.1, and Max Depth = 5**	92
56	**Performance of N Estimators = 400, Learning Rate = 0.1, Max Depth = 5**	93
57	**Confusion Matrix of N Estimators = 400, Learning Rate = 0.01, and Max Depth = 10**	93
58	**Performance of N Estimators = 400, Learning Rate = 0.01, Max Depth = 10**	94
59	**Confusion Matrix of N Estimators = 400, Learning Rate = 0.1, and Max Depth = 10**	94
60	**Performance of N Estimators = 400, Learning Rate = 0.1, Max Depth = 10**	94
61	**Confusion Matrix of C = 0.01**	96
62	**Performance of C = 0.01**	96
63	**Confusion Matrix of C = 0.1**	97
64	**Performance of C = 0.1**	97
65	**Confusion Matrix of C = 1**	98
66	**Performance ofnC = 1**	98
67	**Confusion Matrix of C = 10**	99
68	**Performance of C = 10**	99
69	**Confusion Matrix of C = 100**	100
70	**Performance of C = 100**	100
71	**Confusion Matrix of C = 1 and Gamma = 0.001**	101
72	**Performance of C =1 and Gamma = 0.001**	102
73	**Confusion Matrix of C = 1 and Gamma = 0.01**	102
74	**Performance of C =1 and Gamma = 0.01**	103
75	**Confusion Matrix of C = 1 and Gamma = 0.1**	103
76	**Performance of C =1 and Gamma = 0.1**	103
77	**Confusion Matrix of C = 1 and Gamma = 1**	104
78	**Performance of C =1 and Gamma = 1**	104
79	**Confusion Matrix of C = 10 and Gamma = 0.001**	105

80	**Performance of C =10 and Gamma = 0.001**	105
81	**Confusion Matrix of C = 10 and Gamma = 0.01**	106
82	**Performance of C = 10 and Gamma = 0.01**	107
83	**Confusion Matrix of C = 10 and Gamma = 0.1**	107
84	**Performance of C = 10 and Gamma = 0.1**	107
85	**Confusion Matrix of C = 100 and Gamma = 0.001**	108
86	**Performance of C = 100 and Gamma = 0.001**	108
87	**Confusion Matrix of C = 100 and Gamma = 0.01**	109
88	**Performance of C = 100 and Gamma = 0.01**	109
89	**Confusion Matrix of C = 1000 and Gamma = 0.001**	110
90	**Performance of C = 1000 and Gamma = 0.001**	111
91	**Confusion Matrix of C = 0.01**	112
92	**Performance of C = 0.01)**	112
93	**Confusion Matrix of C = 0.1**	113
94	**Performance of C = 0.1**	113
95	**Confusion Matrix of C = 1**	114
96	**Performance of C = 1**	114
97	**Confusion Matrix of C = 10**	115
98	**Performance of C = 10**	115
99	**Confusion Matrix of C = 100**	116
100	**Performance of C = 100**	116
101	**Confusion Matrix of N Estimators = 200, Max Depth = 10**	117
102	**Performance of N Estimators = 200, Max Depth = 10**	117
103	**Confusion Matrix of N Estimators = 200, Max Depth = 30**	118
104	**Performance of N Estimators = 200, Max Depth = 30**	118
105	**Confusion Matrix of N Estimators = 200, Max Depth = 50**	119
106	**Performance of N Estimators = 200, Max Depth = 50**	119
107	**Confusion Matrix of N Estimators = 400, Max Depth = 10**	120
108	**Performance of N Estimators = 400, Max Depth = 10**	120
109	**Confusion Matrix of N Estimators = 400, Max Depth = 30**	121
110	**Performance of N Estimators = 400, Max Depth = 30**	121
111	**Confusion Matrix of N Estimators = 400, Max Depth = 50**	122
112	**Performance of N Estimators = 400, Max Depth = 50**	122
113	**Confusion Matrix of N Estimators = 600, Max Depth = 10**	123

114	**Performance of N Estimators = 600, Max Depth = 10**	123
115	**Confusion Matrix of h N Estimators = 600, Max Depth = 30/)**	124
116	**Performance of N Estimators = 600, Max Depth = 30**	124
117	**Confusion Matrix of N Estimators = 600, Max Depth = 50)**	125
118	**Performance of N Estimators = 600, Max Depth = 50**	125
119	**Confusion Matrix of C = 1 and Gamma = 0.001**	126
120	**Performance of C = 1 and Gamma = 0.001**	127
121	**Confusion matrix of C = 1 and Gamma = 0.01**	127
122	**Performance of C = 1 and Gamma = 0.01**	128
123	**Confusion Matrix of C = 1 and Gamma = 0.1**	128
124	**Performance of C = 1 and Gamma = 0.1**	129
125	**Confusion Matrix of C = 1 and Gamma**	129
126	**Performance of C = 1 and Gamma = 1**	130
127	**Confusion Matrix of C = 10 and Gamma = 0.001**	130
128	**Performance of C = 10 and Gamma = 0.001**	131
129	**Confusion Matrix of C = 10 and Gamma = 0.01**	131
130	**Performance of C = 10 and Gamma = 0.01**	132
131	**Confusion Matrix of C = 100 and Gamma = 0.01**	132
132	**Performance of C = 100 and Gamma = 0.1**	133
133	**Confusion Matrix of C = 100 and Gamma = 0.001**	133
134	**Performance of C = 100 and Gamma = 0.001**	134
135	**Confusion Matrix of C = 100 and Gamma = 0.1**	134
136	**Performance of C = 100 and Gamma = 0.01**	135
137	**Confusion Matrix of C = 1000 and Gamma = 0.1 Configuration**	135
138	**Performance of C = 1000 and Gamma = 0.1**	136
139	**Confusion Matrix of C = 1000 and Gamma = 0.001**	136
140	**Performance of C = 1000 and Gamma = 0.001**	137
141	**Confusion Matrix of C = 1000 and Gamma = 0.1**	137
142	**Performance of C = 1000 and Gamma = 0.1**	138
143	**Confusion Matrix of Every Subject**	138
144	**Performance of the LSTM-CNN**	139
145	**Performance of the models during testing phase in accordance with ROC-AUC,Recall, Precision, F1-Score and Accuracy**	140

CHAPTER 1

THE PROBLEM AND ITS SETTING

Introduction

Stress is defined as a condition that involves worry or mental strain generated by a challenging circumstance (World Health Organization, 2022). Stress can be found in different aspects or events. In the country of Jordan during the COVID-19 pandemic, the healthcare workers revealed that they are experiencing a high level of stress and found out that 22.5% of participants had severe stress, 16.2% had extremely severe stress and 21.1% had moderate stress in a total of 60% of their healthcare workers experienced this level of stress (Alnazly et al., 2021). These high levels of stress experienced by healthcare workers during the pandemic can have negative impacts on their psychological well-being, job performance, and ability to provide quality patient care. With its negative connotations, numerous studies focused on detecting stress signals that may help in mitigating and managing stress to prevent and handle its effects better.

Coinciding with that, the utilization of wearable sensors to detect psychological and physiological responses has been a trend over the years. In the investigation of Dzedzickis et al., (2020), the researchers conducted a review of sensors and models utilized for human emotion identification. In a publication by Semmlow (2004), many physiological processes produce energy that can be detected directly by a device generally known as a transducer which converts energy from one form to another. The energy that is converted by the input transducer may be generated by the physiological processes of the body itself. The measurement of electrical activity in the heart, muscle, or brain, provides other examples of direct measurement of physiological energy. For these measurements, the energy is already electrical and only needs to be converted from an ionic to an electric current using an electrode. These include physiological processes that produce electrical energy, namely: electrocardiogram (ECG),

electroencephalogram (EEG), electromyogram (EMG), electrooculography (EOG), and electrodermal activity (EDA).

Among the techniques used in measuring physiological responses is Galvanic Skin Response (GSR), also known as Electrodermal Activity (EDA) or more specifically, Skin Conductance Response (SCR) measured in microsiemens (μS).

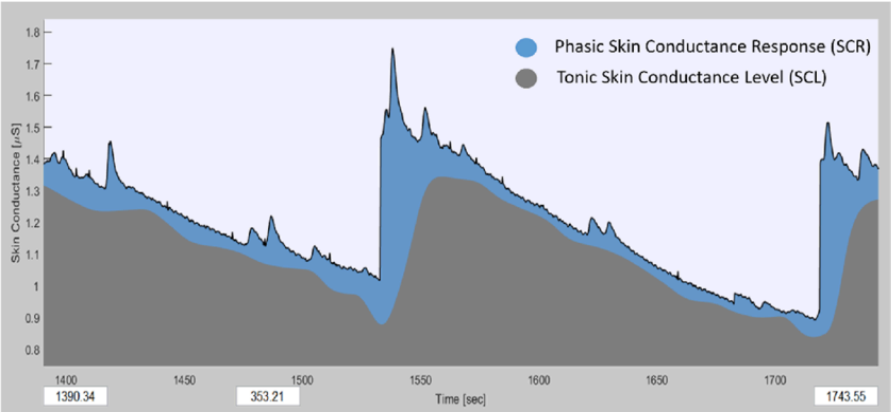
Electrodermal Activity (EDA) is a low-cost and non-intrusive way of monitoring the emotional state of a subject, and a viable gateway to study the Sympathetic Nervous System (SNS), which is responsible for the so-called fight-or-flight responses happening at the unconscious level (Boucsein, 2012). EDA is also a frequently used modality in psychophysiology, because of its ability to obtain a distinct Electrodermal Response (EDR) in response to a stimulus. The possible uses of EDA are well documented; the most important examples include research on epilepsy, autism, stress, and anxiety (Banganho et. al., 2022). More specifically because the phasic component or Skin Conductance Response (SCR) signal obtained from decomposing an EDA signal is closely associated with SNS activation when stimuli such as external stressors are present, numerous studies have emerged detailing how these signals can be exploited to detect stress within an individual. Therefore a brief overview of these studies detailing how indeed stress can be detected using only these physiological signals will be followed in subsequent sections.

Emotional fluctuations cause sweating, which is most evident in the palms, fingers, and soles. This process alters the quantity of salt in the skin and changes its electrical resistance. This sensor provides less information about the emotional state compared to EEG and ECG, but it has a few advantages: it requires fewer measuring electrodes, which allows for easier use of wearable devices and definition of emotional states in ambulatory settings; it provides fewer raw data, which allows for faster analysis

of obtained data; and it requires less computational power, and compared to other measurements, equipment needed for the measurement is much simple and cheaper.

Changes in electrodermal activity (EDA) can be either short-term (phasic) or longer-lasting (tonic). Phasic changes, known as skin conductance responses (SCR), are quick shifts in electrical resistance occurring within seconds. In contrast, tonic changes, measured as skin conductance level (SCL), represent EDA over a longer period (Boucsein et al., 2012).

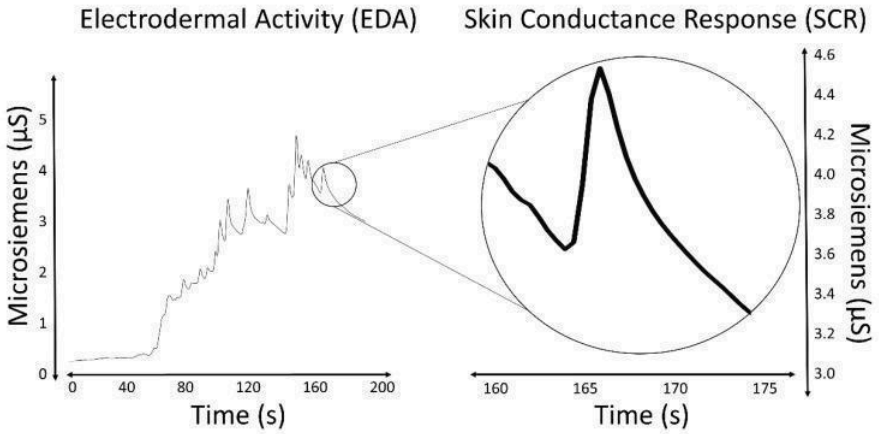
Figure 1. Components of EDA: Phasic (Skin Conductance Response) and Tonic (Skin Conductance Level)



On the other hand, Jose (2022) has stated that within the domain of statistical computing, time series analysis is a statistical technique that deals with trend analysis and time series data. Time series analysis made its way into medicine when the first practical electrocardiograms (ECGs), which can diagnose cardiac conditions by recording the electrical signals passing through the heart, were invented in 1901. Time series analysis is a specific way of analyzing a sequence of data points collected over an interval of time. In time series analysis, analysts record data points at consistent intervals over a set period rather than just recording the data points intermittently or randomly.

Given that signals recorded from ECGs, EEGs, EMGs, EOGs, and GSRs/EDAs share similar features in that they are sequences of data points gathered over time, GSR or EDA data may be classified as time series-specific data.

Figure 2. Skin Conductance Response relative to time



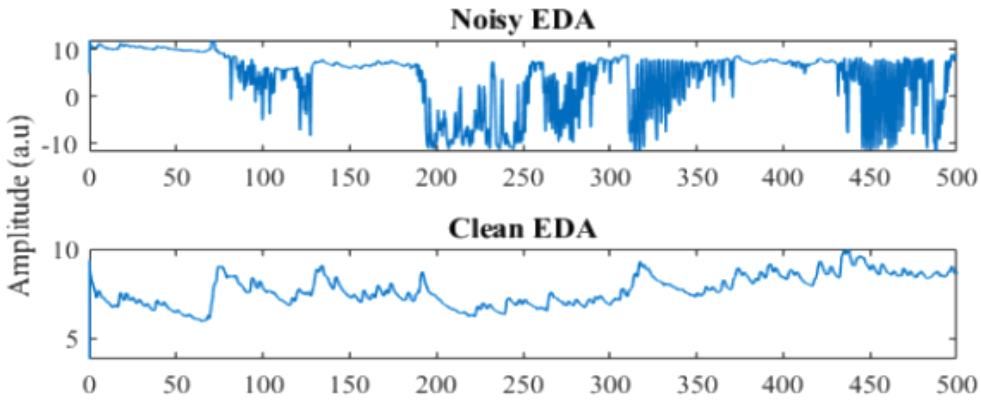
However, similar to other physiological signals, EDA signals face challenges. With the recent trend of wearable technologies for psychophysiological monitoring in ambulatory settings, the data quality from these sensors can be compromised by "noise" or artifacts in long-term recordings. Artifacts, which are unwanted changes in recorded biosignals not originating from the intended source (Boucsein, 2012), can result from unstable electrode contact, environmental factors like temperature and humidity, or movement. The referenced and noisy EDA channels are almost similar in shape, although different in amplitude (Hossain et al., 2022).

According to Boucsein (2012), detecting these artifacts typically requires visual inspection of the data, but this is unreliable in terms of large-scale EDA studies and long-term monitoring outside clinical settings. While signal processing techniques like low-pass filtering can be used to reduce the need for visual inspection, they risk altering

the physiological response, potentially transforming the entire EDA trace and making artifacts appear as genuine responses.

If these noises persist in the signal when it is examined, they are susceptible to misunderstanding and distort the results; for example, they could be confused with a skin conductance response (SCR), a physiological reaction that may indicate increased stress (Taylor et. al., 2015). Recent studies have focused on developing models to automatically identify and remove artifacts (Gashi et al., 2020).

Figure 3. Example of Noisy and Clean EDA signal from the study of Hossain et. al. (2022)



In the study conducted by Llanes-Jurado et al. (2023) on automatic recognition and elimination of artifacts in electrodermal activity (EDA) signals using their EDABE dataset, they collected data from 43 participants in a stress-inducing VR study. The researchers developed and trained four models, two of which replicated traditional machine learning methods by Taylor et al. (2015) and Hossain et al. (2022).

Replicating the Taylor et al. (2015) method, they extracted 62 hand-crafted features, selected 40 via backward selection with a Support Vector Classifier, and used three classifiers: Logistic Regression, Support Vector Classifier, and Random Forest.

They performed hyperparameter tuning and selected the model with the highest accuracy.

Additionally, they also reimplemented the same methodology used in the recent paper by Hossain et al. (2022), where instead of 5s segments as it was in the latter paper, they used 0.5s segments of EDA signals. They engineered typical statistical features as in the paper by Taylor et al. (2015). The researchers included the optimized coefficients of an autoregressive model as features, excluding the bias or intercept coefficient. Finally, they used two time-frequency transformation methods to extract time-frequency features to capture non-stationary characteristics from the signals. A total of 50 features were engineered and extracted from the raw EDA signal data, then reduced to 40 using a Random Forest classifier as a feature selection method. The features before being fed as input for Support Vector Machine, Gradient Boosted Tree, Random Forest, and Logistic Regression classifiers were standard scaled and normalized using min-max. To find the best model for each classifier, the use of hyperparameter tuning has been repeated for each classifier together with the use of 5-fold cross-validation to select the best model out of each classifier. The model that had the highest accuracy out of each classifier category was defined as the best model.

In addition to these, they proposed new models, which includes LSTM with a 1D-CNN and a 2D-CNN for analyzing signal spectrograms. The LSTM-1D CNN model recognized 72% of artifacts with 76% AUC (area under curve), 57% F1-Score, and 88% accuracy on the test set. Future work includes adding expert manual correction, developing movement protocols, and fine-tuning model architectures. This study reaffirmed that SVM, Gradient Boosted Tree, and Random Forest classifiers achieved the best accuracy on the validation set, consistent with previous findings.

In another study of Lee et al. (2020), artifact detection is a crucial aspect addressed in the research. The denoising method proposed in the study focuses on alleviating intrinsic respiration noise and extrinsic noise in Electrodermal Activity (EDA) signals collected by a wearable biosensor. The method involves detecting and attenuating irregular respiration-induced noise in EDA signals. Specifically, irregular respiration is identified and removed as respiration noise using a machine learning model that detects irregular respiration patterns from Photoplethysmography (PPG) signals collected simultaneously. This artifact detection process is essential for improving the accuracy of stress measurement by reducing noise interference in EDA signals, thus enhancing the reliability of stress metrics extracted from the data.

Despite these findings, traditional machine learning methods have limitations in healthcare signal processing. Sun et al. (2018) found that hand-designed EEG feature extraction methods resulted in less analytical performance. They addressed this by using recurrent autoencoders for feature extraction. Hussein et al. (2018) further supported this by employing LSTM networks to capture high-level patterns in EEG signals. In their approach, a fully connected layer was used to extract resilient, epileptic-relevant features, while a softmax layer provided predicted labels. This method maintained high detection performance, particularly in identifying artifacts like eye movements, muscle movements, and background noise. Additionally in a study by Jamshidzadeh et al. (2023), one of these limitations with regards to the use of traditional ML methods was specifically the inability of SVM models to extract significant features from data.

Moreover, while it has been established that convolutional neural networks achieve superior performance on high dimensional data such as images and are used in different automated tasks, such as classification, detection, segmentation, and data augmentation Szegedy et al. (2016) CNNs however according to Wang et al. (2020),

Park & Yang (2019), Li et al. (2022) are still considered as a ‘Black box’ in terms of the underlying mechanism which makes it difficult to interpret the results and have confidence that they provide the optimal solution. A paper by Azam et al. (2023) also states that the ‘black box’ nature of CNNs is a key research interest currently where different research has attempted to provide different insights. There are several research questions related to the ‘black box’, such as the inner mechanism of CNN layers, feature interpretation, and decision-making schemes, and as such CNNs are still poorly understood, not only by non-technical users but also by experts. This lack of knowledge according to Lange et al. (2018) may cause ambiguity and hesitance in relying on the predictions of CNNs, especially in critical applications like the medical domain. And by opening up the ‘black box’ can increase the confidence of users such as medical specialists in the results of neural networks as per studies conducted by Ferdinand & Mercier, (n.d.), Brahimi et al. (2018), and Dependent et al. (2021).

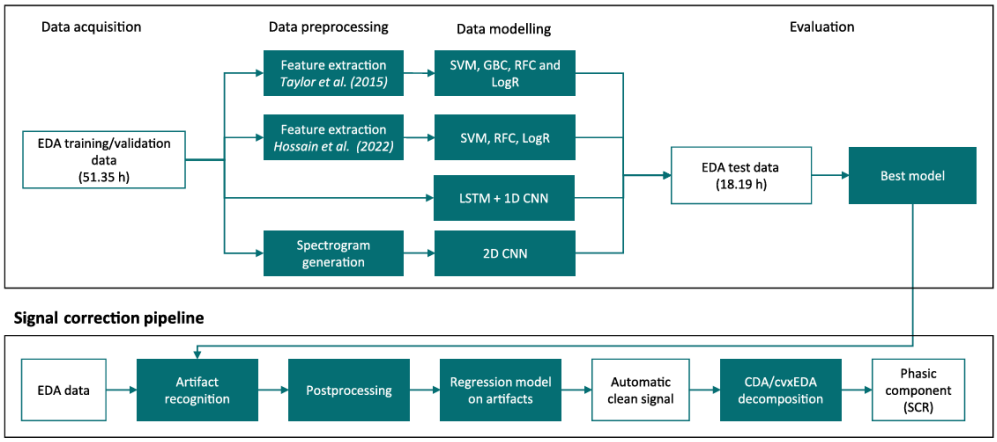
Finally, to deal with the difficulties encountered in the studies mentioned previously, the researchers propose a hybridized LSTM-SVM model, in which the LSTM approach seeks to address the shortcomings of traditional machine learning (ML) techniques of feature extraction from time series data, particularly those involving EDA signals, through the inclusion of higher-order features as well as to lower order manually constructed features used by the previously cited studies, and wherein the integrated mechanism of conventional ML-based techniques such as SVM. This method employs LSTM and SVM to integrate both higher and lower-order features. The implication is that this arrangement will perform better than deep learning models, which predominantly use higher-order features, and typical machine learning models, which rely upon lower-order, handcrafted features. LSTM's capacity to capture complicated, high-level features is predicted to improve performance over utilizing solely manually engineered

lower-order features. Additionally, combining these higher-order characteristics with lower-order, manually constructed ones is anticipated to enhance overall model performance.

Theoretical Framework

The purpose of this study is to improve the detection of artifacts or noise in electrodermal activity (EDA) signals by utilizing advanced machine learning methodologies, specifically through a hybridized Long Short Term Memory (LSTM) hybridized with Support Vector Machine (SVM). This strategy will be contrasted with conventional machine learning approaches like Support Vector Machines, Linear Regression, and Random Forests, which have been previously utilized in stress recognition research.

Figure 4. Theoretical Framework



The theoretical framework for this study is rooted in Machine Learning, Deep Learning, and Signal Processing principles, specifically concentrating on the detection of stress using Electrodermal Activity (EDA) signals. Long Short-Term Memory (LSTM) networks and Support Vector Machines (SVM) are well-regarded machine learning methods, each providing distinct advantages for managing sequential and non-linear

data, respectively (Cimino & Dell'Orletta, 2016). By combining LSTM with SVM, this research seeks to harness the strengths of both approaches to improve the detection of artifacts and noise in EDA signals, particularly in contexts related to stress detection.

Electrodermal Activity (EDA) serves as an essential physiological indicator, reflecting alterations in skin conductance due to the activity of sweat glands as governed by the sympathetic nervous system. It is frequently utilized in studies regarding stress and psychophysiology. Nevertheless, precise analysis of EDA signals necessitates effective artifact elimination to guarantee that the acquired data accurately represents authentic physiological responses rather than distortions induced by movement or external factors. The existence of artifacts in EDA signals can notably affect the precision of stress detection models. Conventional methods for artifact identification typically depend on manual review or heuristic algorithms, which can be labor-intensive and may yield suboptimal outcomes (Gashi et al., 2020).

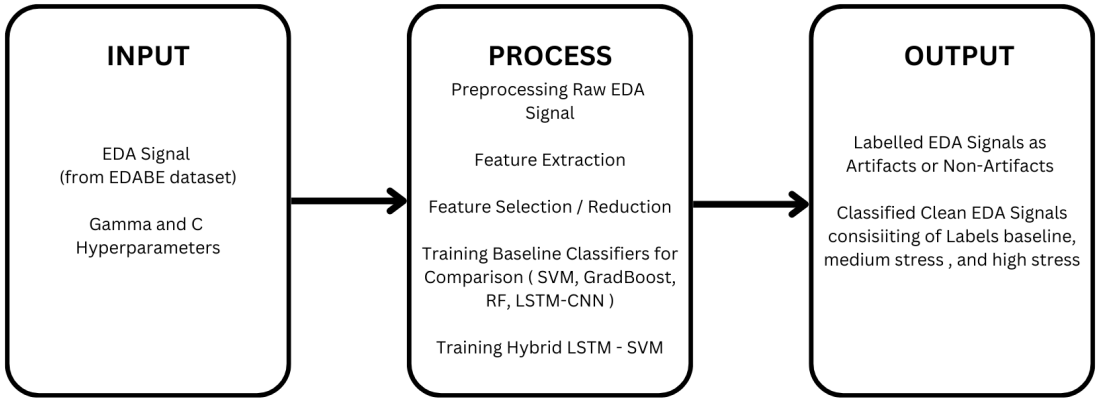
Machine learning methods such as Support Vector Machines, Linear Regression, Random Forests, and Naïve Bayes have been widely used for stress detection utilizing physiological signals. Typically, these methods involve extensive feature engineering and may not completely capture the temporal aspects of EDA signals. Conversely, Sequence-to-Sequence models can learn directly from the unprocessed data, which could enhance both artifact detection and overall model performance. This study will perform a comparative assessment to analyze the effectiveness of the proposed LSTM-SVM model on existing techniques proposed by Taylor et al. (2015), Hossain et al. (2022), and Llanes-Jurado et al. (2023). The evaluation will focus on essential performance metrics, including AUC, F1-score, precision, recall, and the accuracy of baseline classifiers.

The central thesis is that the hybrid LSTM-SVM model will surpass conventional methods in anomaly detection within EDA signals by adeptly utilizing contextual

information from both prior and subsequent data points. Through this comparative analysis of traditional artificial intelligence techniques, this research aims to further the field of stress detection by enhancing the accuracy and dependability of EDA signal analysis. This framework will provide direction for research design, data collection, analysis, and interpretation, ensuring a thorough evaluation of the proposed methodologies.

Conceptual Framework

Figure 5. Conceptual Framework



The conceptual framework illustrates the process of the application of hybridized Long Short Term Memory (LSTM) with Support Vector Machine (SVM). The framework is divided into three stages: input, process, and output. The Input stage involves the EDA signal data which is the Electrodermal Activity artifact correction Benchmark or the EDABE dataset and the gamma and c hyperparameters. The process stage involves the preprocessing of raw data, extracting relevant features, training various classifiers such as SVM, Gradient Boosted Tree, Random Forest, LSTM, and CNN, and training the hybrid LSTM-SVM. The output stage includes the preprocessed EDA signals, the selected

features, performance metrics such as ROC-AUC, Sensitivity, Specificity, F1-Score, Precision, and Accuracy for all baseline classifiers and the hybrid LSTM-SVM model and the labeled artifact whether the signals are artifact or not artifacts.

Statement of the Problem

The objective of this study will be to gauge the performance of our proposed LSTM-SVM model against other models that have already been established previously, and whether or not its performance will be significantly greater than that of the other models. In terms of evaluating the performance, the study asks the following questions:

1. What are the optimal Gamma and C hyperparameters for the SVM mechanism of the model such that it achieves higher evaluation metrics than using other Gamma and C values, to be used as the baseline of comparison to other state-of-the-art automatic artifact detection methods such as SVM, Logistic Regression, Random Forest, Gradient Boosted Tree, and LSTM-CNN?
2. What would be the performance of the hybridized LSTM-SVM model compared to other existing models (SVM, Logistic Regression, Random Forest, GradBoost, LSTM-CNN) that automatically detect artifacts in EDA signals in terms of its:
    a. ROC-AUC
    b. F1-Score
    c. Precision
    d. Recall
    e. Accuracy

Null Hypothesis

There is no significant difference between the performance of the hybridized LSTM-SVM model compared to the existing automatic detection models of SVM, Gradient Boosted Tree, Random Forest, Logistic Regression, and LSTM-CNN in terms of ROC-AUC, F1-Score, Precision, Recall, and Accuracy.

Scope and Delimitations

This study focuses on the application of hybridized Long Short Term Memory (LSTM) with Support Vector Machine (SVM) in the detection of artifacts in Electrodermal Activity signals for stress detection. The study will utilize the use of pre-existing datasets, specifically the Electrodermal Activity Artifact Correction Benchmark or the EDABE datasets which contain electrodermal activity (EDA) recordings of hand and body motion artifacts. This study is limited only to the detection of artifacts as the removal or the correction of artifacts as with the benchmark study of Llanes-Jurado et al. (2023) will be omitted. The process of this study includes the collection as well as the preprocessing of electrodermal activity (EDA) data, training the hybridized LSTM-SVM using the processed data to differentiate between authentic physiological signals and artifacts, and finally identifying artifacts from novel electrodermal data using the trained model.

Significance of the Study

This study holds significance in the improvement of Electrodermal Activity (EDA) for stress detection. This study will be beneficial for the following:

Caretakers. This study will allow caretakers to potentially be able to identify and distinguish what is noise from what is a real stress response.

Healthcare Professionals. This study may help healthcare professionals under the pressure of time in ambulatory situations to easily interpret and distinguish what is noise/artifacts from the stress response of a patient.

Healthcare Technology Companies. This study may grasp this advancement to develop more precise, non-surgical stress detection devices, leading to better stress management solutions and ultimately improving patient outcomes and quality of life.

Future Researchers. This study will help future researchers build on this hybrid approach to enhance the accuracy of data that leads to more reliable results and to the overall performance of EDA-based stress detection, potentially leading to more advanced and practical applications in health monitoring, psychological assessment, and stress monitoring devices.

Definition of Terms

Ambulatory Settings - refers to medical services performed on an outpatient basis, without admission to a hospital or other facility. An ambulatory setting is also defined as any environment where patients receive healthcare services without being admitted to a hospital. This includes clinics, doctor's offices, and home-based care environments where EDA signals might be collected.

Artifact/Artefact - In this study, it refers to unwanted disturbances or noise in EDA data that can compromise the accuracy of the measurements. It can arise due to various factors including motion, quantization errors, sudden changes in EDA associated with movement, or other sources of interference. As any anomalies in the EDA signal data that do not originate from the physiological

responses but from external or technical sources. These need to be detected and filtered out to ensure the accuracy of the stress detection.

Receiving Operating Characteristic-Area Under Curve (ROC-AUC) - it is a performance metric used to assess the effectiveness of the artifact detection model. ROC-AUC indicates better model performance in distinguishing between true artifacts and genuine physiological signals in the EDA data.

F1-Score - performance metric that is used to evaluate the accuracy of the model in predicting artifacts and the ability of the model to detect all the actual artifacts.

Precision - performance metric that is used to evaluate how reliable the model identifies artifacts in Electrodermal Activity (EDA) Signals.

Accuracy - a standard metric that measures the proportion of correctly classified Electrodermal Activity (EDA) Signals including both artifact and non-artifacts signals to the total number of signals evaluated.

Electrodermal Activity - pertains to the skin's electrical properties that shift in response to sweat secretion. These changes are frequently associated with mental and physical arousal. EDA is measured as the variation in the electrical conductance of the skin over time, reflecting the physiological arousal related to stress. The signals are collected using sensors and analyzed to detect stress levels.

Electrodermal Activity artifact correction BEnchmark (EDABE) - refers to a dataset and evaluation framework used to assess the performance of various artifact detection and correction algorithms in electrodermal activity (EDA) signals.

Convolutional Neural Network (CNN) - a deep learning algorithm designed to automatically and effectively identify artifacts in electrodermal activity (EDA)

signals. CNN processes EDA signal data through multiple layers of convolutional filters to extract relevant features and patterns indicative of artifacts.

Gated Recurrent Unit - is a type of recurrent neural network (RNN) that was introduced by Cho et al. in 2014 as a simpler alternative to Long Short-Term Memory (LSTM) networks. GRU is used as a component of the hybridized model for processing time series EDA data, offering a simpler yet efficient way to capture dependencies in the sequential data.

Long-Short Term Memory (LSTM) - is a type of deep neural network that is designed to capture historical information of time series data and is suitable for predicting long-term nonlinear series. LSTM are employed to analyze and predict patterns in EDA signals by retaining information over longer periods, which helps in identifying stress related changes in the data.

Microsiemens (μS) - It is the unit of measurement for skin conductance, which is a key component for EDA. This study's goal is to use a machine learning approach to filter out noise and improve the reliability of stress detection from EDA signals.

Non-Stationary - refers to a process of time series of statistical properties, such as mean, variance, and autocorrelation that change over time.It can also arise due to trends, cycles, abrupt changes, or other time-varying behavior in the data. This involves identifying and addressing the time-varying nature of these EDA signals when detecting artifacts.

Psychophysiological Signals - refers to physiological responses of the human body that are influenced by psychological factors. These signals include EDA, heart rate, respiration rate and blood pressure but it is not limited. It specifically focuses on EDA signals, which are measurements of the skin's electrical conductance caused by sweat gland activity. EDA is closely related to the

sympathetic nervous system's activity and is often used as an indicator of emotional arousal or stress detection.

Recurrent Neural Network - is a deep learning model that is trained to process and convert a sequential data input into a specific sequential data output. RNNs including their variants like LSTM and GRU, are utilized to handle the sequential nature of the EDA data, enabling the detection of stress over time.

Stationary - It is the contrast of nonstationary processes, where the statistical properties such as mean, variance and autocorrelation, do not change over time. By identifying the segments of the signal that deviated, this can pinpoint the artifacts. Also, by analyzing its properties of the signals this can differentiate between normal variations and stress-induced changes.

Support Vector Machine (SVM) - it is a supervised machine learning used in classifying electrodermal activity (EDA) signals using features categorized into statistical, auto-regressive, and time-frequency related features to identify artifacts or non-artifact

CHAPTER 2

REVIEW OF LITERATURE AND STUDIES

Electrodermal Activity Artifacts Detection using Machine Learning Techniques

To establish context, artifacts are specifically defined as "changes in the recorded biosignal that do not stem from the signal source in question", which can be caused by the method of recording or recognized physiological responses in the system which are not electrodermal signals. These may result in waste of efforts in obtaining electrodermal signals since they give rise to unreliable data because of low quality signals extracted. This work used several methods for an autonomous technique to detect artifacts in the structure of EDA signals and evaluated the signal quality in terms of thermoregulation responses (user movement, and ambient temperature). The XGboost Classifier achieved a recall of 98%, which is a significant improvement of 42 percentage points compared to the baseline classifier. It is claimed that the technique can replace or minimize the efforts of human specialists to visually analyze the retrieved data, but more extensive future study into additional elements that may appear similar to genuine signals would make a substantial contribution as Gashi et al. (2020) detailed.

On another note how artifacts are detected by Machine Learning and/or Deep Learning models is a computationally complex process, however manual preliminary annotation and/or labeling of raw EDA signals such as in studies by Taylor et al. (2015), Hossain et al. (2022), and Llanes-Jurado et al. (2023) have all used different criteria whether or not a signal was to labeled an artifact or not an artifact. Firstly in the study by Taylor et al. (2015) a signal was labeled an artifact if firstly, the peak did not show exponential decay, (however it may be dependent on the context e.g. if two SCRs (skin conductance responses) occur close together in time, the first response may not decay before the second begins, yet this is not considered an artifact), secondly, if it has a

quantization error with ≥ 5% of signal amplitude, thirdly if it has a sudden change in EDA correlated with motion, and finally that the signals SCL (skin conductance level) is less than or equal 0.

In the paper by Hossain et al. (2022) they validated their motion artifact detection algorithm, by systematically adjudicating the dataset as artifacts and clean signal albeit manually. Taylor et al. (2015) their process involved obtaining three experts/observers to manually annotate the signals using specific criteria. No fixed window was defined for the annotation, as the observers were able to mark the start and end of artifact segments. Firstly, a signal was annotated as a motion artifact if the EDA signal ranged outside 10 seconds to 40 seconds inclusively, secondly if the EDA signal changed quickly i.e. faster than ± 10, thirdly if the EDA peak decays i.e. (EDA is considered noisy if EDA peak does not follow an exponential decay except when there are two peaks within a short period), and finally if there was correlation of reference and noisy EDA channels (however sometimes a signal is considered clean and without artifacts if and only if the correlation coefficient was greater than or equal 0.95). In addition, in the study by Llanes-Jurado et al. (2023) although the use of experts to manually annotate the raw EDA signals according to a set of criteria was again conducted, explicit criteria such as those used in the two previous studies were not shown.

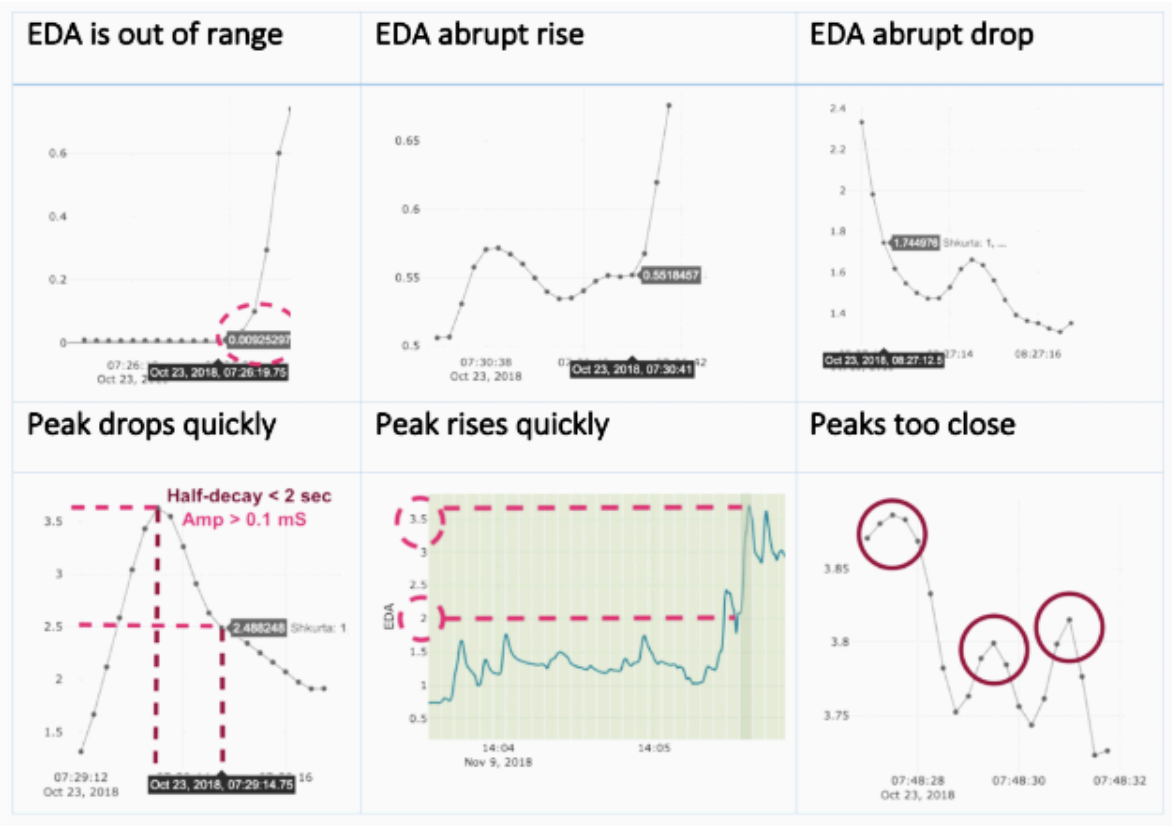
To identify and label whether the signal received from the sensor is a genuine EDA response or an artifact, here are four criteria as guidelines for annotating the signal to label if it's an artifact according to Hossain et. al. (2022):

Table 1

Guidelines for Annotation of EDA Signals

Index	Criteria
1	EDA out of range (*EDA range -10s to 40s*)
2	Quick change in EDA (*if EDA changes faster than* $\pm\ 10s$)
3	EDA peak decays (*EDA is considered noisy if EDA peak does not follow an exponential decay except when there are two peaks within a short period*)
4	Correlation of reference and noisy EDA channels (*considered clean only if the correlation coefficient is* $\geq\ 0.95$)

Figure 6. Some guidelines developed to aid human annotators in the data labeling process from the study of Gashi et. al. (2020)



Limitations of Machine Learning and Deep Learning Techniques in EDA Artifact Detection

Similarly, Llanes-Jurado et al. conducted a study titled "Automatic Artifact Recognition and Correction for Electrodermal Activity in Uncontrolled Environment". al. (2021), wherein the researcher discussed the influence of movement artifacts on recorded EDA signals in uncontrolled circumstances, resulting in the obscurity of significant patterns. This study investigated the application of a variety of machine learning and deep learning technologies, including support vector machines, recurrent neural networks (RNNs), and convolutional neural networks (CNN). In the experiment, the model that employed an RNN fed with the raw data recognized 72% of the artifacts and achieved an 87% accuracy rate. The identified artifacts were then automatically corrected using linear interpolation and a high degree polynomial. When assessed, the automatically and manually adjusted signals showed variations from the raw signals. The study produced significant results with regards to the future experiments that may be able to improve and develop artifact detection in EDA signals.

In an earlier study by Taylor et al. (2015), they at the time also sought develop not only a classification algorithm for automatically detecting artifacts, but an online system hosted at eda-explorer.media.mit.edu that will apply the algorithm to users' uploaded EDA files to provide them with an analysis of which portions contain artifacts. Focusing however on the development of an artifact detection model their research worked on the collection of EDA signals from 32 participants and with these signals they built an interface that allowed their two experts to review these epochs and assign a label of either 'artifact' or 'clean' based on a set of criteria that were defined. Although the classification labels were created using these criteria, the website also provides the ability for other researchers to agree to label their own data reading to their individual application needs. Once signals were annotated/labeled by the experts, statistical and

time frequency related features were then hand crafted and engineered manually to subsequently feed into their proposed set of models for ensemble training. Wrapper feature selection was implemented to reduce redundant features from the set of hand crafted features. The data was then split into training, validation and testing to feed into a variety of algorithms such as Neural Networks, Logistic Regression, Random Forest, Naive Bayes, K-Nearest Neighbor, and Support Vector Machine, with default hyperparameters. The model that produced the highest validation accuracy was selected which ended up being the SVM, as such it was focused upon for the remainder of the paper and in which produced an overall training, validation, and testing accuracy of 76.0% 96.95% 95.67% following a hyperparameter combination that used a Radial Basis Function as its kernel, a β value of 0.1, and a C value of 1000, after prior tuning of the kernel, β, and C hyperparameters. Overall SVM proved to be a promising algorithm given the high validation and testing accuracy it had.

Hossain et. al. (2022) also explored the automatic detection of electrodermal activity data using machine learning. In this study, the researchers worked on an annotated electrodermal database to label data as clean or noisy using a reference signal without motion artifact for a more accurate resolution. In the methodology of the study, a binary classification to detect the EDA segments with motion artifacts was developed. Features from the signals were then extracted, classified, undergone hyperparameter tuning, and evaluated. This experiment resulted in automatic detection of motion artifacts using a 5-fold cross validation method with the Gradient Boosted Tree out performing all other classifiers proposed (including the baseline models originating from Taylor et al. (2015) and Kleckner et al. (2018)) in terms of Accuracy, F1-Score, Sensitivity with mean scores of 94.82%, 91.61%, 92.90% respectively, excluding however Specificity since the LDA (latent dirichlet association) model had a mean score of 97.20% but still making the Gradient Boosted Tree a close second having a mean

score of 95.75%. However, there is still room for further studies as this study had the limitations of the data being collected from only 20 subjects which may not reflect the overall population and hence, might not be sufficient to characterize most typical types of motion artifact.

Sánchez-Reolid et al. (2022) study applied machine learning algorithms to classify arousal levels from electrodermal activity signals while implementing different strategies to deal with EDA signal complexity. The study employed Support Vector Machine (SVMs) with different kernels then employed Auto-HIdden Markov Models (AHMMs) for temporal modeling together with Discriminant Analysis (DA) for dimensionality reduction as its main techniques. The experimental design selected Decision Trees (DTs) in combination with ensemble methods together with Naive Bayes method because of their simple structure and reliable performance.Logistic Regression (LR) served as the designated binary classifier , yet K-Nearest Neighbors (KNN) and Artificial Neural Networks (ANNs) performed analyses on the EDA signal complexity. The artifact detection procedure involved signal normalization techniques with filtering methods to remove artifacts and applied noise reduction followed by feature extraction, and lastly model evaluation using accuracy, precision, recall, specificity, F1-score, AUC, and ROC metrics.

According to Tronstad et al. (2022) Electrodermal Activity data extraction requires unique time series data analysis methods. The analysis of cognitive processes and electrodermal responses through time series data employs Linear time - invariant (LTI) systems to both detect stimulus effects on EDA and derive cognitive or neural inputs from the obtained data. Time series data play a dominant role in deconvolution approaches by enabling sympathetic nervous activity (SNA) time series estimation and peak identification of physiological or psychological behaviors along with statistical

interval information analysis for inverse modeling to derive cognitive and neural inputs from observed EDA responses.

Under Virtual Reality (VR) conditions Llanes-Jurado et.al. (2023) studied electrodermal activity responses from 43 tes participants using Electrodermal Activity Artifact Correction Benchmark (EDABE) dataset. These signals originated from subjects in their natural form and the researchers had provided automatic alterations to the EDA signals for additional reference information. Taylor et al. (2015) and Hossain et al. (2022) developed four specific models through three traditional machine learning methods for their study. Support Vector Classifier in Llanes - Jurado et al. (2023) produced 62 features by running extraction procedures and backward feature selection methods on the 40 characteristics as specified by Taylor et al. (2015). The system passed its features to Logistic Regression (LR), before Support Vector Machine (SVM) processed them followed by Random Forest (RF). Llanes - Jurado et al. (2023) developed the model selection methods for each classifier which the author implemented through hyperparameter tuning methods. The validation accuracy test helped identify which model would be chosen from a set of fitted and trained classifiers based on three types. Replicating the Taylor et al. (2015) and Hosasin et al. (2022) methodologies the models that emerged having the highest performance in terms of accuracy was the Random Forest (contrary to the SVM model by Taylor et al. (2015)) and the Gradient Boosted Tree classifier respectively. These two baseline models were now compared to their proposed LSTM-CNN as this was projected to be able to identify temporal evolutions and features even within raw EDA signals itself. It had the following hyperparameter configuration: first two layers were LSTM layers of 16 neurons that returned the hidden state output for each input time step. Subsequently, the network included four convolutional levels, each of which featured three convolutional layers with a batch-normalization operation performed after each convolution. Finally, each level

included a dropout value of 0.05 and a max-pooling operation of size 2. The numbers of filters in each level were 32, 64, 128, and 256; kernel size was 5. Finally, the model featured two fully connected layers of 256 and 16 neurons and a final fully connected layer comprising a single perceptron with a sigmoid activation function. The model was trained with the rmsprop optimizer at a learning rate of 5×10^{-5} and a batch size of 16. Due to the imbalance of the dataset labels, the cost function used to train the model was the Dice-Sørensen coefficient (DSC). The model had an early stopping threshold of 30 epochs. The percentage of artifacts in the training set was 12.60%. No filter was applied to the raw signal and for each 5 s segment, min–max scaling was applied. After training, testing the LSTM-CNN proposed by Llanes-Jurado et al. (2023) the model had its AUC (area under curve), Dice Sorensen Coefficient (F1-Score), TPR (True Positive Ratio), and Kappa mean scores of 76%, 57%, 65%, and 49% respectively essentially out performing all other baseline models by Taylor et al. (2015) and Hossain et al. (2022) highlighting the ability of the LSTM to extract temporal features from even raw EDA signals without having to hand craft features from the dataset itself. Although the proposed model was third and a close second in terms of the Accuracy and True Negative Ratio metrics, this was negligible, as due to the present imbalance of the dataset, AUC, DSC, TPR, Kappa, are better overall metrics as opposed to solely just relying on an Accuracy metric itself, because of an imbalance of labels in the dataset. Overall despite skewed labels the proposed model represented a large increase in artifact detection performance relative to previous methodologies, being able to detect artifacts 59.88% of the time given a 50% threshold for the loss function, and 81.39% of the time given a 20% threshold for the loss function.

Although previous studies particularly those using traditional machine learning methods did have promising results such methods still do have limitations in healthcare

signal processing. A study by Sun et al. (2018) found that hand-designed EEG feature extraction methods resulted in poor analytical performance. They addressed this by using recurrent autoencoders for feature extraction.

Advantages of LSTM

High-level representations in EEG signals were acquired through models developed by Hussein et al. (2018) which used LSTM networks. A fully connected layer supplied stability features alongside epileptic information and then the softmax layer produced predicted labels. The detection performance remained consistent using this technique when identifying artifacts which included eye movements as well as muscle movement and background noise. Results based on medical clinical data show that the proposed method outperforms standard techniques through complete accuracy measurements and perfect sensitivity and specificity ratings. EEG signals data issues in real-life conditions did not affect the reliability of the researchers proposed model since it managed to maintain high detection performance regardless of common EEG artifacts and background noise.

Additionally, in a study by Jamshidzadeh et al. (2024), one of these limitations with regards to the use of traditional ML methods was specifically the inability of SVM models to extract significant features from data. They also added that there is always a dependency of the accuracy of the model on the choice of input combinations, which requires the use of resilient input selection methods i.e. feature selection which can be prone to bias due to it being manually done by humans and so feature selection methods like SVC and Random Forest have been used as with studies by Taylor et al. (2015) and Hossain et al. (2022) to select appropriate features from the data.

The study conducted by Koklu et al. (2022) and Agarwal et al. (2023) on water quality prediction also support this claim. Developing SVM models through rigorous feature engineering and extraction processes for accurate water quality predictions was necessary as they struggled with time series data, which is quite complex.

In the study by Llanes-Jurado et al. (2023), it has been shown that convolutional neural networks (CNNs) excel in handling high-dimensional data, such as images, and utilized in various automated tasks, such as classification, detection, segmentation, data augmentation (Szegedy et al. (2016). However, as noted by Wang et al. (2020), Park & Yang (2019), Li et al. (2022), CNNs are still considered as a 'Black box' regarding their underlying mechanisms, which complicates the interpretation of results and raises concerns about reliability of the solutions they provide. A study by Azam et al. (2023) also states that the 'black box' nature of CNNs is a key research interest currently where different research has attempted to provide different insight. There are several research questions related to the 'black box', such as the inner mechanism of CNN layers, feature interpretation and decision making schemes and as such CNNs are still poorly understood, not only by non-technical users but also by experts. This lack of knowledge according to Lange et al. (2018) may cause ambiguity and a hesitance in relying on the predictions of CNNs, especially in critical applications like the medical domain. And by opening up the 'black box' it can increase the confidence of users such as medical specialists in the results of neural networks as per the studies conducted by Ferdinand & Mercier, (n.d.), Brahimi et al. (2018), and Dependent et al. (2021).

About CNNs, as it is a subset of deep learning, deep learning is a more complex subset of machine learning. Deep learning models, albeit require more training time, often have provided higher accuracy as detailed by a paper by Kamath et al. (2018)

because they can perform automated feature extraction and classification concurrently, whereas a feature selection process is required before a machine learning algorithm.

Jamshidzadeh et al. (2024) provide an extensive description of RNN as a distinct regular neural-net model. The architectural design of RNNs featuring cyclic connections grants them the distinctive capability to manage sequences of different lengths, as documented by Wu et al. (2020). The RNN's hidden state serves as its critical memory component which undergoes continuous modification at each time step through the integration of incoming input data and the previous time step's hidden state. The architecture of RNNs can typically be depicted through conceptual diagrams where interconnected repeating modules or cells form chains that enable the retention of information from previous time steps through recurrent connections within the RNN structure. The hidden state calculated at time step t-1 emerges as a function dependent on the input x<t> recorded at time t alongside the preceding hidden state h<t-1> from time t-1 through their respective learned weight connections and subsequent activation function to achieve combined operation. Weerakody et al. (2021) examine how RNNs handle sequential data processing by managing inputs during their temporal flow. Through the combination of input x<t> and previous hidden state a<t-1>, a new hidden state a<t> emerges using task-specific learned weights followed by an activation function. Upon the update of a<t>, the RNN becomes capable of producing an output for the present time step t. Jamshidzadeh et al. (2024) Deliver a brief explanation regarding RNN operations for sequential data processing.

However, as Hochreiter (1991) and Mozer (1992) stated, RNNs are limited to look back in time for approximately ten timesteps rendering the RNN incapable of remembering information with longer term dependencies. This is due to the fed back signal that either vanishes or explodes. Nonetheless, as detailed by Gers et al. (2000) Gers et al. (2002), Hochreiter & Schmidhuber (1997), and Perez-Ortiz et al. (2003), this

issue was addressed with the advent of Long Short-Term Memory Recurrent Neural Networks (LSTM-RNN). Unlike its latter and computationally simpler counterpart, the RNN, LSTM networks are to a certain extent biologically plausible and capable of learning more than 1,000 timesteps, depending on the complexity of the built network as O'Reilly & Frank (2006) and Hochreiter (1997) has written.

Figure 7. Internal mechanism of LSTM cell

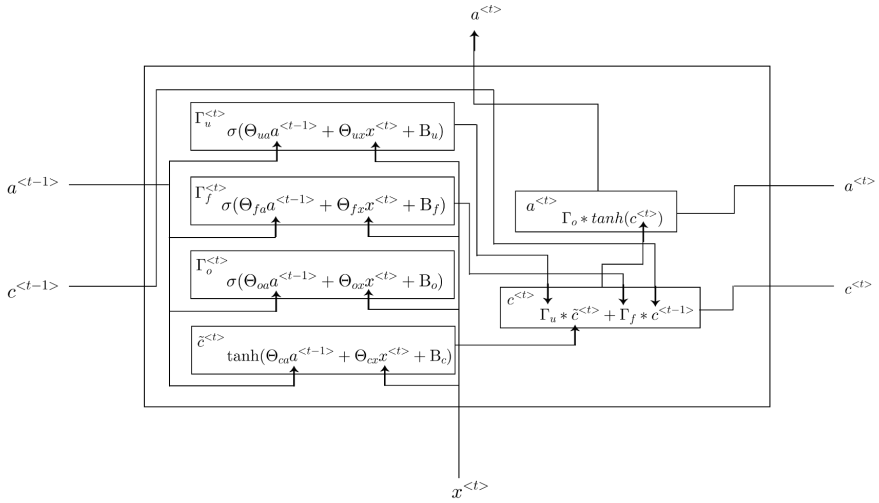
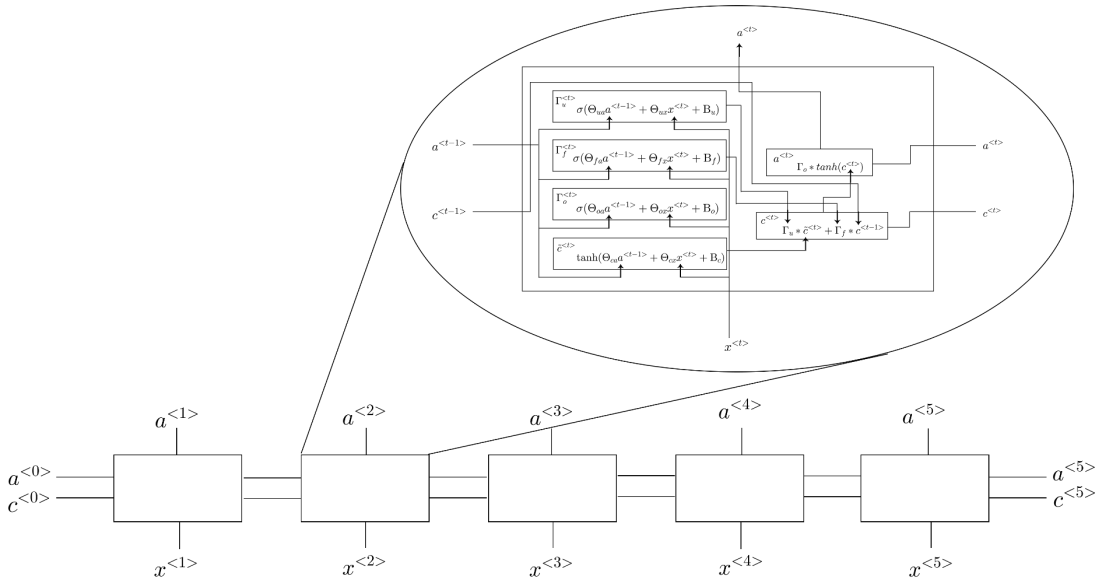


Figure 8. Full LSTM model



In place of a traditional RNN cell, the LSTM according to Wan et al. (2022) and Jaseena & Kovoor (2021) replaces the self-connected hidden units of a traditional RNN with memory blocks. The memory blocks contain multiple memory cells. Each memory cell stores information over a specific period and can be accessed and modified by different gate units. Specifically, LSTMs feature three types of gates: input (u), forget (f), and output (o), which control the flow of information. The input gate controls the flow of incoming information, whereas the forget gate selects which information should be removed from each memory cell.Finally, the output gate determines which information is sent to the LSTM's output, as described by Wan et al. (2022) and Zhang et al. (2022). The outputs of numerous gates, including the next concealed state and cell state at timestep t, are as follows:

$$\Gamma_u^{<t>} = \sigma(\Theta_{ua} a^{<t-1>} + \Theta_{ux} x^{<t>} + B_u)$$
$$\Gamma_f^{<t>} = \sigma(\Theta_{fa} a^{<t-1>} + \Theta_{fx} x^{<t>} + B_f)$$
$$\Gamma_o^{<t>} = \sigma(\Theta_{oa} a^{<t-1>} + \Theta_{ox} x^{<t>} + B_o)$$
$$\tilde{c}^{<t>} = tanh(\Theta_{ca} a^{<t-1>} + \Theta_{cx} x^{<t>} + B_c)$$
$$c^{<t>} = \Gamma_o * tanh(c^{<t>})$$
$$a^{<t>} = \Gamma_o * tanh(c^{<t>})$$

Hybridizing LSTM and SVM

In addition, further studies such as by Mohsen et al. (2023) which involved the classification of EEG signals using an LSTM and a traditional ML method the SVM classifier, shows that the LSTM provided the best performance, with a testing accuracy of 99.00%. Moreover, the weighted average precision, recall, and F1-score for the LSTM are 99.00%. The results of the SVM classifier in terms of accuracy, sensitivity, and specificity reached 91%, 93.52%, and 91.3%. Given such results they have shown that the LSTM classifier provides better performance than SVM in the classification of EEG signals. Reinforcing the idea that when it comes to feature extraction the LSTM itself

gives superior results as opposed to traditional machine learning methods that require manual feature extraction and engineering.

According to Jamshidzadeh et al. (2024), their study showed that for predicting water quality parameters their proposed models which consists of BILSTM-SVM, LSTM-SVM, BILSTM, and both a standalone LSTM, and SVM had accuracy indices of 97%, 92%, 89%, 85%, and 82% respectively at the training level. And thus it was concluded that BILSTM-SVM model performed better than LSTM-SVM model because it considered both past and future observations, secondly that the hybrid model (BILSTM-SVM and LSTM-SVM) performed better than the standalone LSTM, BILSTM, and SVM models, and thirdly that the SVM model had lower Accuracy Index (AI) and R2 values compared to other models, and all other models outperformed the SVM model. Similarly at the testing level, the AI values of BILSTM-SVM, LSTM-SVM, BILSTM, LSTM, and SVM models were 0.94, 0.86, 0.83, 0.81, and 0.78 respectively showing again that SVM models are always pale in comparison to LSTM models in terms of extracting features from time series data such as water quality parameters.

In another study by Juneau et al. (2021) which sought to detect foot strikes in lower extremity amputee populations since a viable model according to the researchers would provide the basis for automated stride parameter calculation and stride segmentation, which is essential for using new fall risk and health status AI models within clinical environments, their proposed models consisted of a traditional ML approach namely a Decision Tree and a Deep Learning approach the LSTM (Long-Short Term Memory) Neural Network. Their study has shown that the best performing classifiers out of the two was the LSTM (having a configuration of batch size of 64, dropout of 0.4, one LSTM layer with 100 hidden LSTM nodes, one dense layer with 50 hidden dense nodes, and a class weighting of 1:2 (label 0: label 1) which had an accuracy, sensitivity, specificity, and precision of 99.0%, 86.4%, 99.4%, and 83.7%

respectively over the decision tree classifier (having the configurations maximum tree depth of 10 and class weighting of 1:20 (label 0: label 1)) which had the aforementioned metrics of 98.7%, 82.8%, 99.2%, and 78.6% respectively showing still that a Deep Learning approach may be used to achieve higher performance metrics than traditional ML methods without the task of manually engineering & extracting features as well as selecting the most important features.

On the other hand, while deep learning methods may potentially address the problem of manual feature extraction and engineering by being able to extract high level features particularly that of biomedical signal data like EDA signals, as mentioned the problem of deep learning models particularly CNNs still have the challenge of interpretability when it comes to the predictions it produces, and due to its ambiguous nature medical professionals in particular have a hard time grasping how it arrived at such a prediction. Recent studies such as those by Mordensky et al. (2023) further support this as they proposed to use predictive modeling using traditional Machine Learning methods such as Logistic Regression, XGBoost, and SVM and more complex methods such as the use of Artificial Neural Networks in the domain of estimating moderate and high temperature geothermal resource favorability in the United States. Albeit having an imbalance of classes in their dataset i.e. having more positive labels, thereby having to use alternative performance metrics particularly the F1-score (as the performance metric accommodates for class imbalance better than other performance metrics e.g., accuracy Guo et al. (2008)), their study showed that while F1 scores of the proposed models appeared poorly, normal score transform was applied to the predictions of each model, and as a result it was found that the highest performing models was the single XGBoost and the ensemble trained XGBoost model having mean normal score transformed values of 1.88 and 1.61 respectively, of all the positive labels it predicted, over other models such as single Logistic Regression (LR), ensemble Logistic

Regression (enLR), single SVM, enSVM, ANN, having similar mean normal score transformed values of 1.32, 1.30, 1.24, 1.21, and 1.30 respectively. These scores have indicated the single trained XGBoost model produces the greatest distinction between known positive and unlabeled samples and that the ensemble trained XGBoost predicts the second greatest distinction between known positive and unlabeled samples. Similarly, the substantial overlap of feature importance in the F1 and ROC-AUC sensitivity analyses of the single ANN suggest that this deep learning algorithm was also detrimentally affected by its complexity when faced by the simplicity of the feature data. Hence, the highly complex machine learning models may not be as appropriate for data like that from the 2008 geothermal resource assessment the researchers used as the less complex algorithms such as XGBoost.

In another study by Sánchez-Reolid et al. (2019) they demonstrate yet again the simplicity of ML methods in the domain of stress detection from EDA signal data, as they've shown that the simplicity of the classification model they proposed, namely the SVM allows the system to work in the long term, as the SVM across 3 different sets of cross validation given 20 second segments of EDA signal features managed to produce at most 70.8%, 75%, and 75% accuracy, for 5, 7, and 10 folds of cross validation. Similarly 4 second segments of EDA signal features given to the SVM managed to produce at most 87.7% accuracy in the 5, 7, and 10 folds of cross validation.

In the more recent literature in the domain of sequential data processing such as NLP and signal processing, the recent study by Llanes-Jurado et al. (2023) where they proposed a hybridized LSTM-CNN model for automatic artifact detection in EDA signal data, it has fairly become common practice to adopt approaches of hybridizing Machine Learning and Deep Learning methods to making more accurate predictions from sequential data.

A study by Cimino & Dell'Orletta (2016) demonstrates this by hybridizing an LSTM and SVM in the domain of sentiment analysis i.e. subject classification, polarity detection, and irony detection. With this proposed model they sought to compare it with other baseline models, these being a standalone LSTM and an SVM model. They consider an SVM since it is an extremely efficient learning algorithm and hardly to outperform, unfortunately these type of algorithms capture "sparse" and "discrete" features in document classification tasks, making hard the detection of relations in sentences, which is often the key factor in detecting the overall sentiment polarity in documents as Tang et al. (2015) details. The mechanism of an SVM is defined as follows as per the book by Vapnik (1995):

$$h_{\theta}(x^{(i)}) = \theta^{\top} x^{(i)} + \beta$$

Equation 1. Linear SVM

Where $x^{(i)}$ is the input feature vector at row i, θ the non-bias weight/coefficient vector, and β the bias weight/coefficient.

$$min\ \frac{1}{2}\|\theta\|^{2}_{1} + C\sum_{i=0}^{m-1} max(0,\ 1 - y'^{(i)}(\theta^{\top} x^{(i)} + \beta))$$

Equation 2. Cost/Loss function

where C is the penalty coefficient, m is the number of training data points, λ− i and λ+ i are the penalty for the prediction errors; and out is the observed output. Equation (1) is rewritten based on the kernel functions:

$$h_{\theta}(x^{(i)}) = \theta^{\top} K(x^{(i)},\ l^{(i)}) + \beta$$

Equation 3. rewritten hypothesis using kernel function

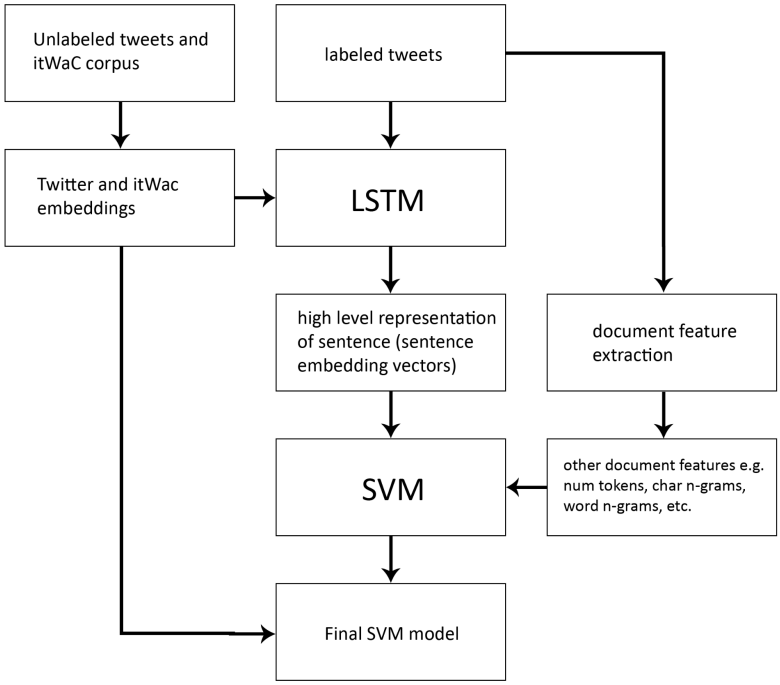
where $K(x^{(i)},\ l^{(i)})$ is the kernel function. In the modeling process, the radial basis function (RBF) is widely used as a kernel function. Previous studies such as those by Vapnik (1995) indicated that the use of a RBF as the kernel function gave high accuracy.

$$K(x^{(i)},\ l^{(i)}) = e^{\left(-\frac{|x^{(i)} - l^{(i)}|}{2\sigma^{2}}\right)}$$

There are many kernel functions i.e. Linear, Polynomial, Sigmoid etc. but the Radial Basis Function involves a the kernel parameter σ.

Furthermore, with the use of LSTM it was able to fortunately capture long term dependencies of a sentence, and hence in the training phase, the LSTM network was trained considering the training sentences and its corresponding gold labels of tweets in which the researchers decided to use for the task of sentiment analysis. Once the model was trained on each sentence of the training set, a sentence embedding vector consisting of real numerical values can then be obtained from the layer before the softmax classifier, which as previously mentioned would be the hidden state of an LSTM network. These sentence embedding vectors were then used as feature inputs to the SVM classifier in conjunction with a set of widely used document classification features i.e. lexical text, morpho-syntactic, and lexicon features.

Figure 9. Cimino & Dell'Orletta (2016) hybridized LSTM-SVM architecture

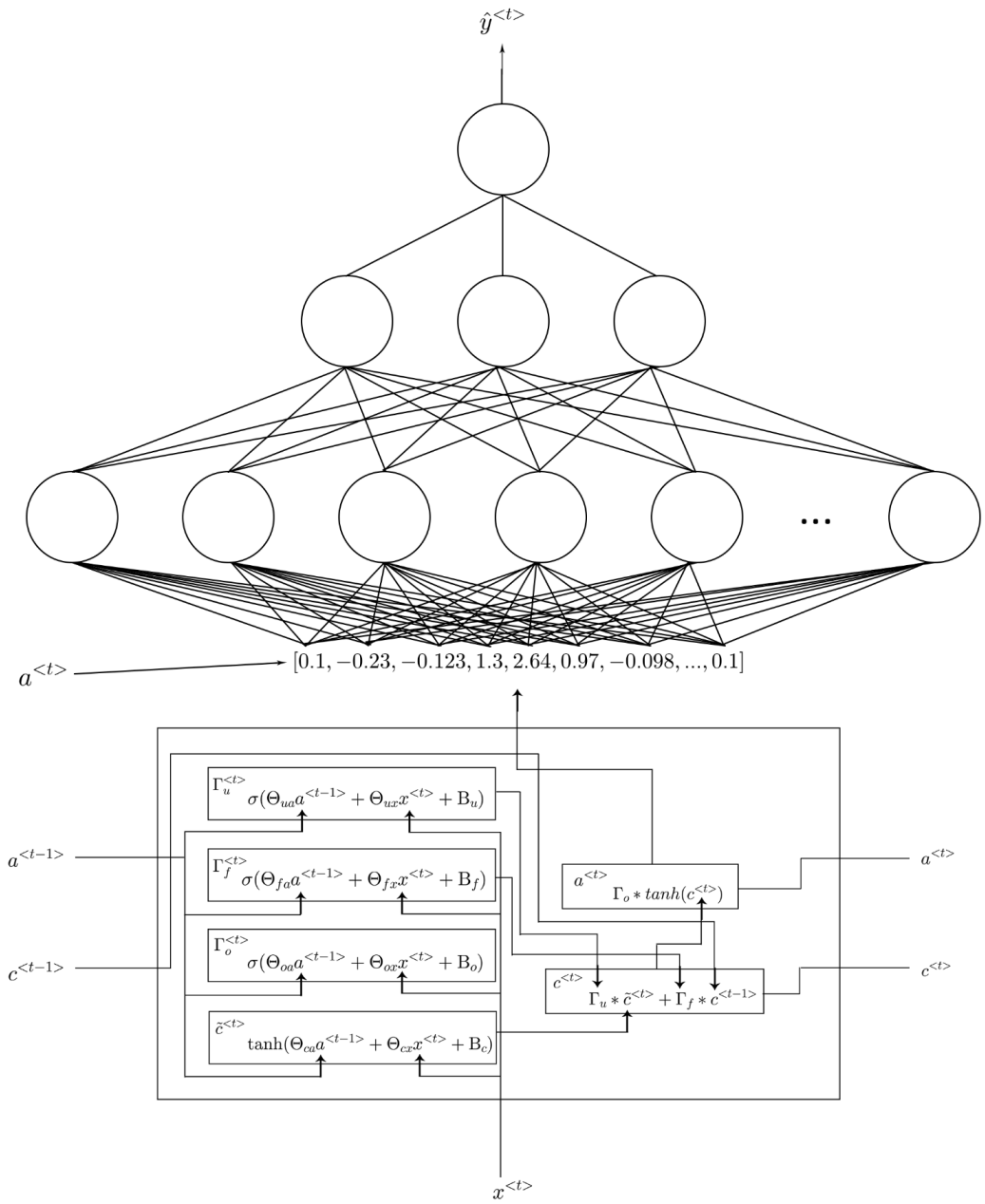


As for their results overall accuracies achieved by the classifiers on the validation data set showed that the standalone LSTM out performs specifically linear and quadratic SVM models in particular, having a 77.7%, 74.7%, and 64.6% accuracy in subject classification, polarity and iron detection tasks respectively over the latter of 74%, 73%, and 59.5% for quadratic and 72.5%, 71.3%, and 63.6% for linear. Moreover, the hybridized linear and quadratic LSTM-SVMs outperformed the standalone LSTM and the linear and quadratic SVMs, achieving 76.4%, 74.3%, and 66.2% validation accuracy for linear and 78.3%, 75.4%, and 67.5% validation accuracy for quadratic for subject classification, polarity and irony detection tasks respectively.

In a more recent study Zhang & Zhang (2020) they also proposed a hybridized model specifically for urban road short-term traffic flow prediction to solve and analyze the problems of periodicity, stationary, and abnormality of time series to improve traffic flow prediction effect, and achieve efficient traffic guidance and control. As this was a regression task the researchers chose mean squared error, its root (RMSE), mean absolute error, and mean absolute percentage error as the performance metrics to use, in place of typical accuracy, precision, recall, F1-score etc. for classification tasks, and where the lower the error value the better the model has performed. In place of the SVM as the traditional ML method as with the previous study the researchers proposed the use of a LSTM hybridized with the XGBoost (Extreme Gradient Boosting) classifier instead, then compared with other baseline models such as CNN, a standalone LSTM, a standalone XGBoost, LSTM-RNN, a model proposed by Wang (2019), and a model proposed by Chen & Chen (2019) to validate their hybridized model. This was because multi-step predictions can be challenging for time series data and because fully connected layers could potentially harbor too many parameters thus leading to overfitting the traditional ML method, the XGBoost classifier was used instead as an alternative

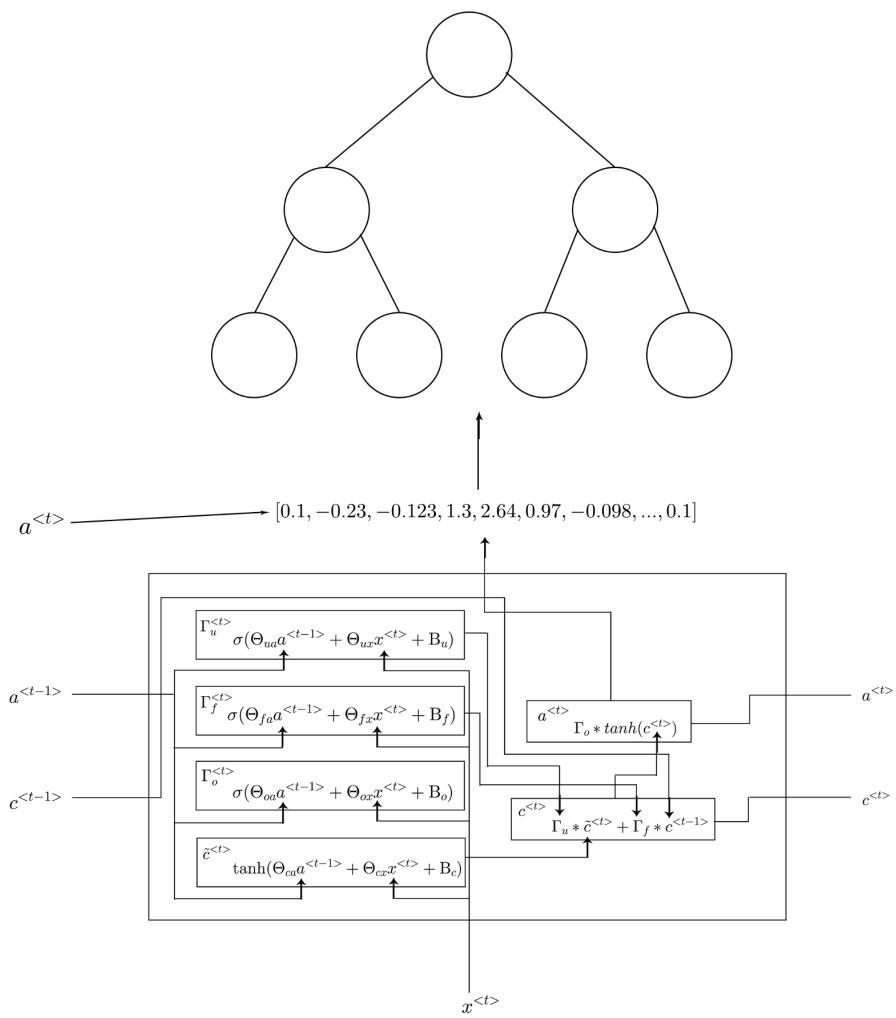
mechanism to which the final hidden state $a^{<T_x>}$ of the LSTM can be sent to, as feature vector inputs.

Figure 10. Typical LSTM network



Here in a typical LSTM network cell the computed hidden state $a^{<t>}$ is sent to a fully connected layer (dense layer) or immediately to a softmax to produce a probability vector $\hat{y}^{<t>}$ as the prediction. However as detailed by Zhang & Zhang (2020) this dense layer will instead be replaced with the XGBoost classifier itself, where instead of the hidden state being sent to a dense layer it is sent as a representation of the samples of the training data i.e. feature vector to the XGBoost classifier.

Figure 11. LSTM-XGBoost architecture



When it came to workday forecasting, performance indices in terms of MSE, RMSE, MAE, MAPE, showed the hybridized LSTM-XGBoost model to reign superlative in all metrics having values of 7.55, 2.75, 1.86, and 4.19% respectively as opposed to other models even the second best model by Chen & Chen (2019) having values of only 9.99, 3.16, 2.03, and 10.61% respectively. Moreover, as opposed to standalone models like the LSTM and XGBoost classifier, the hybridized models prediction accuracy, namely its MAPE as opposed to the two latter standalone models, improved by 6.73% and 0.91%. Similar values were also found when it came to weekday forecasting, when it came to the hybridized model as it achieved values of 4.10, 2.03, 1.49, and 3.3% respectively for MSE, RMSE, MAE, and MAPE, as opposed to even the second best model again by Chen & CHen (2019) having values of only 4.47, 2.11, 1.58, and 7.08% respectively for the aforementioned metrics. Moreover, as compared to the standalone models of LSTM and the XGBoost classifier, the hybridized approach improved by 3.72% and 0.02%. From these results the researchers concluded that their proposed LSTM-XGBoost hybrid model not only can improve prediction accuracy, but can also perform multi-step prediction, which is an effective method for traffic speed prediction.

Finally, in a study by Agarap (2017) the use of a hybrid GRU-SVM (a much computationally simpler version of an LSTM cell) akin to that of Cimino & Dell'Orletta (2016) tandem LSTM-SVM was also implemented to detect intrusions and malicious activity of users in cyberspace. The research attempted to solve this problem by again replacing the Softmax mechanism of a conventional RNN and replacing it with an SVM, because conventionally like most neural networks, the RNN and its variants (LSTM and GRU) employ the Softmax activation function as its final output layer for its prediction, and the cross-entropy function for computing its loss. Furthermore, the paper also amended this norm by introducing a different loss function in place of the cross entropy loss which is typically used with the softmax activation function especially when

predicting multi class labels: a margin-based loss function is used, it is defined as follows:

$$min\ \frac{1}{2}\|w\|^2_1 + C\sum_{i=1}^{n} max(0,\ 1 - y'_i(w^{\top}x_i + \beta_i))$$

However because optimization of a neural network requires backpropagation (differentiation of the loss function concerning the parameters of the network) this loss as detailed by Agarap (2017) cannot be differentiated, instead a different version of this margin-based loss function is used: the squared hinge loss, which will be differentiable and much more stable as Tang (2013) has stated.

$$min\ \frac{1}{2}\|w\|^2_2 + C\sum_{i=1}^{n} max(0,\ 1 - y'_i(w^{\top}x_i + \beta_i))^2$$

After training and testing, results show that, although the GRU-SVM was not able to surpass the prediction rate of GRU-softmax for true negative labels having only an approximate 75.60% as opposed to the latter of approximately 95.91%, and for false positive labels having a greater prediction rate of approximately 10.70% as opposed to the latter of approximately 4.09%, the GRU-SVM outperforms GRU-softmax as it was able to predict true positive labels approximately 89.3005% of the time and for false negative labels having a lesser prediction rate of approximately 24.40% as opposed to the latter of approximately 44.39%. Moreover in terms of overall accuracy the GRU-SVM outperformed GRU-softmax in both training and testing splits with values of 81.54% and 84.15% concerning each data split as opposed to the latter of 63.07% and 70.75% concerning each data split. As Agarap (2017) concludes, not only did the GRU-SVM model outperform the GRU-Softmax in terms of prediction accuracy, but it also outperformed the conventional model in terms of training time and testing time. Thus, supporting the theoretical implication as per the respective algorithm complexities of each classifier.

The LSTM automatically learns higher-order features, which capture complicated relationships between time and abstract patterns in the data provided. These representations outperform manually created features since they are based on the model's capacity to generalize and discover hidden relationships. As a result, models that use these higher-order features usually outperform those that depend entirely on lower-order, manually developed characteristics.

However, integrating higher-order features with lower-order, manually developed features improves performance even more. Manually built features frequently incorporate particular insights, such as statistical metrics or morphological qualities, that the LSTM may not have learnt directly. By merging these interpretable, particular tasks. characteristics with the LSTM's rich, abstract representations, the combined technique generates a more complete and resilient feature set. This combination produces a model that outperforms one that relies purely on higher- or lower-order features.

In a study by Sethi (2006), characteristics of low level and high-level features are stated. Low level features are the effect of high-level features. It is semantically less meaningful and includes components of a high-dimensional representation. It is easy to define and to extract. However, it is the weaker requirement of invariant extraction, and a weaker notion of invariance. On the other hand, high-level features are the cause of low-level features and are semantically more meaningful. It includes components of a low-dimensional representation, but it is difficult to define and extract. Moreover, it is the stronger requirement of invariant extraction and stronger notion of invariance.

In summary, low-order features are easier to define but tend to be less meaningful on their own and are more sensitive to noise and variations in the data. Essentially, they represent basic signal components. In contrast, high-order features are more difficult to define, more abstract, and capture broader patterns or concepts.

Stress Detection using Electrodermal Activity Signals

Stress detection is a crucial point in maintaining our mental and physical health, particularly in our rapid modern world, this involves identifying and assessing different indicators that may signify the existence of stress in an individual. Using machine learning in detecting stress provides an innovative combination of using technology in understanding and managing a psychological condition such as stress. Stress levels can be measured by using ML to identify patterns and signals found from physiological signals like heart rate, breathing patterns, electrodermal activity and others.

Stress can be differentiated into three types. Bakker et. al. (2011) provided three flavors of stress, which are acute, episodic acute, and chronic. Acute stress is caused by an acute short-term stress factor. Episodic acute on the other hand is an acute stress that occurs more frequently and/or periodically.

A recent systematic review by Sanchez-Reolid et al. (2022) details this as they describe how arousal classification pipelines are set up particularly those that involve stress. Arousal, as they define, is a general physiological and psychological activation of an organism, varying on a continuum from deep sleep to intense excitation. Arousal encompasses a wide terminology, as the construct arousal is a term that corresponds to the level of cortical activation that is regulated by the ascending reticular activation system, which varies from a level of over-activation, as in the case of intense emotions or alert states, to at best attentional level for intentional action, or to levels of under-activation, as in the case of relaxation or sleep states. Because stress is closely related to arousal in many works it is strongly associated with the use of the terms distress (negative stress) and eustress (positive stress) as Le Fevre et al. (2003) states and in addition stress is considered to be a form of negative affect arousal according to Boucsein (2012). Many researchers such as Moruzzi et al. (1947) and Posner et al. (2005) agree that variation in arousal (stress being one) correlates with increases in

many physiological variables such as heart rate, electrodermal activity (EDA), breath intervals and skin temperature, among others.

Numerous physiological variables have been used for arousal detection and its applications. EDA, according to Boucsein (2012), is considered especially useful in assessment of the arousal level due to its connection with the sympathetic nervous system (SNS). Alterations in the state of activation are reflected as variations in skin perspiration, which affects the conductance of the skin. In this aspect, many causal models are used to conclude arousal from EDA signals such as curve fitting, inverse filtering, general linear model for evoked skin conductance response (SCR), non-negative deconvolution, continuous deconvolution, dynamic causal model (DCM) for anticipated SCR and DCM for spontaneous fluctuations as Bach & Friston (2005) writes. As previously mentioned, many physiological signals or variables can directly be collected from the human body, these being Electrocardiograph (ECG) which describe the change in heartbeat and pattern of beating, Electromyograph (EMG) which describes changes in neuromuscular activity, Electrooculography (EOG) which describe eye movements, Electroencephalography (EEG) which describes the variation of electrical signals produced in different areas of the brain, and Electrodermal Activity (EDA) which as mentioned can be used to check the arousal of an individual given that this is an important variable for measuring one's emotional state.

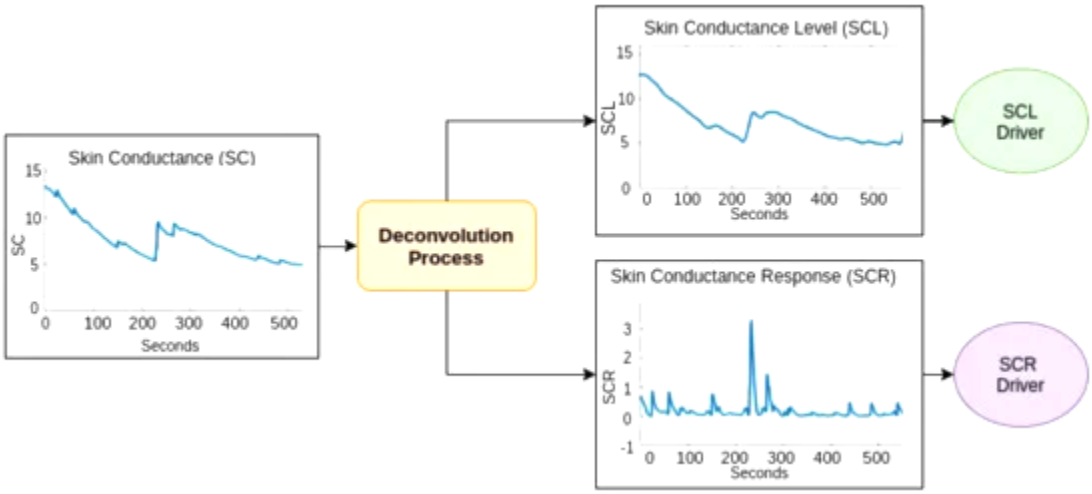
How EDA signals are specifically used for stress detection generally involves three stages. Firstly, acquiring the EDA signals through a transducer or a device that measures an individual's Electrodermal Activity. The measurement of EDA signals is usually conducted in two separate ways. Through exosomatic or endosomatic means. These measurements are composed of the convolution of two signals: the electrodermal level (EDL) or the tonic component of the EDA signal which varies slowly, and the electrodermal response (EDR) or the phasic component of the EDA signal which varies

rapidly. In addition, EDR, the phasic component of EDA is closely related to the activity of the sweat motor system, which is strongly associated with the sympathetic nervous system at the same time according to Bartolomé-Tomás (2020) which as Chu et al. (2024) also states is activated through stressful situations whether it be environmental or psychological and is largely in part responsible for us either avoiding it (flight) or facing it head on (fight) as a response to such a situation as Boucsein (2012) has written. To expand further the exosomatic way of measuring EDA is done by obtaining the variation of the resistance or conductance of the skin by injecting a small current onto it. This moreover is a measurement that is composed of two groups, AC and DC, depending on whether alternating or direct current is injected into the skin between the electrodes; more specifically the use of a direct current is what enables more specific EDA signals like Skin Conductance (SC), and Skin Resistance Response (SR) to be measured.

The next stage is pre-processing, which eliminates all the defects that have caused interference during the acquisition process. Here, artifacts are removed and the signal is filtered, thus making it softer and eliminating noise. Pre-processing as Sanchez-Reolid et al. (2022) review puts, includes three different steps: signal normalization, detection and elimination of artifacts and filtering of noise. The researchers also highlighted the importance of how artifacts that interfere with the signal must be removed, this is because a motion artifact (MAt) degrades signals very quickly and makes them unusable as Zhang et al. (2017) details. Although methods of eliminating artifacts by deflecting the signal through various softening filters can be applied, this procedure as Chen et al. (2015) and Hernandez et al. (2011) argues, causes in most cases a loss of information in EDA signals. This is the reason it may be imperative that other methods of detecting and removing artifacts such as those that are ML based be considered.

Furthermore, because EDA signals such as SC are initially structured in a way that the phasic and tonic components are entwined or convolved together, a process of deconvolution/decomposition as an important stage of signal processing must be done to obtain the components from these measurements. The most common of these methods in which EDA-based literature have mostly referred to are continuous decomposition analysis, which decomposes SC data in continuous tonic and phasic activity, and discrete decomposition analysis, which separates the SC data in a tonic component and discrete phasic components with a no-negative deconvolution. In addition Convex Optimisation on EDA signals (cvxEDA) is another method of deconvolution that represents the SC as the composite of three terms: the phasic component, the tonic component and an additive white Gaussian noise that incorporates the model's prediction errors as well as measurement errors and artifacts.

Figure 12. EDA Signal (SC) decomposition/deconvolution from Sanchez-Reolid et al. (2022)



Once the SC is deconvolved, the phasic component and the tonic component of the signal called the Skin Conductance Response (SCR) and the Skin Conductance Level (SCL) respectively are then obtained, and within the context of clinical analysis,

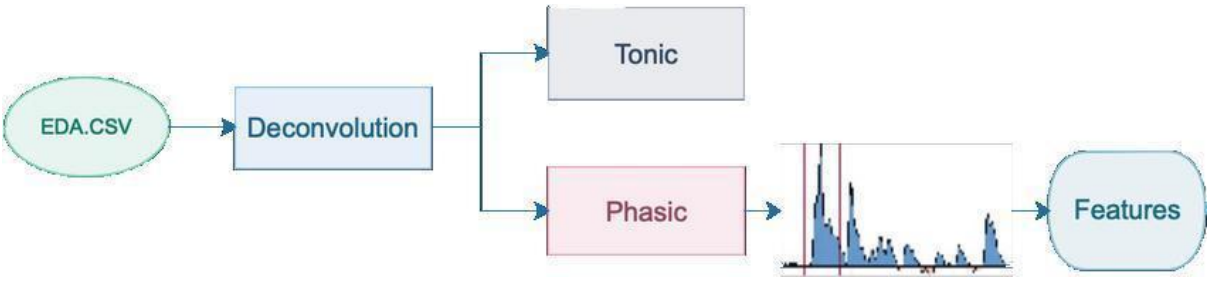
the SCR which as mentioned is a physiological variable under the phasic component of the SC signal is what commonly provides the features used in EDA-based studies to provide valuable information for many scientific research fields as Posada-Quintero & Chon (2020) writes. Benedek & Kaernbach (2010), Ellaway et al. (2010), Greco et al. (2016), and again Posada-Quintero & Chon (2020) further reinforce this concept since SCR has been used to assess pain, schizophrenia, peripheral neuropathy, and more importantly stress, since the pathways of the sympathetic nervous system activity are expressed as a change in the electrical properties in SC.

Once deconvolved the SCR signal from here can be used to extract several features to perform numerous tasks of arousal classification using Machine Learning/Deep Learning models, more specifically those that involve detecting stress. The systematic review by Sanchez-Reolid et al. (2022) highlights this as the 5 most common categories of features found in EDA-based studies that used the signal as basis wherein to extract features were those relating to time, frequency, statistics, morphology (which involves the quantification of the shape of the signal), and time-frequency (which characterize the signal in time and frequency domains simultaneously).

An earlier study conducted by Sanchez-Reolid et al. (2019) demonstrates this by their use of Support Vector Machines in measuring an individual's stress response using electrodermal activity. Their experiment began by recruiting a total number of sixteen participants and measuring their EDA, Heart Rate or Inter-Beat (HR), Skin Temperature (TMP), Blood Volume Pulse (BVP) using the Empatica E4 wristband which measures such variables in an individual. A group of images from the International Affective Picture System (IAPS) image library were then shown to the participants which is and was used to evoke a feeling of calmness or stress in the participant as Lang et al. (2005) instructive manual has stated. How this works is that different batches of images that have similar values of valence, arousal and dominance are selected and shown to the

participants. The higher the arousal and the lower the valence (HL) was in an individual the more it indicated that he/she was in a state of calm. Conversely, when the arousal was lower and the valence was higher (LH), the more it indicated that the individual was in a state of stress. Furthermore, the system then starts to randomly repeat each of the image blocks that elicit both HL and LH conditions, and when each block is finished, the Self-Assessment Manikin (SAM) questionnaire which is used to enable individuals to rate how pleasant/unpleasant (valence), calm/excited (arousal) and controlled (dominance) they felt when looking at the images, was answered. This was largely the annotation process as the signals that were recorded during image showing are synchronized with the events related to the occurrence of each batch of images. This synchronization will help to detect calmness and stress conditions as this study by Sanchez-Reolid et al. (2019) details.

Figure 13. Flowchart of feature extraction from EDA signals from Sanchez-Reolid et al. (2019)



In this study three categories of these kinds of features were used for their stress detection model (SVM): time-domain, morphological, and frequency based features. Under time-domain based features mean value (M), standard deviation (SD), maximum peak value (MA), minimum peak value (MI), dynamic range (DR), mean of the first order derivative (FM), standard deviation of the first order derivative (FSD), mean of the second order derivative (SM), and standard deviation of the second order derivative

(SSD) of the SCR signal were all calculated in windows/segments of both 4 and 20 seconds. Under morphology-based features arc length (AL), integral area (IN), normalized mean power (AP), perimeter and area ratio (IL), energy and perimeter ratio (EL), skewness (SK) and kurtosis (KU) of the SCR signal were all calculated in windows/segments of both 4 and 20 seconds. Lastly, under frequency-based features the fast Fourier transform (FFT) through bandwidths F1(0.1,0.2), F2(0.2,0.3) and F3(0.3,0.4) were calculated over the obtained SCR signal.

Once the features were obtained windows/segments of both 20 seconds and 4 seconds were given as input to the SVM model. Conclusively, this gave as Sanchez-Reolid et al. (2019) writes good results as their SVM models (which used fine-gaussian, linear, cubic kernel functions respectively) gave 5, 7, and 10-fold cross validation accuracies of 70.8%, 75%, and 70% respectively using a 20 second window. Similarly using a 4 second window their SVM models gave 5, 7, and 10-fold cross validation accuracies of 87.7% for all 3 kinds of folds; these models used quadratic, linear, fine gaussian, and cubic as their kernel functions.

In a similar study by Albertetti et al. (2021) where they also sought to use Physiological Signals like EDA for stress detection, their experiment akin to the study conducted by Sanchez-Reolid et al. (2019) also involved participants subjected to a stressful period that ranged from emotional arousal, intellectual efforts, physical exercises, and pain. And after each stressful period exposed to the participants a minimum 7 min of relaxation with some peaceful music and videos of nature followed thereafter. Once done, each participant was asked to report their perceived stress level which consisted of 4 options (1) I feel more relaxed, (2) No difference (3) I feel less relaxed, and (4) I feel more stressed. Hence, labeling and annotation of the signals were in large part achieved through the participants' relative changes of their perceived stress levels after being subjected to the numerous activities involving stressful conditions.

Once the signals were acquired, decomposition of the EDA signal to obtain the phasic component or the SCR signal and the tonic component or the SCL signal were applied, as the SCR as previously established is more closely associated with activation of the SNS when faced with stimuli such as stressors. Features extracted in this study consisted of time-based and frequency-based features, which calculated the SCR amplitude peak counts, the minimum value found in the section/segment, the maximum value in the section/segment, area under curve, mean of first order derivatives, mean of negative values of first order derivatives, and Hjorth features [19] and the energy of the signal, summation of FFT harmonics, area under curve of FFT, standard deviation of FFT, mean of FFT, and signal values in the frequency domain concerning each feature domain. As the researchers used Macro F1-Score and AUC (Area Under Curve) to gauge the performance of their proposed models these being Decision Trees, Recurrent Neural Networks (RNNs), Convolutional Recurrent Neural Networks (CRNNs), these said models over multiple windows had achieved the highest scores of 58%, 71%, and 65% respectively, showing that the use of an RNN based method achieved the highest Macro F1-Score in terms of detecting stress using EDA signals.

While the experiments conducted by Sanchez-Reolid et al. 2019 and Albertetti et al. (2021) has shown promising results in detecting stress using physiological signals such as EDA, such signals tend to still be corrupted with noise/artifacts, hence the challenges encountered by previous research and thereby the researchers, are all the more reasons to consider more reliable and efficient ways of detecting artifacts such as those that involve Machine Learning. Although Albertetti et al. (2021) argues that that other physiological signals such as HR (Heart Rate), HRV (Heart Rate Variability) in contrast, are less reliable in long terms studies since they are often heavily corrupted with the movement artifacts, they also consider EDA signals to be affected by this problem. Sanchez-Reolid et al. (2019) moreover emphasizes this as in using EDA

signals for stress detection, several limiting factors must be considered. This most importantly revolved around the quality of the data obtained. Because, in acquisition systems based on physiological signals, it is common that artifacts that damage or worsen the signal appear, hence, it is of utmost importance to avoid artifacts and events of disconnection of the electrodes with the skin. In addition Zhang et al. (2017) also emphasizes the disadvantages of these artifacts as motion artifact (MAt) degrades signals very quickly and makes them unusable. Taylor et al. (2015) moreover has detailed that continuous and unobtrusive measurement of EDA using wearable devices makes the signal collected vulnerable to several types of noise/artifacts. If these artifacts remain in the signal when it is analyzed they can easily be misinterpreted and skew the analysis; for example, they may be mistaken for a skin conductance response (SCR) (a physiological reaction that may indicate increased stress). Although methods of eliminating artifacts by deflecting the signal through various softening filters can be applied, this procedure as Chen et al. (2015) and Hernandez et al. (2011) argues, causes in most cases a loss of information in EDA signals. This is why it may be imperative that other methods of detecting and removing artifacts such as those that are ML based be considered, as the enablement to detect these problems in time will help to improve the rest of the process of stress detection. As this section established the promise of stress detection as well as the challenges it faces such as the presence of artifacts the next section will focus on the literature precisely aiming to address the detection of artifacts.

Synthesis of the Study

Although it has been established that utilizing machine learning techniques and approaches shows promising results, particularly in the field of artifact detection in EDA data, it still has its disadvantages when it comes to the process of manual feature

extraction and engineering, along with its inability to extract higher level features from signal data, as proven by the studies of Mohsen et. al. (2023), Jamshidzadeh et al. (2024), and Juneau et al. (2021). Their studies have shown that LSTMs were more suited to perform the task of feature extraction, opposing the aforementioned methods. On a similar note, while using DL methods for classifying labels from signal data, such as the hybridized LSTM-CNN for EDA signals by Llanes-Jurado et. al. (2023), presents great outcomes in automatic detection of artifacts, it still has its downside with its interpretability.

Researches by Mordensky et al. (2023) and Sánchez-Reolid et al. (2019) suggested that when faced with the simplicity of features deep learning algorithms performance may even more so be detrimentally affected than gain benefit. Hence, much simpler data may be better solved by traditional ML methods i.e. XGBoost, Logistic Regression, SVM, etc. than highly complex machine learning models such as the aforementioned CNN. In addition CNNs as mentioned has the concern of the lack of knowledge in the processes it has done to arrive at a prediction surrounding it, and as Lange et al. (2018) argues may cause ambiguity and a hesitance in relying on the predictions of CNNs, especially in critical applications like the medical domain.

Hence, to deal with these problems at hand, the researchers propose a hybridized LSTM-SVM approach. The LSTM model will serve as a mechanism that automatically extracts high level features from electrodermal activity signals, while SVM is responsible for addressing the problem of complexity and interpretability of CNN's predictions. The researchers will be using both LSTM and SVM to leverage both higher and order features, presupposing that it will surpass the performance of both standalone DL architectures and ML algorithm

CHAPTER 3

METHODOLOGY

Research Design

The researchers adopted an experimental and quantitative method to achieve the objective of automatically detecting artifacts from electrodermal activity (EDA) signals used for stress detection. The study employed hybridized Long Short-Term Memory (LSTM) and Support Vector Machine (SVM) models and compared their performance to existing models of previous researchers, Taylor et. al. (2015), Hossain et. al. (2022), and Llanes-Jurado et. al. (2023), specifically. The goal is to determine the efficiency of the hybrid LSTM-SVM model in improving the overall performance in terms of Accuracy, Precision, Recall, ROC-AUC, and F1-score, and the reliability of stress detection from EDA signals by effectively handling artifacts. To achieve this goal, the researchers presented the application of hybridized LSTM-SVM to address the concern with the non-stationary characteristic of EDA data: it can change over time. The study consisted of independent variables which are the EDABE dataset, the existing models by Taylor et. al. (2015), Hossain et. al. (2022), and Llanes-Jurado et. al. (2023), and the model, hybridized LSTM and SVM. Dependent variables consist of the performance of the baseline models and the LSTM-SVM model in terms of accuracy, precision, recall, and F1-score. The researchers expected that the results from the model either matched or excelled over the performance of the existing models.

Sources Of Data

This study utilized the Electrodermal Activity artifact correction BEnchmark (EDABE) dataset as its primary source of data. The EDABE dataset consists of electrodermal activity (EDA) recordings of hand and body motion artifacts. EDABE

includes 74.46 hours of EDA recordings, which are influenced by hand and body movement artifacts, from 43 subjects. The dataset is split into a training set with 33 subjects (56.27 hours) and a test set with 10 subjects (18.19 hours). The recordings were obtained using a Shimmer3 GSR+ Unit at a sampling rate of 128 Hz. Each file's name contains the user ID and the expert who corrected the signal. Additionally, the file includes the signal with the following variables:

Table 2

Signal components in EDABE dataset

time	Timestamp of the signal
rawdata	Raw data obtained by Shimmer3 GSR+ Unit
cleandata	Reconstructed clean signal performed by a human expert
binarytarget	Label of each sample as artifact or no artifact
signal_automatic	Automatic cleaning of the signal performed by the automatic pipeline
predArtifacts	Label predicted by the automatic cleaning pipeline
postProcessedPredArtifacts	Label predicted by the automatic cleaning pipeline after post processing

Llanes-Jurado et. et al. (2023) collected data during a virtual reality (VR) study intended at producing stress in participants by imitating frequent working scenarios in a virtual environment. To attain this goal, participants had to complete a variety of activities within the virtual settings. Initially, they were put in an office setting and had conversations with a virtual avatar about both work and personal issues. They were then led through a meeting scenario with five virtual avatars, where they actively engaged in decision-making procedures. Throughout the chat, participants may select from four alternatives presented at the bottom of the screen.

The last stage featured three separate minigames. The first game included putting out a simulated forest fire as rapidly as feasible. The second game required reconstructing a pipe so that water could flow in the least amount of time feasible. The third minigame required participants to navigate a maze while also performing simple arithmetic questions, with greater scores given for faster completion of both tasks. All three games required players to use both hands, which enhanced noise in the electrodermal activity (EDA) signal caused by stress and fast hand movements.

The VR scenario was done using an HTC Vive Pro Eye head-mounted display, which runs at a 90 Hz frame rate and has a resolution of 1440 × 1600 pixels per eye with a 110-degree field of vision. The EDA data were acquired at a sampling rate of 128 Hz using a Shimmer3 device and Consensys software. A total of 43 EDA signals were recorded with an average experiment duration of 104 ± 8 minutes, yielding 74.46 hours of data. The virtual world was built on the Unity3D platform.

Research Instrument

This study was implemented using Python as the programming language. Python is an interpreted, object-oriented, high-level programming language. It has a simple syntax that promotes code readability, which helps reduce program maintenance. This programming language provides modules and packages that aids in sustaining program modularity and code reuse (*What is Python?, n.d.*).

It also utilized the dataset, EDABE (Electrodermal Activity artifact correction BEnchmark), which comprises 74.46 hours of EDA recordings impacted by hand and body motion artifacts from 43 subjects. It is split into a training set with 33 subjects (56.27 hours) and a test set with 10 subjects (18.19 hours). The data was recorded using a Shimmer3 GSR+ Unit at a sampling rate of 128 Hz.

For the tools and technologies, the study used the following: TensorFlow, scikit-learn, and Department of Science and Technology – Advanced Science and Technology Institute's (DOST-ASTI) Computing and Archiving Research Environment (COARE) which provided High-Performance Computing (HPC) service.

TensorFlow is an open-source platform that is created for developing advanced machine learning applications. It is frequently used by data scientists, software developers, and educators. Initially developed by Google Brain Team with the intent of conducting ML and deep neural networks (DNNs) research (*What is TensorFlow?, n.d.*).

Scikit-learn (sklearn) is an open-source machine learning library for Python. It includes several classification, regression, and clustering algorithms and is designed to work together with the NumPy and SciPy, other libraries supported by Python. This library carries out data modeling and machine learning algorithms (Domino Data Lab, n.d.).

COARE, a service DOST-ASTI service, provides a platform to facilitate the efficient storage, analysis, and exchange of scientific data through three main services: HPC, Science Cloud, and Data Archiving. The HPC (High-Performance Computing) service is highly efficient for handling massive datasets that demand resource-intensive computations and fast processing. It has significant computational capabilities that produce faster and more exact results than traditional desktop PCs. HPC service was used for training the models to effectively handle and to process the data faster.

Finally, the resources such as Figma, GitHub, and ReactJS were utilized throughout, with regards to the overall user interface, storage, and collaboration for the implementation of the system.

System Architecture

Figure 14. System Architecture

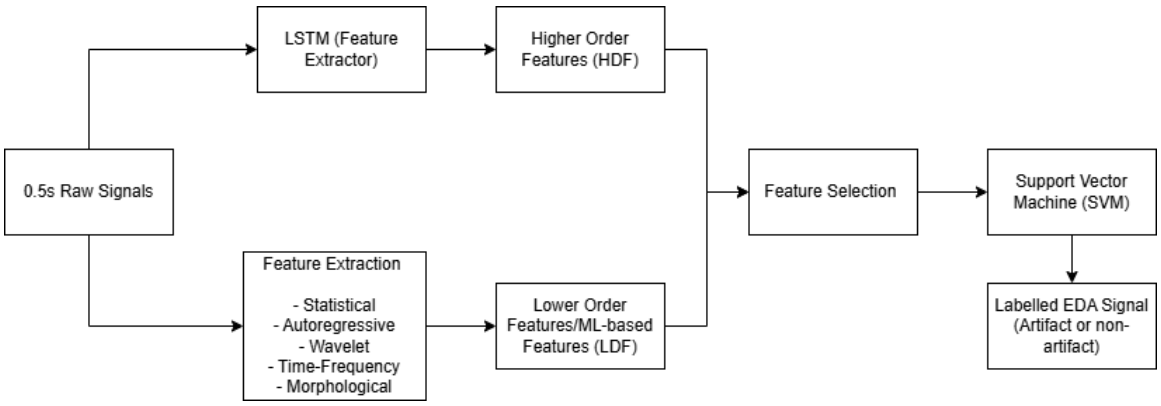
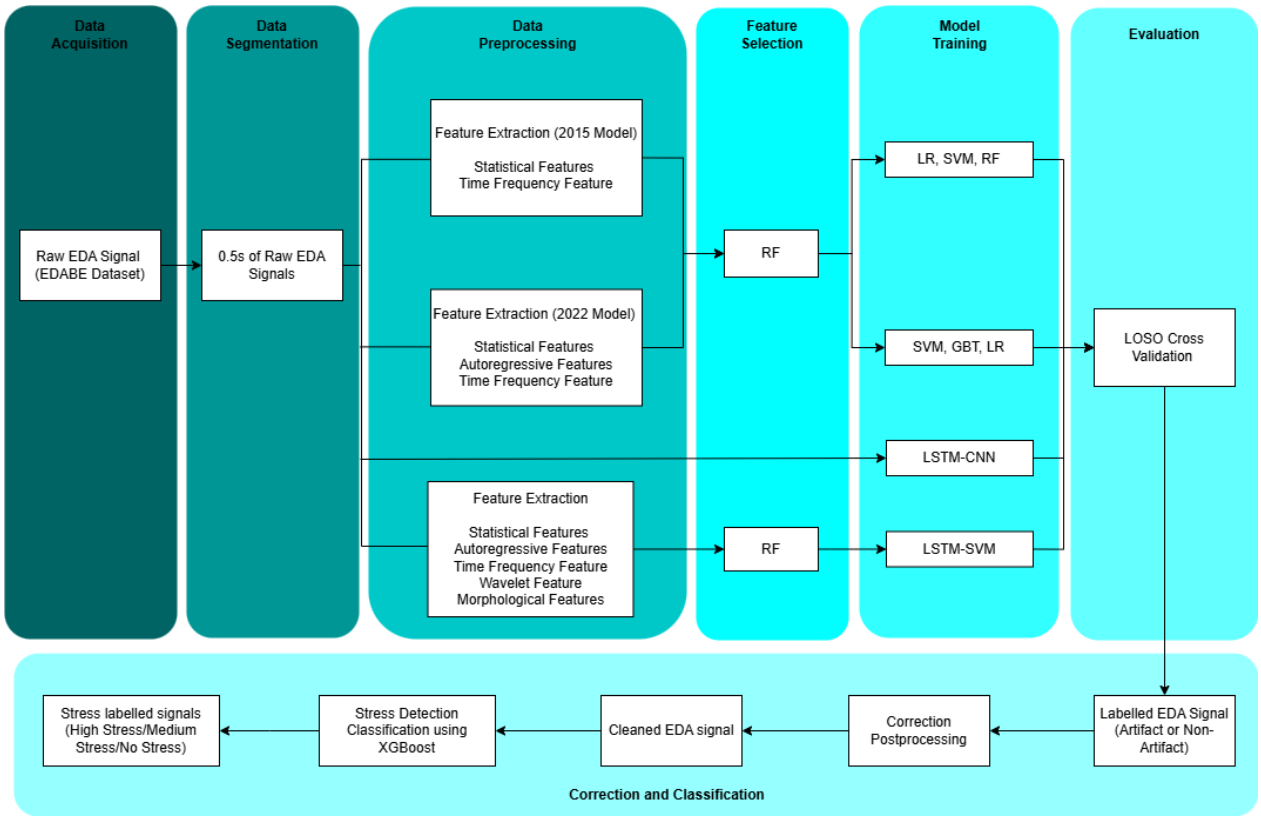


Figure 15. System Workflow



The system architecture for analyzing Electrodermal Activity (EDA) signals, typically used in stress detection, is divided into several stages. It starts with data acquisition from the EDABE dataset, which contains EDA signals from 42 subjects. The raw signals are segmented into 0.5-second windows for further processing. Two feature extraction techniques, based on prior research by Taylor et al. (2015) and Hossain et al. (2022), are applied to extract statistical, autoregressive, and time-frequency features. Relevant features are then selected using Recursive Feature Elimination (RFE) based on a Random Forest (RF) model. Multiple machine learning models, including Support Vector Machine (SVM), Logistic Regression (LR), Random Forest (RF) and Gradient Boosted Trees (GBT), are trained using the selected features. Additionally, hybrid models such as LSTM-CNN and LSTM-SVM are trained. The models undergo Leave-One-Subject-Out cross validation technique to search for the best hyper parameter configuration, and then classify the electrodermal activity signal into artifacts and non-artifacts. Thereafter, the labeled signals will be subjected to correction, which will result in clean EDA signals. These now clean signals will be further classified into 3 levels of stress, namely baseline or no perceived stress, medium stress, and high stress.

Focusing on hybrid models, particularly the LSTM-SVM approach, the researchers use not only using lower order hand crafted features i.e. statistical, autoregressive, time-frequency based as with previous studies but also integrating the resulting high level/order features resulting in the LSTM’s capability to capture temporal dependencies in EDA signals with the SVM's reability in classification, this architecture enhances the accuracy of artifact detection. Consequently, this hybridized method is crucial for improving the reliability of stress detection outcomes, ultimately contributing to more accurate assessments of physiological responses in stress-related studies.

Procedures

This study aims to assess the effectiveness of a proposed LSTM-SVM model by comparing it to various baseline models established in earlier studies. To set these baselines, the research replicates the approach of Llanes-Jurado et al. (2023), who themselves duplicated the methodologies of Taylor et al. (2015) and Hossain et al. (2022). The dataset utilized is the publicly accessible EDABE dataset, segmented into 0.5-second intervals for signal processing. Feature extraction entails calculating a range of statistical, time-frequency, and wavelet features, followed by the use of recursive feature elimination with a Random Forest classifier to minimize redundancy. In the case of Taylor et al. (2015), three classifiers Logistic Regression, Support Vector Machine (SVM), and Random Forest which are trained and fine-tuned using grid search and Leave-One-Subject-Out (LOSO) cross-validation, with the top-performing model serving as a baseline. Likewise, Hossain et al. (2022) implements additional techniques such as autoregressive modeling and variable frequency complex demodulation while introducing an extra classifier, Gradient Boosted Trees. The highest-performing models for each classifier are selected based on validation AUC and are used as comparative baselines.

Another baseline is the LSTM-CNN model put forth by Llanes-Jurado et al. (2023), which analyzes the same dataset using deep learning. It converts EDA signals into 3D matrixes for input into an LSTM-CNN structure comprising two LSTM layers with 16 neurons, four convolutional layers, batch normalization, dropout, max-pooling, and fully connected layers. This model is trained using the RMSProp optimizer at a learning rate of 5×10^{-5} and a batch size of 16, with the Dice-Sørensen coefficient applied as a loss function to address label imbalance. The training dataset contained 12.60% artifact data, and min-max scaling was employed on the signals.

The proposed LSTM-SVM model utilizes an LSTM for automatic high-level feature extraction from EDA signals while incorporating hand-crafted low-level features based on the studies of Taylor et al. (2015) and Hossain et al. (2022). This method substitutes the fully connected or Softmax activation layer typically found in RNNs with an SVM classifier, inspired by Agarap (2017) and Chen & Chen (2019). The architecture includes two stacked LSTM layers with 16 neurons each, where solely the last LSTM layer produces a final hidden state, which acts as the high-level feature input for the SVM. Furthermore, the model consists of hand-crafted statistical, time-frequency, wavelet, autoregressive, and morphological features, all normalized using the StandardScaler() function from the scikit-learn library. The SVM classifier is fine-tuned with hyperparameters such as C (1, 10, and 100) and Gamma (0.1, 0.5, and 1).

For assessment, the study adopts the LOSO cross-validation method to evaluate model generalization across various subjects. The dataset comprises 43 subjects, with 33 allocated for training and cross-validation, while the remaining 10 are reserved for testing. This strategy ensures that the model does not encounter signals from the same subject during both training and testing, tackling interpersonal variance and improving generalization. The training-validation split mitigates bias stemming from specific subject characteristics, as documented in earlier research by Di Lascio et al. (2018), Gjoreski et al. (2017), Saeb et al. (2016), and others. The final model selection hinges on cross-validation performance, and the model that performs best is utilized for testing and evaluation against the baseline models.

Ethical Considerations

In this conducted research of a developed hybridized LSTM-SVM for automatic artifact detection, the researchers ensured that the study provided ethical considerations

by ensuring consented, open-source datasets and following all agreements for data usage. The model's performance, as well as its limitations, are transparently presented and reported. The researchers recognize the positive nature and negative nature of technology and sustained practices against its exploitation. The comparison of the model's performance against the existing baseline models were conducted fairly, with the utilization of standardized metrics.

Data Analysis

To measure and to evaluate the performance of the proposed hybridized LSTM-SVM to automatically detect the artifacts found in an EDA signal, the researchers utilized the following performance metrics:

- ROC-AUC
- Precision
- F1-Score
- Recall
- Accuracy

ROC-AUC

Area Under Curve (AUC) is a proposed metric for evaluation by Llanes-Jurado et. al. (2023) to address the concern regarding the unbalanced classes in the dataset, as this metric provides a more reliable measure for future comparisons utilizing the dataset. This performance metric refers to the measure of the total area covered by a curve on a graph, which allows quantification of the extent region beneath the curve between points.

$$Area = \int_{a}^{b} y.dx$$

$$\Rightarrow Area = \int_a^b f(x).dx$$

$$\Rightarrow Area = [g(x)]_a^b$$

$$\Rightarrow Area = g(b) - g(a)$$

Accuracy

Accuracy refers to how close a measurement is to the true or accepted values. In this study, it is used to evaluate the performance of the models of Taylor et. al. (2015), Hossain et. al. (2022), and Llanes-Jurado et. al. (2023), and our proposed LSTM-SVM model, in terms of the quotient of correctly predicted instances to the total number of instances.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

- True Positives (TP) pertains to the instances wherein artifacts are correctly detected.
- True Negatives (TN) refers to the occurrence of correctly detecting non-artifacts.
- False Positives (FP) on the other hand pertains to incorrectly identified genuine signals as artifacts. These are non-artifacts that are predicted as an artifact.
- False Negatives (FN) are incorrectly defined artifacts as genuine EDA signals. These are artifacts that are predicted as a non-artifact.

Precision

Precision refers to how close measurements of the same item are to each other. It is defined as the ratio of true positive predictions to the total number of positive predictions, which consists of both true positives and false positives. It is also utilized by the researchers of this study to assess the relevance of positive predictions, especially when the dataset is imbalanced.

$$Precision = \frac{TP}{TP + FP}$$

F1-Score

To evaluate the given automatic artifact detection models in terms of their performance, F1-Score is also used as a metric. It is a machine learning evaluation metric that measures a model's accuracy. It combines the precision and recall scores of a model. By using F1-score, there is a single metric that includes both characteristics of precision and recall that can be used to compare different models easily.

$$\text{F1-Score} = 2\,x\,\frac{Precision * Recall}{Precision + Recall}$$

Recall

The researchers also used recall as a performance metric to evaluate the automatic artifact detection models of Taylor et. al. (2015), Hossain et. al. (2022), and Llanes-Jurado et. al. (2023), compared to our own LSTM-SVM model's performance. Also known as sensitivity or True Positive Rate, this parameter is utilized to calculate the percentage of results that are positive findings among all actual positive samples. Recall can be determined with the formula:

$$Recall = \frac{TP}{TP + FN}$$

By utilizing these performance metrics, the researchers aim to address the research question of the study, which looks for the difference of the automatic artifact detection by Taylor et. al. (2015), Hossain et. al. (2022), and Llanes-Jurado et. al. (2023) in comparison to our hybridized LSTM-SVM model.

With these metrics, a confusion matrix will be employed to see and compare the values of predicted and actual instances of artifacts and non-artifacts identified by the model. It follows the structure where the columns show the predicted values, while rows show the actual ones. Top-left displays the true negatives, while top-right shows the false

positives. On the bottom-left, the false negatives are displayed, while on the bottom-right is where the true positives are displayed.

Table 3

Confusion Matrix

	Actual Values	**Predicted Values**	
		Non-Artifact (0)	**Artifact (1)**
Subjects	**Non-Artifact (0)**	True Negative (TN)	False Positive (FP)
	Artifact (1)	False Negative (FN)	True Positive (TP)

Table 4

Confusion Matrix with the Subjects

Subject	**Actual**	**Predicted**	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*		
	Artifact (1)		
ajeric_expert2	*Non-Artifact (0)*		
	Artifact (1)		
ekamis_expert2	*Non-Artifact (0)*		
	Artifact (1)		
iguted_expert1	*Non-Artifact (0)*		
	Artifact (1)		
inefoh_expert1	*Non-Artifact (0)*		
	Artifact (1)		
otafeh_expert1	*Non-Artifact (0)*		
	Artifact (1)		
oxused_expert2	*Non-Artifact (0)*		

	Artifact (1)		
pqbqpr_expert2	*Non-Artifact (0)*		
	Artifact (1)		
uhepah_expert1	*Non-Artifact (0)*		
	Artifact (1)		
ukudab_expert2	*Non-Artifact (0)*		
	Artifact (1)		

Table 5

Comparison of Models in terms of ROC-AUC, Recall, Precision, and F1-Score and Accuracy

Artifact Detection Models	ROC-AUC	Recall	Precision	F1-Score	Accuracy
Proposed LSTM-SVM					
Logistic Regression (Taylor et. al., 2015)					
Random Forest (Taylor et. al., 2015)					
Support Vector Machine (Taylor et. al., 2015)					
Logistic Regression (Hossain et. al., 2022)					
Gradient Boost (Hossain et. al., 2022)					
Support Vector Machine (Hossain et. al., 2022)					
LSTM-CNN (Llanes-Jurado et. al., 2023)					

CHAPTER 4

RESULTS AND DISCUSSIONS

This chapter gives a detailed presentation, analysis, and interpretation of the results produced by the experimentation using the system developed in this study. Utilizing the EDABE dataset, the 43 subjects were divided into a training set with 33 subjects (56,27 hours) and a testing set of 10 subjects (18.19 hours). Subsequently, High-Performance Computing (HPC) service from DOST-ASTI's COARE was operated to train the models to efficiently handle the data faster.

The artifact detection models were evaluated using the performance metrics accuracy, precision, F1-score, recall, and ROC-AUC. The results gathered from the experimentation aim to address the research questions defined in the Statement of the Problem. These findings are comprehensively arranged and presented in a tabular format to ensure clarity and coherence.

1) Optimal Gamma and C Hyperparameters

This section provides a thorough presentation and analysis of the outcome that answers the first statement of the problem of the study, which is:

1. What would be the optimal Gamma and C hyperparameters for the SVM mechanism of the model such that it achieves higher evaluation metrics than using other Gamma and C values, to be used as baseline of comparison to other state of the art automatic artifact detection methods such as SVM, Logistic Regression, Random Forest, Gradient Boosted Tree, and LSTM-CNN?

Gamma and C hyperparameters define the behavior of the SVM model. It can ensure that the model can accurately classify segments of EDA signals. Hyperparameter

tuning was conducted using a range of values to the parameters, with C = [1, 10, 100] and Gamma = [0.1, 0.5, 1]. Each possible combination of these values was tested to find out which produced the best performance. Thereafter, the researchers handpicked the most effective configuration for the LSTM-SVM model. This is then compared to the results acquired from the hyperparameter configuration across different models.

This analysis provides an in-depth investigation on the performance of the different models and their optimized parameters in detecting and correcting artifacts in an EDA signal. This procedure ensures that the model is reliable and is capable of generating the best possible output, which emphasizes the significance of fine-tuning in machine learning procedures.

Table 6

Confusion Matrix for C = 100 and Gamma = 0.1

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	4454	2691
	Artifact (1)	1116	4257
ajeric_expert2	*Non-Artifact (0)*	10509	851
	Artifact (1)	752	649
ekamis_expert2	*Non-Artifact (0)*	11966	452
	Artifact (1)	857	334
iguted_expert1	*Non-Artifact (0)*	10957	688
	Artifact (1)	912	568
inefoh_expert1	*Non-Artifact (0)*	12062	445
	Artifact (1)	513	347
otafeh_expert1	*Non-Artifact (0)*	11118	193
	Artifact (1)	439	162
oxused_expert2	*Non-Artifact (0)*	9361	538
	Artifact (1)	1251	794
pqbqpr_expert2	*Non-Artifact (0)*	13773	541
	Artifact (1)	704	287
uhepah_expert1	*Non-Artifact (0)*	10525	573
	Artifact (1)	435	501
ukudab_expert2	*Non-Artifact (0)*	11415	1288
	Artifact (1)	899	685

Table 7

Average Performance of C = 100 and Gamma = 0.1

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.6959	0.6965	0.7786	0.7194	0.6959
ajeric_expert2	0.8744	0.8763	0.7836	0.8783	0.8744
ekamis_expert2	0.9038	0.8947	0.6836	0.8887	0.9038
iguted_expert1	0.8781	0.8737	0.7797	0.8701	0.8781
inefoh_expert1	0.9283	0.9270	0.8278	0.9257	0.9283
otafeh_expert1	0.9469	0.9404	0.7824	0.9365	0.9469
oxused_expert2	0.8502	0.8370	0.7587	0.8331	0.8502
pqbqpr_expert2	0.9187	0.9152	0.7198	0.9122	0.9187
uhepah_expert1	0.9162	0.9189	0.8276	0.9219	0.9162
ukudab_expert2	0.8469	0.8541	0.7385	0.8627	0.8469
	0.8759	**0.8734**	**0.7680**	**0.8749**	**0.8759**

Table 7 depicts the results of each test subject using the values C = 100 and Gamma = 0.1 in a confusion matrix. This achieved an accuracy of 0.8759, an F1-score of 0.8734, a ROC-AUC of 0.7680, a precision of 0.8749, and a recall of 0.8759. The configuration of these values has shown promising results in enhancing the SVM component of LSTM-SVM, to efficiently detect artifacts from non-artifacts in an EDA signal.

Table 8

Confusion Matrix for C = 100 and Gamma = 0.5

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5886	1259
	Artifact (1)	4347	1026
ajeric_expert2	*Non-Artifact (0)*	10639	721
	Artifact (1)	1134	267
ekamis_expert2	*Non-Artifact (0)*	12253	165
	Artifact (1)	1114	77
iguted_expert1	*Non-Artifact (0)*	11572	73
	Artifact (1)	1423	57
inefoh_expert1	*Non-Artifact (0)*	12433	74
	Artifact (1)	812	48
otafeh_expert1	*Non-Artifact (0)*	11281	30
	Artifact (1)	592	9
oxused_expert2	*Non-Artifact (0)*	9626	273
	Artifact (1)	1837	208
pqbqpr_expert2	*Non-Artifact (0)*	14219	95
	Artifact (1)	946	45
uhepah_expert1	*Non-Artifact (0)*	10980	118
	Artifact (1)	795	141
ukudab_expert2	*Non-Artifact (0)*	11700	1003
	Artifact (1)	1180	404

Table 9

Average Performance of C = 100 and Gamma = 0.5

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.5522	0.5017	0.6124	0.5210	0.5522
ajeric_expert2	0.8546	0.8434	0.7878	0.8341	0.8546
ekamis_expert2	0.9060	0.8766	0.6681	0.8643	0.9060
iguted_expert1	0.8860	0.8414	0.8174	0.8395	0.8860
inefoh_expert1	0.9337	0.9098	0.8612	0.9036	0.9337
otafeh_expert1	0.9478	0.9255	0.8458	0.9138	0.9478
oxused_expert2	0.8233	0.7751	0.8187	0.7700	0.8233
pqbqpr_expert2	0.9320	0.9074	0.7530	0.8977	0.9320
uhepah_expert1	0.9241	0.9038	0.8610	0.9023	0.9241
ukudab_expert2	0.8472	0.8432	0.7569	0.8395	0.8472
	0.8607	**0.8328**	**0.7782**	**0.8286**	**0.8607**

Table 9 outlines the performance of the test subjects that went through hyperparameter tuning using the values 100 and 0.5, C and Gamma respectively. This tuning indicated an overall accuracy of 0.8607, an F1-score of 0.8328, a ROC-AUC of 0.7782, a precision of 0.8286, and a recall of 0.8607.

Table 10

Confusion Matrix for C = 100 and Gamma = 1

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	7019	126
	Artifact (1)	5317	56
ajeric_expert2	*Non-Artifact (0)*	10998	362
	Artifact (1)	1359	42
ekamis_expert2	*Non-Artifact (0)*	12402	16
	Artifact (1)	1189	2
iguted_expert1	*Non-Artifact (0)*	11642	3
	Artifact (1)	1480	0
inefoh_expert1	*Non-Artifact (0)*	12503	4
	Artifact (1)	860	0
otafeh_expert1	*Non-Artifact (0)*	11308	3
	Artifact (1)	601	0
oxused_expert2	*Non-Artifact (0)*	9810	89
	Artifact (1)	2016	29
pqbqpr_expert2	*Non-Artifact (0)*	14305	9
	Artifact (1)	986	5
uhepah_expert1	*Non-Artifact (0)*	11091	7
	Artifact (1)	930	6
ukudab_expert2	*Non-Artifact (0)*	12354	349
	Artifact (1)	1523	61

Table 11

Average Performance of in C = 100 and Gamma = 1

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.5652	0.4200	0.5507	0.4568	0.5652
ajeric_expert2	0.8651	0.8307	0.7573	0.8037	0.8651
ekamis_expert2	0.9115	0.8705	0.6757	0.8424	0.9115
iguted_expert1	0.8870	0.8341	0.8060	0.7872	0.8870
inefoh_expert1	0.9354	0.9044	0.8530	0.8754	0.9354
otafeh_expert1	0.9493	0.9248	0.8397	0.9016	0.9493
oxused_expert2	0.8238	0.7531	0.8104	0.7296	0.8238
pqbqpr_expert2	0.9350	0.9045	0.7511	0.8981	0.9350
uhepah_expert1	0.9221	0.8858	0.8506	0.8868	0.9221
ukudab_expert2	0.8690	0.8333	0.7464	0.8080	0.8690
	0.8663	**0.8161**	**0.7641**	**0.7990**	**0.8663**

Table 11 depicts the confusion matrix of the results from the hyperparameters C=100 and Gamma=1. The evaluation metrics for this configuration indicate an overall accuracy of 0.8863, an F1-score of 0.8161 a ROC-AUC of 0.7641, a precision of 0.7990, and a recall of 0.8607.

Table 12

Confusion Matrix for C = 10 and Gamma = 0.1

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	4586	2559
	Artifact (1)	970	4403
ajeric_expert2	*Non-Artifact (0)*	10764	596
	Artifact (1)	796	605
ekamis_expert2	*Non-Artifact (0)*	12124	294
	Artifact (1)	923	268
iguted_expert1	*Non-Artifact (0)*	11097	548
	Artifact (1)	931	549
inefoh_expert1	*Non-Artifact (0)*	12181	326
	Artifact (1)	522	338
otafeh_expert1	*Non-Artifact (0)*	11184	127
	Artifact (1)	454	147
oxused_expert2	*Non-Artifact (0)*	9523	376
	Artifact (1)	1270	775
pqbqpr_expert2	*Non-Artifact (0)*	13899	415
	Artifact (1)	737	254
uhepah_expert1	*Non-Artifact (0)*	10709	389
	Artifact (1)	461	475
ukudab_expert2	*Non-Artifact (0)*	11724	979
	Artifact (1)	951	633

Table 13

Average Performance of C = 10 and Gamma = 0.1

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7181	0.7186	0.7951	0.7426	0.7181
ajeric_expert2	0.8909	0.8872	0.8015	0.8842	0.8909
ekamis_expert2	0.9106	0.8956	0.6707	0.8897	0.9106
iguted_expert1	0.8873	0.8799	0.8157	0.8750	0.8873
inefoh_expert1	0.9366	0.9327	0.8497	0.9300	0.9366
otafeh_expert1	0.9512	0.9425	0.8006	0.9396	0.9512
oxused_expert2	0.8622	0.8459	0.7816	0.8465	0.8622
pqbqpr_expert2	0.9247	0.9178	0.7087	0.9127	0.9247
uhepah_expert1	0.9294	0.9281	0.8515	0.9269	0.9294
ukudab_expert2	0.8649	0.8654	0.7722	0.8660	0.8649
	0.8876	**0.8814**	**0.7847**	**0.8813**	**0.8876**

Table 13 shows the average performance of the test subjects after going through hyperparameter tuning using the values C=10 and Gamma=0.1. This tuning presented an accuracy of 0.8876, an F1-score of 0.8814, a ROC-AUC of 0.7847, a precision of 0.8813, and a recall of 0.8876. The results present that the model can properly classify artifacts and non-artifacts in an EDA signal.

Table 14

Confusion Matrix for C = 10 and Gamma = 0.5

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	4812	2333
	Artifact (1)	1097	4276
ajeric_expert2	*Non-Artifact (0)*	10912	448
	Artifact (1)	862	539
ekamis_expert2	*Non-Artifact (0)*	12206	212
	Artifact (1)	996	195
iguted_expert1	*Non-Artifact (0)*	11261	384
	Artifact (1)	1011	469
inefoh_expert1	*Non-Artifact (0)*	12266	241
	Artifact (1)	584	276
otafeh_expert1	*Non-Artifact (0)*	11215	96
	Artifact (1)	453	148
oxused_expert2	*Non-Artifact (0)*	9516	383
	Artifact (1)	1322	723
pqbqpr_expert2	*Non-Artifact (0)*	13871	443
	Artifact (1)	779	212
uhepah_expert1	*Non-Artifact (0)*	10804	294
	Artifact (1)	487	449
ukudab_expert2	*Non-Artifact (0)*	11851	852
	Artifact (1)	1003	581

Table 15

Average Performance of C = 10 and Gamma = 0.5

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7260	0.7272	0.7844	0.7425	0.7260
ajeric_expert2	0.8973	0.8894	0.7732	0.8850	0.8973
ekamis_expert2	0.9112	0.8908	0.6072	0.8856	0.9112
iguted_expert1	0.8937	0.8808	0.7472	0.8761	0.8937
inefoh_expert1	0.9383	0.9310	0.7861	0.9275	0.9383
otafeh_expert1	0.9539	0.9445	0.7518	0.9433	0.9539
oxused_expert2	0.8573	0.8392	0.7836	0.8396	0.8573
pqbqpr_expert2	0.9202	0.9125	0.6525	0.9065	0.9202
uhepah_expert1	0.9351	0.9317	0.8287	0.9294	0.9351
ukudab_expert2	0.8702	0.8673	0.7309	0.8647	0.8702
	0.8903	**0.8814**	**0.7445**	**0.8800**	**0.8903**

Table 15 presents the average performance metrics of the test subjects evaluated using the model with hyperparameters C = 10 and Gamma = 0.5. This configuration achieved an overall accuracy of 0.8903, an F1-score of 0.8814, a ROC-AUC of 0.7445, a precision of 0.8800, and a recall of 0.8903.

Table 16

Confusion Matrix for C = 10 and Gamma = 1

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	7030	115
	Artifact (1)	5317	56
ajeric_expert2	*Non-Artifact (0)*	11112	248
	Artifact (1)	1356	45
ekamis_expert2	*Non-Artifact (0)*	12404	14
	Artifact (1)	1189	2
iguted_expert1	*Non-Artifact (0)*	11642	3
	Artifact (1)	1480	0
inefoh_expert1	*Non-Artifact (0)*	12502	5
	Artifact (1)	860	0
otafeh_expert1	*Non-Artifact (0)*	11308	3
	Artifact (1)	601	0
oxused_expert2	*Non-Artifact (0)*	9846	53
	Artifact (1)	2016	29
pqbqpr_expert2	*Non-Artifact (0)*	14307	7
	Artifact (1)	986	5
uhepah_expert1	*Non-Artifact (0)*	11091	7
	Artifact (1)	930	6
ukudab_expert2	*Non-Artifact (0)*	12484	219
	Artifact (1)	1522	62

Table 17

Average Performance of C = 10 and Gamma = 1

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.5661	0.4204	0.5520	0.4655	0.5661
ajeric_expert2	0.8743	0.8361	0.7504	0.8103	0.8743
ekamis_expert2	0.9116	0.8706	0.6814	0.8436	0.9116
iguted_expert1	0.8870	0.8341	0.8061	0.7872	0.8870
inefoh_expert1	0.9353	0.9044	0.8533	0.8754	0.9353
otafeh_expert1	0.9493	0.9248	0.8401	0.9016	0.9493
oxused_expert2	0.8268	0.7547	0.8187	0.7485	0.8268
pqbqpr_expert2	0.9351	0.9045	0.7522	0.9019	0.9351
uhepah_expert1	0.9221	0.8858	0.8528	0.8868	0.9221
ukudab_expert2	0.8781	0.8385	0.7568	0.8170	0.8781
	0.8686	**0.8174**	**0.7664**	**0.8038**	**0.8686**

Table 17 displays the average performance metrics of the test subjects evaluated using the model with hyperparameters C = 10 and Gamma = 1. This configuration achieved an overall accuracy of 0.8686, an F1-score of 0.8174, a ROC-AUC of 0.7664, a precision of 0.8038, and a recall of 0.8686. The results show a notable improvement in precision and ROC-AUC, indicating the model's ability to achieve high confidence in correct classifications.

Table 18

Confusion Matrix for C = 1 and Gamma = 0.1 Configuration

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	4374	2771
	Artifact (1)	726	4647
ajeric_expert2	*Non-Artifact (0)*	10953	407
	Artifact (1)	821	580
ekamis_expert2	*Non-Artifact (0)*	12266	152
	Artifact (1)	1017	174
iguted_expert1	*Non-Artifact (0)*	11381	264
	Artifact (1)	1054	426
inefoh_expert1	*Non-Artifact (0)*	12364	143
	Artifact (1)	632	228
otafeh_expert1	*Non-Artifact (0)*	11259	52
	Artifact (1)	498	103
oxused_expert2	*Non-Artifact (0)*	9594	305
	Artifact (1)	1291	754
pqbqpr_expert2	*Non-Artifact (0)*	14111	203
	Artifact (1)	792	199
uhepah_expert1	*Non-Artifact (0)*	10834	264
	Artifact (1)	467	469
ukudab_expert2	*Non-Artifact (0)*	11981	722
	Artifact (1)	1018	566

Table 19

Average Performance of C = 1 and Gamma = 0.1

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7206	0.7196	0.8074	0.7584	0.7206
ajeric_expert2	0.9038	0.8963	0.8285	0.8927	0.9038
ekamis_expert2	0.9141	0.8911	0.6430	0.8893	0.9141
iguted_expert1	0.8996	0.8829	0.8136	0.8817	0.8996
inefoh_expert1	0.9420	0.9311	0.8365	0.9297	0.9420
otafeh_expert1	0.9538	0.9407	0.7968	0.9429	0.9538
oxused_expert2	0.8664	0.8483	0.7858	0.8524	0.8664
pqbqpr_expert2	0.9350	0.9219	0.7002	0.9176	0.9350
uhepah_expert1	0.9393	0.9358	0.8450	0.9339	0.9393
ukudab_expert2	0.8782	0.8726	0.7794	0.8682	0.8782
	0.8953	**0.8840**	**0.7836**	**0.8867**	**0.8953**

Table 19 displays the average performance metrics of the test subjects evaluated using the model with hyperparameters C = 1 and Gamma = 0.1. This configuration achieved an overall accuracy of 0.8953, an F1-score of 0.8840, a ROC-AUC of 0.7836, a precision of 0.8867, and a recall of 0.8953. The results show a notable improvement in precision and ROC-AUC, indicating the model's ability to achieve high confidence in correct classifications.

Table 20

Confusion Matrix for C = 1 and Gamma = 0.5

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	6173	972
	Artifact (1)	4494	879
ajeric_expert2	*Non-Artifact (0)*	11180	180
	Artifact (1)	1213	188
ekamis_expert2	*Non-Artifact (0)*	12369	49
	Artifact (1)	1160	31
iguted_expert1	*Non-Artifact (0)*	11626	19
	Artifact (1)	1454	26
inefoh_expert1	*Non-Artifact (0)*	12494	13
	Artifact (1)	833	27
otafeh_expert1	*Non-Artifact (0)*	11311	0
	Artifact (1)	597	4
oxused_expert2	*Non-Artifact (0)*	9832	67
	Artifact (1)	1916	129
pqbqpr_expert2	*Non-Artifact (0)*	14290	24
	Artifact (1)	969	22
uhepah_expert1	*Non-Artifact (0)*	11057	41
	Artifact (1)	822	114
ukudab_expert2	*Non-Artifact (0)*	12322	381
	Artifact (1)	1283	301

Table 21

Average Performance of C = 1 and Gamma = 0.5

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.5633	0.5001	0.6243	0.5341	0.5633
ajeric_expert2	0.8908	0.8613	0.8396	0.8592	0.8908
ekamis_expert2	0.9112	0.8742	0.6903	0.8682	0.9112
iguted_expert1	0.8878	0.8382	0.8342	0.8538	0.8878
inefoh_expert1	0.9367	0.9089	0.8765	0.9206	0.9367
otafeh_expert1	0.9499	0.9258	0.8585	0.9524	0.9499
oxused_expert2	0.8340	0.7726	0.8362	0.8063	0.8340
pqbqpr_expert2	0.9351	0.9066	0.7533	0.9068	0.9351
uhepah_expert1	0.9283	0.9038	0.8795	0.9156	0.9283
ukudab_expert2	0.8835	0.8623	0.8024	0.8542	0.8835
	0.8721	**0.8354**	**0.7995**	**0.8471**	**0.8721**

Table 21 displays the average performance metrics of the test subjects evaluated using the model with hyperparameters C = 1 and Gamma = 0.5. This configuration achieved an overall accuracy of 0.8721, an F1-score of 0.8354, a ROC-AUC of 0.7958, a precision of 0.8471, and a recall of 0.8721. The results show a notable improvement in precision and ROC-AUC, indicating the model's ability to achieve high confidence in correct classifications.

Table 22

Confusion Matrix for C = 1 and Gamma = 1

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	7101	44
	Artifact (1)	5337	36
ajeric_expert2	*Non-Artifact (0)*	11337	23
	Artifact (1)	1382	19
ekamis_expert2	*Non-Artifact (0)*	12418	0
	Artifact (1)	1191	0
iguted_expert1	*Non-Artifact (0)*	11645	0
	Artifact (1)	1480	0
inefoh_expert1	*Non-Artifact (0)*	12507	0
	Artifact (1)	860	0
otafeh_expert1	*Non-Artifact (0)*	11311	0
	Artifact (1)	601	0
oxused_expert2	*Non-Artifact (0)*	9892	7
	Artifact (1)	2029	16
pqbqpr_expert2	*Non-Artifact (0)*	14314	0
	Artifact (1)	991	0
uhepah_expert1	*Non-Artifact (0)*	11094	4
	Artifact (1)	932	4
ukudab_expert2	*Non-Artifact (0)*	12655	48
	Artifact (1)	1558	26

Table 23

Average Performance of C = 1 and Gamma = 1

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.5701	0.4196	0.5631	0.5190	0.5701
ajeric_expert2	0.8899	0.8412	0.8036	0.8432	0.8899
ekamis_expert2	0.9125	0.8707	0.6858	0.8326	0.9125
iguted_expert1	0.8872	0.8342	0.8121	0.7872	0.8872
inefoh_expert1	0.9357	0.9046	0.8607	0.8755	0.9357
otafeh_expert1	0.9495	0.9250	0.8461	0.9016	0.9495
oxused_expert2	0.8295	0.7541	0.8386	0.8068	0.8295
pqbqpr_expert2	0.9352	0.9040	0.7533	0.8747	0.9352
uhepah_expert1	0.9222	0.8856	0.8592	0.8896	0.9222
ukudab_expert2	0.8876	0.8396	0.7956	0.8306	0.8876
	0.8720	**0.8178**	**0.7818**	**0.8161**	**0.8720**

Table 23 displays the average performance metrics of the test subjects evaluated using the model with hyperparameters C = 1 and Gamma = 1. This configuration achieved an overall accuracy of 0.8720, an F1-score of 0.8178, a ROC-AUC of 0.7818, a precision of 0.8161, and a recall of 0.8720.

Performance of C and Gamma Hyperparameter Tuning for LSTM-SVM

Table 24

Performance of C and Gamma Hyperparameter Tuning for LSTM-SVM

Hyperparameter (*C, Gamma*)	Accuracy	F1-Score	ROC-AUC	Precision	Recall
1, 0.1	0.8953	0.8840	0.7836	0.8867	0.8953
1, 0.5	0.8721	0.8354	0.7995	0.8471	0.8721
1, 1	0.8720	0.8178	0.7818	0.8161	0.8720
10, 0.1	0.8876	0.8814	0.7847	0.8813	0.8876
10, 0.5	0.8903	0.8814	0.7445	0.8800	0.8903
10, 1	0.8686	0.8174	0.7664	0.8038	0.8686
100, 0.1	0.8759	0.8734	0.7680	0.8749	0.8759
100, 0.5	0.8607	0.8328	0.7782	0.8286	0.8607

The hyperparameter tuning process involved experimenting with various combinations of C and Gamma values. Based on the evaluation metrics, the accuracy, precision, F1-score, ROC-AUC, and recall, the ideal configuration was handpicked by

the researchers and identified as C = 1 and Gamma = 0.1. This specific combination delivered the most favorable outcomes, outperforming other tested configurations.

This configuration was then compared to the hyperparameter tuning outcomes of existing models, SVM, RF, LR, GBT, and LSTM-CNN, in the second part of the analysis and discussion of the results. This comparative analysis gives a thorough evaluation of the LSTM-SVM's performance with different hyperparameter tuning using C and Gamma values and how it affects the results when used in detecting artifacts in an electrodermal activity signal.

2) LSTM-SVM Model Compared to other Existing Artifact Detection Models

This following provides a thorough presentation and analysis of the outcome that answers the second statement of the problem of the study, which is:

> 2. What would be the performance of the hybridized LSTM-SVM model compared to other existing models (SVM, Logistic Regression, Random Forest, GradBoost, LSTM-CNN) that automatically detect artifacts in EDA signals in terms of its (a) ROC-AUC, (b) F1-Score, (c) Precision, (d) Recall, and (e) Accuracy?

Each section includes parameter tuning for each model, illustrated with a confusion matrix, and a tabular comparison of the performance metrics for each test subject per model, including accuracy, precision, ROC-AUC, and recall. The best-performing model from the configurations was selected and compared to the artifact detection model, LSTM-SVM.

Gradient Boosted Tree (Hossain et. al., 2022)

To measure the efficiency and overall performance of the LSTM-SVM model in detecting artifacts, models from previous studies were replicated to compare the

performance of each. One of the models is the Gradient Boosted Tree (GBT) proposed by Hossain et al. (2022). Hyperparameter tuning was conducted for the GBT model using three key parameters: learning_rate, n_estimators, and max_depth.

The following tables represent the confusion matrix of every hyperparameter configuration per test subject, as well as the averaged performance of each in terms of accuracy, precision, recall, F1-score, and ROC-AUC. The parameter configuration that produces the best results for the Gradient Boosted Tree model will be handpicked and compared to the proposed study LSTM-SVM.

Table 25

Confusion Matrix of N Estimators = 600, Learning Rate = 0.01, and Max Depth = 5

Subject	**Actual**	**Predicted**	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5428	1727
	Artifact (1)	1132	4241
ajeric_expert2	*Non-Artifact (0)*	11094	276
	Artifact (1)	731	670
ekamis_expert2	*Non-Artifact (0)*	12345	83
	Artifact (1)	1005	186
iguted_expert1	*Non-Artifact (0)*	11562	93
	Artifact (1)	1063	417
inefoh_expert1	*Non-Artifact (0)*	12438	79
	Artifact (1)	637	223
otafeh_expert1	*Non-Artifact (0)*	11290	31
	Artifact (1)	486	115
oxused_expert2	*Non-Artifact (0)*	9745	165
	Artifact (1)	1077	967
pqbqpr_expert2	*Non-Artifact (0)*	14114	211
	Artifact (1)	680	310
uhepah_expert1	*Non-Artifact (0)*	10918	190
	Artifact (1)	549	387
ukudab_expert2	*Non-Artifact (0)*	12052	661
	Artifact (1)	849	735

Table 25 shows the confusion matrixes of the results produced by the hyperparameter tuning with n_estimators = 600, learning_rate = 0.01, and max_depth = 5. This configuration resulted in accuracy of 0.9099, F1-score of 0.8994, ROC-AUC of 0.8436, precision of 0.9030, and recall of 0.9099 as depicted in Table 26.

Table 26

Performance of N Estimators = 600, Learning Rate = 0.01, Max Depth = 5

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7718	0.7728	0.8572	0.7773	0.7718
ajeric_expert2	0.9211	0.9143	0.8499	0.9130	0.9211
ekamis_expert2	0.9201	0.8963	0.7602	0.9043	0.9201
iguted_expert1	0.9120	0.8923	0.8681	0.9047	0.9120
inefoh_expert1	0.9465	0.9342	0.8163	0.9376	0.9465
otafeh_expert1	0.9566	0.9439	0.8720	0.9501	0.9566
oxused_expert2	0.8961	0.8835	0.8976	0.8926	0.8961
pqbqpr_expert2	0.9418	0.9333	0.8204	0.9308	0.9418
uhepah_expert1	0.9386	0.9319	0.8854	0.9303	0.9386
ukudab_expert2	0.8944	0.8914	0.8091	0.8890	0.8944
	0.9099	**0.8994**	**0.8436**	**0.9030**	**0.9099**

Table 27

Confusion Matrix of N Estimators = 600, Learning Rate = 0.1, and Max Depth = 5

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5429	1726
	Artifact (1)	1099	4274
ajeric_expert2	*Non-Artifact (0)*	11075	295
	Artifact (1)	715	686
ekamis_expert2	*Non-Artifact (0)*	12310	118
	Artifact (1)	980	211
iguted_expert1	*Non-Artifact (0)*	11533	122
	Artifact (1)	1015	465
inefoh_expert1	*Non-Artifact (0)*	12407	110
	Artifact (1)	619	241
otafeh_expert1	*Non-Artifact (0)*	11269	52
	Artifact (1)	467	134
oxused_expert2	*Non-Artifact (0)*	9739	171
	Artifact (1)	1059	985
pqbqpr_expert2	*Non-Artifact (0)*	14076	249
	Artifact (1)	687	303
uhepah_expert1	*Non-Artifact (0)*	10903	205
	Artifact (1)	532	404
ukudab_expert2	*Non-Artifact (0)*	11951	762
	Artifact (1)	813	771

A configuration of 600, 0.1, and 5 for the n_estimators, learning_rate and max_depth, respectively, to optimize the GBT model for automatic artifact detection. This generated performance of 0.9095, 0.9002, 0.8380, 0.9023, and 0.9095 in accuracy, F1-score, ROC-AUC, precision, recall. A tabular representation of the predicted and actual cases is shown in Table 27, while Table 28 presents the comparison of the results per metric.

Table 28

Performance of N Estimators = 600, Learning Rate = 0.1, Max Depth = 5

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7745	0.7756	0.8533	0.7805	0.7745
ajeric_expert2	0.9209	0.9147	0.8661	0.9130	0.9209
ekamis_expert2	0.9194	0.8979	0.7104	0.9013	0.9194
iguted_expert1	0.9134	0.8963	0.8637	0.9048	0.9134
inefoh_expert1	0.9455	0.9346	0.8354	0.9354	0.9455
otafeh_expert1	0.9565	0.9454	0.8705	0.9481	0.9565
oxused_expert2	0.8971	0.8850	0.9022	0.8934	0.8971
pqbqpr_expert2	0.9389	0.9307	0.7923	0.9273	0.9389
uhepah_expert1	0.9388	0.9328	0.8855	0.9309	0.9388
ukudab_expert2	0.8898	0.8890	0.8009	0.8883	0.8898
	0.9095	**0.9002**	**0.8380**	**0.9023**	**0.9095**

Table 29

Confusion Matrix with N Estimators = 600, Learning Rate = 0.1, and Max Depth = 3

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5469	1686
	Artifact (1)	1091	4282
ajeric_expert2	*Non-Artifact (0)*	11087	283
	Artifact (1)	728	673
ekamis_expert2	*Non-Artifact (0)*	12337	91
	Artifact (1)	993	198
iguted_expert1	*Non-Artifact (0)*	11567	88
	Artifact (1)	1064	416
inefoh_expert1	*Non-Artifact (0)*	12436	81
	Artifact (1)	642	218
otafeh_expert1	*Non-Artifact (0)*	11288	33
	Artifact (1)	474	127
oxused_expert2	*Non-Artifact (0)*	9754	156
	Artifact (1)	1075	969
pqbqpr_expert2	*Non-Artifact (0)*	14137	188
	Artifact (1)	688	302
uhepah_expert1	*Non-Artifact (0)*	10908	200
	Artifact (1)	527	409
ukudab_expert2	*Non-Artifact (0)*	12022	691
	Artifact (1)	858	726

Table 29 represents the total number of actual and predicted instances in the GBT model that was optimized using n_estimator of 600, learning_rate of 0.1, and max_depth of 3. These instances were calculated using the formula of accuracy, F1-score, ROC-AUC, precision, and recall to quantify its performance in detecting artifacts in an EDA signal. The configuration resulted in 0.9106, 0.9003, 0.8375, 0.9039, and 0.9106, in the same order with its respective metric.

Table 30

Performance of N Estimators = 600, Learning Rate = 0.1, Max Depth = 3

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7783	0.7793	0.8574	0.7839	0.7783
ajeric_expert2	0.9208	0.9141	0.8653	0.9127	0.9208
ekamis_expert2	0.9204	0.8975	0.7085	0.9045	0.9204
iguted_expert1	0.9123	0.8925	0.8633	0.9056	0.9123
inefoh_expert1	0.9460	0.9335	0.8225	0.9366	0.9460
otafeh_expert1	0.9575	0.9456	0.8651	0.9513	0.9575
oxused_expert2	0.8970	0.8844	0.9007	0.8940	0.8970
pqbqpr_expert2	0.9428	0.9336	0.8064	0.9318	0.9428
uhepah_expert1	0.9396	0.9337	0.8874	0.9320	0.9396
ukudab_expert2	0.8917	0.8890	0.7986	0.8867	0.8917
	0.9106	0.9003	0.8375	0.9039	0.9106

Table 31

Confusion Matrix of N Estimators = 600, Learning Rate = 0.01, and Max Depth = 3

Subject	**Actual**	**Predicted**	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5440	1715
	Artifact (1)	1094	4279
ajeric_expert2	*Non-Artifact (0)*	11099	271
	Artifact (1)	750	651
ekamis_expert2	*Non-Artifact (0)*	12347	81
	Artifact (1)	1017	174
iguted_expert1	*Non-Artifact (0)*	11599	56
	Artifact (1)	1119	361
inefoh_expert1	*Non-Artifact (0)*	12451	66
	Artifact (1)	681	179
otafeh_expert1	*Non-Artifact (0)*	11294	27
	Artifact (1)	512	89
oxused_expert2	*Non-Artifact (0)*	9728	182
	Artifact (1)	1055	989
pqbqpr_expert2	*Non-Artifact (0)*	14163	162
	Artifact (1)	717	273
uhepah_expert1	*Non-Artifact (0)*	10999	109
	Artifact (1)	623	313
ukudab_expert2	*Non-Artifact (0)*	12104	609
	Artifact (1)	904	680

Table 32

Performance of N Estimators = 600, Learning Rate = 0.01, Max Depth = 3

Subject	**Accuracy**	**F1-Score**	**ROC-AUC**	**Precision**	**Recall**
afegip_expert1	0.7758	0.7768	0.8583	0.7817	0.7758
ajeric_expert2	0.9201	0.9126	0.8327	0.9114	0.9201
ekamis_expert2	0.9194	0.8947	0.7350	0.9028	0.9194
iguted_expert1	0.9105	0.8874	0.8670	0.9068	0.9105
inefoh_expert1	0.9442	0.9293	0.8087	0.9342	0.9442
otafeh_expert1	0.9548	0.9400	0.8699	0.9471	0.9548
oxused_expert2	0.8965	0.8847	0.8920	0.8923	0.8965
pqbqpr_expert2	0.9426	0.9320	0.8145	0.9309	0.9426
uhepah_expert1	0.9392	0.9284	0.8750	0.9305	0.9392
ukudab_expert2	0.8942	0.8893	0.8079	0.8859	0.8942
	0.9097	**0.8975**	**0.8361**	**0.9023**	**0.9097**

Another hyperparameter tuning using the combination of parameters n_estimator with the value of 600, learning_rate of 0.01, and max_depth with value of 3 was applied to find the one configuration that produces the best results. This tuning resulted in accuracy, F1-score, ROC-AUC, precision, and recall with values of 0.9097, 0.8975, 0.8361, 0.9023, and 0.9097, as shown in Table 32.

Table 33

Confusion Matrix of N Estimators = 600, Learning Rate = 0.01, and Max Depth = 10

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5444	1711
	Artifact (1)	1128	4245
ajeric_expert2	*Non-Artifact (0)*	11071	299
	Artifact (1)	723	678
ekamis_expert2	*Non-Artifact (0)*	12337	91
	Artifact (1)	967	224
iguted_expert1	*Non-Artifact (0)*	11540	115
	Artifact (1)	1015	465
inefoh_expert1	*Non-Artifact (0)*	12419	98
	Artifact (1)	619	241
otafeh_expert1	*Non-Artifact (0)*	11282	39
	Artifact (1)	469	132
oxused_expert2	*Non-Artifact (0)*	9727	183
	Artifact (1)	1052	992
pqbqpr_expert2	*Non-Artifact (0)*	14111	214
	Artifact (1)	693	297
uhepah_expert1	*Non-Artifact (0)*	10911	197
	Artifact (1)	511	425
ukudab_expert2	*Non-Artifact (0)*	12003	710
	Artifact (1)	815	769

Table 34

Performance of N Estimators = 600, Learning Rate = 0.01, Max Depth = 10

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7734	0.7744	0.8543	0.7788	0.7734
ajeric_expert2	0.9200	0.9136	0.8634	0.9118	0.9200
ekamis_expert2	0.9223	0.9010	0.7448	0.9084	0.9223
iguted_expert1	0.9140	0.8968	0.8703	0.9059	0.9140
inefoh_expert1	0.9464	0.9353	0.8352	0.9370	0.9464
otafeh_expert1	0.9574	0.9459	0.8822	0.9506	0.9574
oxused_expert2	0.8967	0.8849	0.9000	0.8925	0.8967
pqbqpr_expert2	0.9408	0.9318	0.8104	0.9291	0.9408
uhepah_expert1	0.9412	0.9357	0.8896	0.9341	0.9412
ukudab_expert2	0.8933	0.8917	0.7842	0.8903	0.8933
	0.9105	**0.9011**	**0.8434**	**0.9039**	**0.9105**

Table 33 shows the confusion matrixes of the results produced by the hyperparameter tuning with n_estimators = 600, learning_rate = 0.01, and max_depth = 10. This configuration resulted in accuracy of 0.9105, F1-score of 0.9011, ROC-AUC of 0.8434, precision of 0.9039, and recall of 0.9105 as depicted in Table 34.

Table 35

Confusion Matrix of N Estimators = 600, Learning Rate = 0.1, and Max Depth = 10

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5311	1844
	Artifact (1)	1071	4302
ajeric_expert2	*Non-Artifact (0)*	10976	394
	Artifact (1)	696	705
ekamis_expert2	*Non-Artifact (0)*	12248	180
	Artifact (1)	943	248
iguted_expert1	*Non-Artifact (0)*	11457	198
	Artifact (1)	980	500
inefoh_expert1	*Non-Artifact (0)*	12325	192
	Artifact (1)	593	267
otafeh_expert1	*Non-Artifact (0)*	11215	106
	Artifact (1)	447	154
oxused_expert2	*Non-Artifact (0)*	9682	228
	Artifact (1)	1036	1008
pqbqpr_expert2	*Non-Artifact (0)*	13988	337
	Artifact (1)	688	302
uhepah_expert1	*Non-Artifact (0)*	10802	306
	Artifact (1)	507	429
ukudab_expert2	*Non-Artifact (0)*	11923	790
	Artifact (1)	807	777

Table 36

Performance of N Estimators = 600, Learning Rate = 0.1, Max Depth = 10

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7673	0.7685	0.8502	0.7755	0.7673
ajeric_expert2	0.9147	0.9101	0.8607	0.9076	0.9147
ekamis_expert2	0.9175	0.8993	0.7494	0.8980	0.9175
iguted_expert1	0.9103	0.8957	0.8608	0.8981	0.9103
inefoh_expert1	0.9413	0.9329	0.8457	0.9302	0.9413
otafeh_expert1	0.9536	0.9448	0.8688	0.9431	0.9536
oxused_expert2	0.8943	0.8833	0.8954	0.8883	0.8943
pqbqpr_expert2	0.9331	0.9263	0.7977	0.9221	0.9331
uhepah_expert1	0.9325	0.9287	0.8799	0.9263	0.9325
ukudab_expert2	0.8883	0.8880	0.7889	0.8878	0.8883
	0.9053	**0.8978**	**0.8397**	**0.8977**	**0.9053**

A configuration of 600, 0.1, and 10 for the n_estimators, learning_rate and max_depth, respectively, to optimize the GBT model for automatic artifact detection. This generated performance of 0.9053, 0.8978, 0.8397, 0.8977, and 0.9053 in accuracy, F1-score, ROC-AUC, precision, recall. A tabular representation of the predicted and actual cases is shown in Table 35, while Table 36 presents the comparison of the results per metric.

Table 37

Confusion Matrix of N Estimators = 200, Learning Rate = 0.01, and Max Depth = 3

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5560	1595
	Artifact (1)	1337	4036
ajeric_expert2	*Non-Artifact (0)*	11167	203
	Artifact (1)	812	589
ekamis_expert2	*Non-Artifact (0)*	12403	25
	Artifact (1)	1093	98
iguted_expert1	*Non-Artifact (0)*	11629	26
	Artifact (1)	1263	217
inefoh_expert1	*Non-Artifact (0)*	12500	17
	Artifact (1)	776	84
otafeh_expert1	*Non-Artifact (0)*	11310	11
	Artifact (1)	563	38
oxused_expert2	*Non-Artifact (0)*	9745	165
	Artifact (1)	1078	966
pqbqpr_expert2	*Non-Artifact (0)*	14260	65
	Artifact (1)	820	170
uhepah_expert1	*Non-Artifact (0)*	11059	49
	Artifact (1)	733	203
ukudab_expert2	*Non-Artifact (0)*	12253	460
	Artifact (1)	954	630

Table 38

Performance of N Estimators = 200, Learning Rate = 0.01, Max Depth = 3

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7760	0.7666	0.8367	0.7678	0.7760
ajeric_expert2	0.9205	0.9105	0.8114	0.9115	0.9205
ekamis_expert2	0.9179	0.8862	0.6691	0.9083	0.9179
iguted_expert1	0.9019	0.8691	0.8615	0.9010	0.9019
inefoh_expert1	0.9407	0.9182	0.8011	0.9345	0.9407
otafeh_expert1	0.9519	0.9320	0.8590	0.9437	0.9519
oxused_expert2	0.8960	0.8834	0.8851	0.8925	0.8960
pqbqpr_expert2	0.9422	0.9251	0.7858	0.9313	0.9422
uhepah_expert1	0.9351	0.9173	0.8228	0.9276	0.9351
ukudab_expert2	0.9011	0.8929	0.8101	0.8890	0.9011
	0.9083	**0.8901**	**0.8143**	**0.9007**	**0.9083**

The hyperparameter tuning was performed using a combination of parameters: n_estimators set to 200, learning_rate set to 0.01, and max_depth set to 3, to identify the optimal configuration for achieving the best results. This configuration produced an accuracy of 0.9083, an F1-score of 0.8901, a ROC-AUC of 0.8143, a precision of 0.9007, and a recall of 0.9083, as summarized in Table 38.

Table 39

Confusion Matrix of N Estimators = 200, Learning Rate = 0.1, and Max Depth = 3

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5425	1730
	Artifact (1)	1076	4297
ajeric_expert2	*Non-Artifact (0)*	11070	300
	Artifact (1)	731	670
ekamis_expert2	*Non-Artifact (0)*	12323	105
	Artifact (1)	991	200
iguted_expert1	*Non-Artifact (0)*	11576	79
	Artifact (1)	1075	405
inefoh_expert1	*Non-Artifact (0)*	12450	67
	Artifact (1)	653	207
otafeh_expert1	*Non-Artifact (0)*	11294	27
	Artifact (1)	487	114
oxused_expert2	*Non-Artifact (0)*	9755	155
	Artifact (1)	1091	953
pqbqpr_expert2	*Non-Artifact (0)*	14144	181
	Artifact (1)	698	292
uhepah_expert1	*Non-Artifact (0)*	10931	177
	Artifact (1)	538	398
ukudab_expert2	*Non-Artifact (0)*	10041	672
	Artifact (1)	886	698

The hyperparameter tuning was performed using a combination of parameters: n_estimators set to 200, learning_rate set to 0.1, and max_depth set to 3, to identify the optimal configuration for achieving the best results. This configuration yielded an accuracy of 0.9100, an F1-score of 0.8993, a ROC-AUC of 0.8434, a precision of 0.9032, and a recall of 0.9100, as summarized in Table 40.

Table 40

Performance of N Estimators = 200, Learning Rate = 0.1, Max Depth = 3

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7760	0.7771	0.8601	0.7824	0.7760
ajeric_expert2	0.9193	0.9127	0.8561	0.9109	0.9193
ekamis_expert2	0.9195	0.8971	0.7684	0.9020	0.9195
iguted_expert1	0.9121	0.8917	0.8634	0.9062	0.9121
inefoh_expert1	0.9462	0.9329	0.8168	0.9376	0.9462
otafeh_expert1	0.9569	0.9440	0.8600	0.9511	0.9569
oxused_expert2	0.8958	0.8826	0.8979	0.8927	0.8958
pqbqpr_expert2	0.9426	0.9330	0.8212	0.9313	0.9426
uhepah_expert1	0.9406	0.9340	0.8880	0.9328	0.9406
ukudab_expert2	0.8910	0.8875	0.8017	0.8847	0.8910
	0.9100	**0.8993**	**0.8434**	**0.9032**	**0.9100**

Table 41

Confusion Matrix of N Estimators = 200, Learning Rate = 0.01, and Max Depth = 5

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5549	1606
	Artifact (1)	1284	4089
ajeric_expert2	*Non-Artifact (0)*	11159	211
	Artifact (1)	790	611
ekamis_expert2	*Non-Artifact (0)*	12386	42
	Artifact (1)	1063	128
iguted_expert1	*Non-Artifact (0)*	11620	35
	Artifact (1)	1193	287
inefoh_expert1	*Non-Artifact (0)*	12484	33
	Artifact (1)	732	128
otafeh_expert1	*Non-Artifact (0)*	11303	18
	Artifact (1)	538	63
oxused_expert2	*Non-Artifact (0)*	9755	155
	Artifact (1)	1126	918
pqbqpr_expert2	*Non-Artifact (0)*	14225	100
	Artifact (1)	759	231
uhepah_expert1	*Non-Artifact (0)*	11029	79
	Artifact (1)	656	280
ukudab_expert2	*Non-Artifact (0)*	12211	502
	Artifact (1)	916	668

Table 41 shows the confusion matrixes of the results produced by the hyperparameter tuning with n_estimators = 200, learning_rate = 0.01, and max_depth = 5. This configuration resulted in accuracy of 0.9089, F1-score of 0.8941, ROC-AUC of 0.8340, precision of 0.9021, and recall of 0.9089 as depicted in Table 42.

Table 42

Performance of N Estimators = 200, Learning Rate = 0.01, Max Depth = 5

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7693	0.7700	0.8494	0.7717	0.7693
ajeric_expert2	0.9216	0.9124	0.8232	0.9130	0.9216
ekamis_expert2	0.9189	0.8900	0.7419	0.9063	0.9189
iguted_expert1	0.9065	0.8787	0.8679	0.9051	0.9065
inefoh_expert1	0.9428	0.9240	0.8068	0.9350	0.9428
otafeh_expert1	0.9534	0.9361	0.8725	0.9457	0.9534
oxused_expert2	0.8928	0.8787	0.8925	0.8895	0.8928
pqbqpr_expert2	0.9439	0.9306	0.8141	0.9331	0.9439
uhepah_expert1	0.9390	0.9262	0.8678	0.9311	0.9390
ukudab_expert2	0.9008	0.8942	0.8038	0.8904	0.9008
	0.9089	**0.8941**	**0.8340**	**0.9021**	**0.9089**

Table 43

Confusion Matrix of N Estimators = 200, Learning Rate = 0.1, and Max Depth = 5

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5410	1745
	Artifact (1)	1111	4262
ajeric_expert2	*Non-Artifact (0)*	11084	286
	Artifact (1)	710	691
ekamis_expert2	*Non-Artifact (0)*	12340	88
	Artifact (1)	993	198
iguted_expert1	*Non-Artifact (0)*	11556	99
	Artifact (1)	1049	431
inefoh_expert1	*Non-Artifact (0)*	12424	93
	Artifact (1)	636	224
otafeh_expert1	*Non-Artifact (0)*	11280	41
	Artifact (1)	475	126
oxused_expert2	*Non-Artifact (0)*	9738	172
	Artifact (1)	1059	985
pqbqpr_expert2	*Non-Artifact (0)*	14102	223
	Artifact (1)	683	307
uhepah_expert1	*Non-Artifact (0)*	10900	208
	Artifact (1)	525	411
ukudab_expert2	*Non-Artifact (0)*	11992	721
	Artifact (1)	827	757

Table 43 presents the confusion matrixes generated from hyperparameter tuning using the parameters n_estimators=200, learning_rate =0.1, and max_depth=5. This setup achieved an accuracy of 0.9098, an F1-score of 0.9000, a ROC-AUC of 0.8372, a precision of 0.9031, and a recall of 0.9098, as shown in Table 44.

Table 44

Performance of N Estimators = 200, Learning Rate = 0.1, Max Depth = 5

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7720	0.7731	0.8547	0.7781	0.7720
ajeric_expert2	0.9220	0.9158	0.8639	0.9143	0.9220
ekamis_expert2	0.9206	0.8977	0.7094	0.9051	0.9206
iguted_expert1	0.9126	0.8937	0.8651	0.9051	0.9126
inefoh_expert1	0.9455	0.9335	0.8252	0.9356	0.9455
otafeh_expert1	0.9567	0.9449	0.8695	0.9493	0.9567
oxused_expert2	0.8970	0.8850	0.9002	0.8933	0.8970
pqbqpr_expert2	0.9408	0.9324	0.7927	0.9296	0.9408
uhepah_expert1	0.9391	0.9334	0.8878	0.9315	0.9391
ukudab_expert2	0.8917	0.8901	0.8039	0.8886	0.8917
	0.9098	**0.9000**	**0.8372**	**0.9031**	**0.9098**

Table 45

Confusion Matrix N Estimators = 200, Learning Rate = 0.01, and Max Depth = 10

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5636	1519
	Artifact (1)	1290	4083
ajeric_expert2	*Non-Artifact (0)*	11137	233
	Artifact (1)	800	601
ekamis_expert2	*Non-Artifact (0)*	12361	67
	Artifact (1)	1010	181
iguted_expert1	*Non-Artifact (0)*	11569	86
	Artifact (1)	1104	376
inefoh_expert1	*Non-Artifact (0)*	12456	61
	Artifact (1)	665	195
otafeh_expert1	*Non-Artifact (0)*	11291	30
	Artifact (1)	496	105
oxused_expert2	*Non-Artifact (0)*	9768	142
	Artifact (1)	1141	903
pqbqpr_expert2	*Non-Artifact (0)*	14186	139
	Artifact (1)	734	256
uhepah_expert1	*Non-Artifact (0)*	10980	128
	Artifact (1)	578	358
ukudab_expert2	*Non-Artifact (0)*	12180	533
	Artifact (1)	910	674

Table 45 represents the total number of actual and predicted instances in the GBT model that was optimized using n_estimator of 200, learning_rate of 0.01, and max_depth of 10. These instances were calculated using the formula of accuracy, F1-score, ROC-AUC, precision, and recall. The configuration resulted in 0.9103, 0.8976, 0.8386, 0.9026, and 0.9103, in the same order with its respective metric.

Table 46

Performance of N Estimators = 200, Learning Rate = 0.01, Max Depth = 10

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7758	0.7763	0.8537	0.7773	0.7758
ajeric_expert2	0.9191	0.9098	0.8534	0.9097	0.9191
ekamis_expert2	0.9209	0.8965	0.7294	0.9074	0.9209
iguted_expert1	0.9094	0.8876	0.8708	0.9017	0.9094
inefoh_expert1	0.9457	0.9317	0.8240	0.9373	0.9457
otafeh_expert1	0.9559	0.9424	0.8826	0.9488	0.9559
oxused_expert2	0.8927	0.8779	0.8978	0.8901	0.8927
pqbqpr_expert2	0.9430	0.9313	0.8089	0.9312	0.9430
uhepah_expert1	0.9414	0.9327	0.8867	0.9334	0.9414
ukudab_expert2	0.8991	0.8930	0.7791	0.8893	0.8991
	0.9103	**0.8979**	**0.8386**	**0.9026**	**0.9103**

Table 47

Confusion Matrix N Estimators = 200, Learning Rate = 0.1, and Max Depth = 10

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5374	1781
	Artifact (1)	1119	4254
ajeric_expert2	*Non-Artifact (0)*	11039	339
	Artifact (1)	691	710
ekamis_expert2	*Non-Artifact (0)*	12289	139
	Artifact (1)	955	236
iguted_expert1	*Non-Artifact (0)*	11489	166
	Artifact (1)	997	483
inefoh_expert1	*Non-Artifact (0)*	12380	137
	Artifact (1)	596	264
otafeh_expert1	*Non-Artifact (0)*	11247	74
	Artifact (1)	456	145
oxused_expert2	*Non-Artifact (0)*	9712	198
	Artifact (1)	1036	1008
pqbqpr_expert2	*Non-Artifact (0)*	14041	284
	Artifact (1)	693	297
uhepah_expert1	*Non-Artifact (0)*	10846	262
	Artifact (1)	499	437
ukudab_expert2	*Non-Artifact (0)*	11937	776
	Artifact (1)	809	775

Table 48

Performance of N Estimators = 200, Learning Rate = 0.1, Max Depth = 10

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7685	0.7696	0.8520	0.7750	0.7685
ajeric_expert2	0.9193	0.9142	0.8653	0.9121	0.9193
ekamis_expert2	0.9197	0.9000	0.7435	0.9018	0.9197
iguted_expert1	0.9115	0.8957	0.8658	0.9003	0.9115
inefoh_expert1	0.9452	0.9357	0.8373	0.9351	0.9452
otafeh_expert1	0.9555	0.9456	0.8777	0.9460	0.9555
oxused_expert2	0.8968	0.8856	0.8952	0.8920	0.8968
pqbqpr_expert2	0.9362	0.9284	0.7955	0.9244	0.9362
uhepah_expert1	0.9368	0.9326	0.8851	0.9303	0.9368
ukudab_expert2	0.8891	0.8886	0.7819	0.8881	0.8891
	0.9079	**0.8996**	**0.8399**	**0.9005**	**0.9079**

Another hyperparameter tuning using the combination of parameters n_estimator with the value of 200, learning_rate of 0.1, and max_depth with value of 10 was applied to find the one configuration that produces the best results. This tuning resulted in accuracy, F1-score, ROC-AUC, precision, and recall with values of 0.9079, 0.8996, 0.8399, 0.9005, and 0.9079, as shown in Table 48.

Table 49

Confusion Matrix of N Estimators = 400, Learning Rate = 0.01, and Max Depth = 3

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5432	1723
	Artifact (1)	1136	4237
ajeric_expert2	*Non-Artifact (0)*	11116	254
	Artifact (1)	758	643
ekamis_expert2	*Non-Artifact (0)*	12363	65
	Artifact (1)	1035	156
iguted_expert1	*Non-Artifact (0)*	11606	49
	Artifact (1)	1165	315
inefoh_expert1	*Non-Artifact (0)*	12464	53
	Artifact (1)	709	151
otafeh_expert1	*Non-Artifact (0)*	11302	19
	Artifact (1)	528	73
oxused_expert2	*Non-Artifact (0)*	9726	184
	Artifact (1)	1044	1000
pqbqpr_expert2	*Non-Artifact (0)*	14181	144
	Artifact (1)	738	252
uhepah_expert1	*Non-Artifact (0)*	11025	83
	Artifact (1)	665	271
ukudab_expert2	*Non-Artifact (0)*	12143	570
	Artifact (1)	913	671

Table 50

Performance of N Estimators = 400, Learning Rate = 0.01, Max Depth = 3

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7718	0.7728	0.8547	0.7772	0.7718
ajeric_expert2	0.9208	0.9129	0.8261	0.9121	0.9208
ekamis_expert2	0.9192	0.8930	0.7128	0.9038	0.9192
iguted_expert1	0.9076	0.8817	0.8680	0.9039	0.9076
inefoh_expert1	0.9430	0.9262	0.8054	0.9329	0.9430
otafeh_expert1	0.9541	0.9378	0.8699	0.9472	0.9541
oxused_expert2	0.8973	0.8857	0.8903	0.8931	0.8973
pqbqpr_expert2	0.9424	0.9307	0.8070	0.9302	0.9424
uhepah_expert1	0.9379	0.9247	0.8623	0.9293	0.9379
ukudab_expert2	0.8963	0.8907	0.8115	0.8869	0.8963
	0.9090	**0.8956**	**0.8308**	**0.9017**	**0.9090**

Table 50 shows the confusion matrixes of the results produced by the hyperparameter tuning with n_estimators = 400, learning_rate = 0.01, and max_depth = 3. This configuration resulted in accuracy of 0.9090, F1-score of 0.8956, ROC-AUC of 0.8308, precision of 0.9017, and recall of 0.9090 as shown in Table 50.

Table 51

Confusion Matrix N Estimators = 400, Learning Rate = 0.1, and Max Depth = 3

Subject	Actual	Predicted	
		Non-Artifact (07	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5449	1706
	Artifact (1)	1088	4285
ajeric_expert2	*Non-Artifact (0)*	11081	289
	Artifact (1)	729	672
ekamis_expert2	*Non-Artifact (0)*	12335	93
	Artifact (1)	989	202
iguted_expert1	*Non-Artifact (0)*	11574	81
	Artifact (1)	1067	413
inefoh_expert1	*Non-Artifact (0)*	12445	72
	Artifact (1)	651	209
otafeh_expert1	*Non-Artifact (0)*	11290	31
	Artifact (1)	482	119
oxused_expert2	*Non-Artifact (0)*	9750	160
	Artifact (1)	1074	970
pqbqpr_expert2	*Non-Artifact (0)*	14139	186
	Artifact (1)	694	296
uhepah_expert1	*Non-Artifact (0)*	10904	204
	Artifact (1)	524	412
ukudab_expert2	*Non-Artifact (0)*	12024	689
	Artifact (1)	866	718

Table 52

Performance of N Estimators = 400, Learning Rate = 0.1, Max Depth = 3

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7770	0.7780	0.8583	0.7828	0.7770
ajeric_expert2	0.9203	0.9136	0.8640	0.9121	0.9203
ekamis_expert2	0.9206	0.8980	0.7256	0.9047	0.9206
iguted_expert1	0.9126	0.8925	0.8642	0.9066	0.9126
inefoh_expert1	0.9460	0.9328	0.8190	0.9370	0.9460
otafeh_expert1	0.9570	0.9445	0.8622	0.9507	0.9570
oxused_expert2	0.8968	0.8842	0.8999	0.8953	0.8968
pqbqpr_expert2	0.9425	0.9331	0.8116	0.9313	0.9425
uhepah_expert1	0.9396	0.9338	0.8891	0.9320	0.9396
ukudab_expert2	0.8912	0.8884	0.7993	0.8860	0.8912
	0.9104	**0.8999**	**0.8393**	**0.9039**	**0.9104**

Table 51 presents the confusion matrixes generated from hyperparameter tuning using the parameters n_estimators = 400, learning_rate = 0.1, and max_depth = 3. This setup achieved an accuracy of 0.9104, an F1-score of 0.8999, a ROC-AUC of 0.8393, a precision of 0.9039, and a recall of 0.9104, as shown in Table 52.

Table 53

Confusion Matrix of N Estimators = 400, Learning Rate = 0.01, and Max Depth = 5

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5472	1683
	Artifact (1)	1167	4206
ajeric_expert2	*Non-Artifact (0)*	11109	261
	Artifact (1)	749	652
ekamis_expert2	*Non-Artifact (0)*	12352	76
	Artifact (1)	1018	173
iguted_expert1	*Non-Artifact (0)*	11570	85
	Artifact (1)	1072	408
inefoh_expert1	*Non-Artifact (0)*	12440	77
	Artifact (1)	651	209
otafeh_expert1	*Non-Artifact (0)*	11292	29
	Artifact (1)	493	108
oxused_expert2	*Non-Artifact (0)*	9742	168
	Artifact (1)	1071	973
pqbqpr_expert2	*Non-Artifact (0)*	14143	182
	Artifact (1)	694	296
uhepah_expert1	*Non-Artifact (0)*	10952	156
	Artifact (1)	578	358
ukudab_expert2	*Non-Artifact (0)*	12085	628
	Artifact (1)	872	712

Table 54

Performance of N Estimators = 400, Learning Rate = 0.01, Max Depth = 5

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7725	0.7735	0.8563	0.7770	0.7725
ajeric_expert2	0.9209	0.9134	0.8446	0.9124	0.9209
ekamis_expert2	0.9197	0.8949	0.7469	0.9038	0.9197
iguted_expert1	0.9119	0.8917	0.8684	0.9053	0.9119
inefoh_expert1	0.9456	0.9326	0.8115	0.9362	0.9456
otafeh_expert1	0.9562	0.9429	0.8745	0.9496	0.9562
oxused_expert2	0.8964	0.8839	0.8976	0.8927	0.8964
pqbqpr_expert2	0.9428	0.9333	0.8191	0.9316	0.9428
uhepah_expert1	0.9391	0.9308	0.8816	0.9302	0.9391
ukudab_expert2	0.8951	0.8912	0.8082	0.8882	0.8951
	0.9100	**0.8988**	**0.8409**	**0.9027**	**0.9100**

A configuration of 400, 0.01, and 5 for the n_estimators, learning_rate and max_depth, respectively, to optimize the GBT model for automatic artifact detection. This generated performance of 0.9100, 0.8988, 0.8409, 0.9027, and 0.9100 in accuracy, F1-score, ROC-AUC, precision, recall. A tabular representation of the predicted and actual cases is shown in Table 53, while Table 54 presents the comparison of the results per metric.

Table 55

Confusion Matrix N Estimators = 400, Learning Rate = 0.1, and Max Depth = 5

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5429	1726
	Artifact (1)	1099	4274
ajeric_expert2	*Non-Artifact (0)*	11075	295
	Artifact (1)	715	686
ekamis_expert2	*Non-Artifact (0)*	12310	118
	Artifact (1)	980	211
iguted_expert1	*Non-Artifact (0)*	11533	122
	Artifact (1)	1015	465
inefoh_expert1	*Non-Artifact (0)*	12407	110
	Artifact (1)	619	241
otafeh_expert1	*Non-Artifact (0)*	11269	52
	Artifact (1)	467	134
oxused_expert2	*Non-Artifact (0)*	9739	171
	Artifact (1)	1059	985
pqbqpr_expert2	*Non-Artifact (0)*	14076	249
	Artifact (1)	687	303
uhepah_expert1	*Non-Artifact (0)*	10903	205
	Artifact (1)	532	404
ukudab_expert2	*Non-Artifact (0)*	11951	762
	Artifact (1)	813	771

Table 55 represents the total number of actual and predicted instances in the GBT model that was optimized using an n_estimator of 400, learning_rate of 0.1, and max_depth of 5. These instances were calculated using the formula of accuracy, F1-score, ROC-AUC, precision, and recall to quantify its performance in detecting artifacts in an EDA signal. The configuration resulted in 0.9098, 0.9003, 0.8375, 0.9029, and 0.9098, in the same order with its respective metric.

Table 56

Performance of N Estimators = 400, Learning Rate = 0.1, Max Depth = 5

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7740	0.7751	0.8537	0.7799	0.7740
ajeric_expert2	0.9213	0.9150	0.8657	0.9134	0.9213
ekamis_expert2	0.9206	0.8985	0.7090	0.9046	0.9206
iguted_expert1	0.9134	0.8955	0.8643	0.9054	0.9134
inefoh_expert1	0.9454	0.9340	0.8298	0.9352	0.9454
otafeh_expert1	0.9570	0.9458	0.8701	0.9493	0.9570
oxused_expert2	0.8968	0.8846	0.9016	0.8931	0.8968
pqbqpr_expert2	0.9401	0.9319	0.7926	0.9288	0.9401
uhepah_expert1	0.9391	0.9331	0.8860	0.9313	0.9391
ukudab_expert2	0.8904	0.8892	0.8017	0.8881	0.8904
	0.9098	**0.9003**	**0.8375**	**0.9029**	**0.9098**

Another hyperparameter tuning using the combination of parameters n_estimator with the value of 400, learning_rate of 0.01, and max_depth with value of 10 was applied to find the one configuration that produces the best results. This tuning resulted in accuracy, F1-score, ROC-AUC, precision, and recall with values of 0.9105, 0.9005, 0.8417, 0.9035, and 0.9105, as shown in Table 58.

Table 57

Confusion Matrix of N Estimators = 400, Learning Rate = 0.01, and Max Depth = 10

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5476	1679
	Artifact (1)	1163	4210
ajeric_expert2	*Non-Artifact (0)*	11083	287
	Artifact (1)	732	669
ekamis_expert2	*Non-Artifact (0)*	12341	87
	Artifact (1)	980	211
iguted_expert1	*Non-Artifact (0)*	11551	104
	Artifact (1)	1035	445
inefoh_expert1	*Non-Artifact (0)*	12433	84
	Artifact (1)	628	232
otafeh_expert1	*Non-Artifact (0)*	11283	38
	Artifact (1)	479	122
oxused_expert2	*Non-Artifact (0)*	9729	181
	Artifact (1)	1059	985
pqbqpr_expert2	*Non-Artifact (0)*	14133	192
	Artifact (1)	698	292
uhepah_expert1	*Non-Artifact (0)*	10921	187
	Artifact (1)	528	408
ukudab_expert2	*Non-Artifact (0)*	12034	679
	Artifact (1)	834	750

Table 58

Performance of N Estimators = 400, Learning Rate = 0.01, Max Depth = 10

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7731	0.7741	0.8537	0.7777	0.7731
ajeric_expert2	0.9202	0.9134	0.8602	0.9119	0.9202
ekamis_expert2	0.9217	0.8995	0.7389	0.9073	0.9217
iguted_expert1	0.9133	0.8951	0.8698	0.9057	0.9133
inefoh_expert1	0.9468	0.9350	0.8309	0.9379	0.9468
otafeh_expert1	0.9566	0.9445	0.8820	0.9494	0.9566
oxused_expert2	0.8963	0.8843	0.8986	0.8921	0.8963
pqbqpr_expert2	0.9419	0.9324	0.8100	0.9303	0.9419
uhepah_expert1	0.9406	0.9345	0.8892	0.9330	0.9406
ukudab_expert2	0.8942	0.8918	0.7839	0.8897	0.8942
	0.9105	**0.9005**	**0.8417**	**0.9035**	**0.9105**

Table 59

Confusion Matrix of N Estimators = 400, Learning Rate = 0.1, and Max Depth = 10

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5390	1765
	Artifact (1)	1094	4279
ajeric_expert2	*Non-Artifact (0)*	11014	356
	Artifact (1)	698	703
ekamis_expert2	*Non-Artifact (0)*	12252	176
	Artifact (1)	936	255
iguted_expert1	*Non-Artifact (0)*	11460	195
	Artifact (1)	992	488
inefoh_expert1	*Non-Artifact (0)*	12349	168
	Artifact (1)	600	260
otafeh_expert1	*Non-Artifact (0)*	11224	97
	Artifact (1)	445	156
oxused_expert2	*Non-Artifact (0)*	9712	198
	Artifact (1)	1027	1017
pqbqpr_expert2	*Non-Artifact (0)*	13989	336
	Artifact (1)	695	295
uhepah_expert1	*Non-Artifact (0)*	10817	291
	Artifact (1)	515	421
ukudab_expert2	*Non-Artifact (0)*	11925	788
	Artifact (1)	810	774

Table 60

Performance of N Estimators = 400, Learning Rate = 0.1, Max Depth = 10

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7718	0.7729	0.8521	0.7784	0.7718
ajeric_expert2	0.9175	0.9123	0.8653	0.9101	0.9175
ekamis_expert2	0.9183	0.9004	0.7468	0.8995	0.9183
iguted_expert1	0.9096	0.8945	0.8634	0.8971	0.9096
inefoh_expert1	0.9426	0.9334	0.8430	0.9314	0.9426
otafeh_expert1	0.9545	0.9456	0.8722	0.9445	0.9545
oxused_expert2	0.8975	0.8865	0.8955	0.8929	0.8975
pqbqpr_expert2	0.9327	0.9256	0.7991	0.9213	0.9327
uhepah_expert1	0.9331	0.9289	0.8798	0.9263	0.9331
ukudab_expert2	0.8882	0.8879	0.7911	0.8875	0.8882
	0.9066	**0.8988**	**0.8408**	**0.8989**	**0.9066**

Hyperparameter tuning was performed using a combination of parameters: n_estimators set to 400, learning_rate set to 0.1, and max_depth set to 10, to identify the optimal configuration for achieving the best results. This configuration yielded an accuracy of 0.9066, an F1-score of 0.8988, a ROC-AUC of 0.8408, a precision of 0.8989, and a recall of 0.9066, as summarized in Table 60.

To summarize, a hyperparameter tuning was employed to the GBT model used in artifact detection in EDA signals. Combinations of parameters n_estimators [200, 400, 600], learning_rate [0.01, 0.1], and max_depth [3, 5, 10] were experimented with to optimize the GBT model, to find the best one that could produce excellent results. The configuration that yielded the best results is n_estimators=600, learning_rate=0.01, max_depth=10, where it generated the accuracy of 0.9105, F1-score of 0.9011, ROC-AUC of 0.8434, precision of 0.9039, and recall of 0.9105. The GBT model with this configuration will then be compared to other existing models and to LSTM-SVM to study if there is a significant difference between the models used.

Linear Regression (Hossain et. al., 2022)

Similarly, Hossain et. al. (2022) utilized another machine learning model for automatic artifact detection, which is Linear Regression. In this study, it is also replicated by the researchers to learn if its performance has a significant difference over the LSTM-SVM model. To get the best results to be compared to the proposed model, a hyperparameter tuning was employed in the LR model with the use of C parameter, with values 0.01, 0.1, 1, 10, and 100. Each parameter was tested on ten (10) test subjects and the results from the experiment are organized in a tabular manner to present the details.

Table 61

Confusion Matrix of C = 0.01

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5510	1645
	Artifact (1)	1318	4055
ajeric_expert2	*Non-Artifact (0)*	11204	166
	Artifact (1)	830	571
ekamis_expert2	*Non-Artifact (0)*	12402	26
	Artifact (1)	1105	86
iguted_expert1	*Non-Artifact (0)*	11620	35
	Artifact (1)	1218	262
inefoh_expert1	*Non-Artifact (0)*	12460	57
	Artifact (1)	734	126
otafeh_expert1	*Non-Artifact (0)*	11296	25
	Artifact (1)	519	82
oxused_expert2	*Non-Artifact (0)*	9778	132
	Artifact (1)	1247	797
pqbqpr_expert2	*Non-Artifact (0)*	14155	170
	Artifact (1)	776	214
uhepah_expert1	*Non-Artifact (0)*	11039	69
	Artifact (1)	724	212
ukudab_expert2	*Non-Artifact (0)*	12320	393
	Artifact (1)	940	644

Table 62

Performance of C = 0.01

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7635	0.7642	0.8385	0.7660	0.7635
ajeric_expert2	0.9220	0.9110	0.7924	0.9139	0.9220
ekamis_expert2	0.9170	0.8843	0.7770	0.9050	0.9170
iguted_expert1	0.9046	0.8752	0.8395	0.9025	0.9046
inefoh_expert1	0.9409	0.9225	0.7791	0.9279	0.9409
otafeh_expert1	0.9544	0.9389	0.8155	0.9465	0.9544
oxused_expert2	0.8846	0.8661	0.8754	0.8819	0.8846
pqbqpr_expert2	0.9382	0.9252	0.8185	0.9228	0.9382
uhepah_expert1	0.9342	0.9174	0.8355	0.9242	0.9342
ukudab_expert2	0.9068	0.8980	0.8186	0.8950	0.9068
	0.9066	**0.8903**	**0.8190**	**0.8986**	**0.9066**

Table 61 shows the confusion matrixes of the results produced by the hyperparameter tuning with C = 0.01. This configuration resulted in accuracy of 0.9066, F1-score of 0.8903, ROC-AUC of 0.8190, precision of 0.8986, and recall of 0.9066 as depicted in Table 62.

Table 63

Confusion Matrix of C = 0.1

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5503	1652
	Artifact (1)	1313	4060
ajeric_expert2	*Non-Artifact (0)*	11205	165
	Artifact (1)	826	575
ekamis_expert2	*Non-Artifact (0)*	12404	24
	Artifact (1)	1098	93
iguted_expert1	*Non-Artifact (0)*	11617	38
	Artifact (1)	1206	274
inefoh_expert1	*Non-Artifact (0)*	12461	56
	Artifact (1)	728	132
otafeh_expert1	*Non-Artifact (0)*	11292	29
	Artifact (1)	513	88
oxused_expert2	*Non-Artifact (0)*	9776	134
	Artifact (1)	1242	802
pqbqpr_expert2	*Non-Artifact (0)*	14175	150
	Artifact (1)	776	214
uhepah_expert1	*Non-Artifact (0)*	11039	69
	Artifact (1)	718	218
ukudab_expert2	*Non-Artifact (0)*	12325	388
	Artifact (1)	941	643

Table 64

Performance of C = 0.1

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7633	0.7641	0.8382	0.7659	0.7633
ajeric_expert2	0.9224	0.9115	0.7926	0.9144	0.9224
ekamis_expert2	0.9176	0.8855	0.7784	0.9079	0.9176
iguted_expert1	0.9053	0.8767	0.8403	0.9028	0.9053
inefoh_expert1	0.9414	0.9234	0.7815	0.9292	0.9414
otafeh_expert1	0.9545	0.9397	0.8165	0.9462	0.9545
oxused_expert2	0.8849	0.8665	0.8765	0.8821	0.8849
pqbqpr_expert2	0.9395	0.9262	0.8183	0.9248	0.9395
uhepah_expert1	0.9347	0.9182	0.8361	0.9250	0.9347
ukudab_expert2	0.9070	0.8982	0.8200	0.8952	0.9070
	0.9071	**0.8910**	**0.8199**	**0.8994**	**0.9071**

Another hyperparameter tuning using the parameter C with value of 0.1 was applied to find the one configuration that produces the best results. This tuning resulted in accuracy, F1-score, ROC-AUC, precision, and recall with values of 0.9071, 0.8910, 0.8199, 0.8994, and 0.9071, as shown in Table 64.

Table 65

Confusion Matrix of C = 1

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5494	1661
	Artifact (1)	1310	4063
ajeric_expert2	*Non-Artifact (0)*	11205	165
	Artifact (1)	828	573
ekamis_expert2	*Non-Artifact (0)*	12402	26
	Artifact (1)	1100	91
iguted_expert1	*Non-Artifact (0)*	11619	36
	Artifact (1)	1201	279
inefoh_expert1	*Non-Artifact (0)*	12458	59
	Artifact (1)	727	133
otafeh_expert1	*Non-Artifact (0)*	11293	28
	Artifact (1)	512	89
oxused_expert2	*Non-Artifact (0)*	9776	134
	Artifact (1)	1239	805
pqbqpr_expert2	*Non-Artifact (0)*	14177	148
	Artifact (1)	770	220
uhepah_expert1	*Non-Artifact (0)*	11038	70
	Artifact (1)	718	218
ukudab_expert2	*Non-Artifact (0)*	12325	388
	Artifact (1)	941	643

Table 66

Performance of C = 1

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7629	0.7636	0.8386	0.7656	0.7629
ajeric_expert2	0.9222	0.9113	0.7930	0.9142	0.9222
ekamis_expert2	0.9173	0.8851	0.7796	0.9062	0.9173
iguted_expert1	0.9058	0.8775	0.8408	0.9040	0.9058
inefoh_expert1	0.9412	0.9234	0.7826	0.9287	0.9412
otafeh_expert1	0.9547	0.9399	0.8173	0.9468	0.9547
oxused_expert2	0.8851	0.8669	0.8772	0.8823	0.8851
pqbqpr_expert2	0.9401	0.9270	0.8200	0.9258	0.9401
uhepah_expert1	0.9346	0.9182	0.8364	0.9248	0.9346
ukudab_expert2	0.9070	0.8982	0.8207	0.8952	0.9070
	0.9071	**0.8911**	**0.8206**	**0.8994**	**0.9071**

Table 65 represents the total number of actual and predicted instances in the LR model that was optimized using C = 1. The instances were calculated using the formula of accuracy, F1-score, ROC-AUC, precision, and recall to quantify its performance in detecting artifacts in an EDA signal. The configuration resulted in 0.9071, 0.8911, 0.8206, 0.8994, and 0.9071, in the same order with its respective metric.

Table 67

Confusion Matrix of C = 10

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5501	1654
	Artifact (1)	1318	4055
ajeric_expert2	*Non-Artifact (0)*	11209	161
	Artifact (1)	828	573
ekamis_expert2	*Non-Artifact (0)*	12403	25
	Artifact (1)	1093	98
iguted_expert1	*Non-Artifact (0)*	11615	40
	Artifact (1)	1199	281
inefoh_expert1	*Non-Artifact (0)*	12460	57
	Artifact (1)	722	138
otafeh_expert1	*Non-Artifact (0)*	11291	30
	Artifact (1)	511	90
oxused_expert2	*Non-Artifact (0)*	9775	135
	Artifact (1)	1243	801
pqbqpr_expert2	*Non-Artifact (0)*	14186	139
	Artifact (1)	773	217
uhepah_expert1	*Non-Artifact (0)*	11039	69
	Artifact (1)	719	217
ukudab_expert2	*Non-Artifact (0)*	12327	386
	Artifact (1)	943	641

Table 68

Performance of C = 10

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7628	0.7635	0.8382	0.7654	0.7628
ajeric_expert2	0.9226	0.9116	0.7929	0.9147	0.9226
ekamis_expert2	0.9179	0.8862	0.7793	0.9083	0.9179
iguted_expert1	0.9057	0.8776	0.8403	0.9029	0.9057
inefoh_expert1	0.9418	0.9242	0.7821	0.9300	0.9418
otafeh_expert1	0.9546	0.9400	0.8161	0.9463	0.9546
oxused_expert2	0.8847	0.8663	0.8766	0.8818	0.8847
pqbqpr_expert2	0.9405	0.9271	0.8181	0.9264	0.9405
uhepah_expert1	0.9346	0.9181	0.8365	0.9249	0.9346
ukudab_expert2	0.9070	0.8981	0.8206	0.8952	0.9070
	0.9072	**0.8913**	**0.8201**	**0.8996**	**0.9072**

Hyperparameter tuning was performed using a parameter of C = 10 to identify the optimal configuration for achieving the best results. This configuration yielded an accuracy of 0.9072, an F1-score of 0.8913, a ROC-AUC of 0.8201, a precision of 0.8996, and a recall of 0.9072, as summarized in Table 68.

Table 69

Confusion Matrix ofC = 100

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5499	1656
	Artifact (1)	1312	4061
ajeric_expert2	*Non-Artifact (0)*	11205	165
	Artifact (1)	828	573
ekamis_expert2	*Non-Artifact (0)*	12402	26
	Artifact (1)	1102	89
iguted_expert1	*Non-Artifact (0)*	11617	38
	Artifact (1)	1201	279
inefoh_expert1	*Non-Artifact (0)*	12458	59
	Artifact (1)	726	134
otafeh_expert1	*Non-Artifact (0)*	11292	29
	Artifact (1)	511	90
oxused_expert2	*Non-Artifact (0)*	9776	134
	Artifact (1)	1240	804
pqbqpr_expert2	*Non-Artifact (0)*	14175	150
	Artifact (1)	770	220
uhepah_expert1	*Non-Artifact (0)*	11040	68
	Artifact (1)	721	215
ukudab_expert2	*Non-Artifact (0)*	12324	389
	Artifact (1)	939	645

Table 70

Performance of C = 100

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7631	0.7638	0.8383	0.7658	0.7631
ajeric_expert2	0.9222	0.9113	0.7926	0.9142	0.9222
ekamis_expert2	0.9172	0.8848	0.7790	0.9058	0.9172
iguted_expert1	0.9057	0.8774	0.8407	0.9034	0.9057
inefoh_expert1	0.9413	0.9235	0.7840	0.9288	0.9413
otafeh_expert1	0.9547	0.9400	0.8189	0.9466	0.9547
oxused_expert2	0.8851	0.8668	0.8770	0.8823	0.8851
pqbqpr_expert2	0.9399	0.9269	0.8189	0.9256	0.9399
uhepah_expert1	0.9345	0.9179	0.8361	0.9248	0.9345
ukudab_expert2	0.9071	0.8983	0.8205	0.8954	0.9071
	0.9071	**0.8911**	**0.8206**	**0.8993**	**0.9071**

Finally, Table 69 represents the last batch of confusion matrixes generated from hyperparameter tuning using the parameters C = 100. This setup achieved an accuracy of 0.9071, an F1-score of 0.8911, a ROC-AUC of 0.82106, a precision of 0.8993, and a recall of 0.9071, as shown in Table 70.

Support Vector Machine (Hossain et. al., 2022)

The third and final model used by Hossain et. al. (2022) is the Support Vector Machine or SVM. Identical to the hyperparameter tuning using C and Gamma in the proposed LSTM-SVM model, the SVM by the previous researchers utilized the same parameters to search for the configuration that produces the best result, which will then be compared to the proposed model of the study. The following displays the detailed and organized actual and predicted instances in a tabular manner, along with the performance of the configuration per test subject.

Table 71

Confusion Matrix C = 1 and Gamma = 0.001

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	6225	930
	Artifact (1)	1790	3583
ajeric_expert2	*Non-Artifact (0)*	11302	68
	Artifact (1)	1026	375
ekamis_expert2	*Non-Artifact (0)*	12399	29
	Artifact (1)	1101	90
iguted_expert1	*Non-Artifact (0)*	11609	46
	Artifact (1)	1192	288
inefoh_expert1	*Non-Artifact (0)*	12495	22
	Artifact (1)	750	110
otafeh_expert1	*Non-Artifact (0)*	11310	11
	Artifact (1)	530	71
oxused_expert2	*Non-Artifact (0)*	9816	94
	Artifact (1)	1323	721
pqbqpr_expert2	*Non-Artifact (0)*	14264	61
	Artifact (1)	799	191
uhepah_expert1	*Non-Artifact (0)*	11084	24
	Artifact (1)	766	170
ukudab_expert2	*Non-Artifact (0)*	12463	250
	Artifact (1)	1142	442

Table 71 shows the confusion matrixes of the results produced by the hyperparameter tuning with C = 1 and Gamma = 0.001. This configuration resulted in accuracy of 0.9079, F1-score of 0.8883, ROC-AUC of 0.8095, precision of 0.9019, and recall of 0.9079 as depicted in Table 72.

Table 72

Performance of C =1 and Gamma = 0.001

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7829	0.7796	0.8512	0.7841	0.7829
ajeric_expert2	0.9143	0.8938	0.8328	0.9091	0.9143
ekamis_expert2	0.9170	0.8848	0.7208	0.9043	0.9170
iguted_expert1	0.9057	0.8782	0.8121	0.9019	0.9057
inefoh_expert1	0.9423	0.9219	0.7938	0.9363	0.9423
otafeh_expert1	0.9546	0.9379	0.7713	0.9507	0.9546
oxused_expert2	0.8815	0.8594	0.8672	0.8818	0.8815
pqbqpr_expert2	0.9438	0.9279	0.7880	0.9347	0.9438
uhepah_expert1	0.9344	0.9139	0.8205	0.9308	0.9344
ukudab_expert2	0.9026	0.8852	0.8370	0.8853	0.9026
	0.9079	**0.8883**	**0.8095**	**0.9019**	**0.9079**

Table 73

Confusion Matrix C = 1 and Gamma = 0.01

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5880	1275
	Artifact (1)	1423	3950
ajeric_expert2	*Non-Artifact (0)*	11253	117
	Artifact (1)	903	498
ekamis_expert2	*Non-Artifact (0)*	12379	49
	Artifact (1)	1067	124
iguted_expert1	*Non-Artifact (0)*	11589	66
	Artifact (1)	1133	347
inefoh_expert1	*Non-Artifact (0)*	12465	52
	Artifact (1)	686	174
otafeh_expert1	*Non-Artifact (0)*	11301	20
	Artifact (1)	514	87
oxused_expert2	*Non-Artifact (0)*	9791	119
	Artifact (1)	1221	823
pqbqpr_expert2	*Non-Artifact (0)*	14182	143
	Artifact (1)	754	236
uhepah_expert1	*Non-Artifact (0)*	11058	50
	Artifact (1)	691	245
ukudab_expert2	*Non-Artifact (0)*	12365	348
	Artifact (1)	1021	563

A hyperparameter configuration of 1 and 0.01 for the C and Gamma parameters, respectively, to optimize the SVM model for automatic artifact detection. This generated performance of 0.9104, 0.8949, 0.7887, 0.9026, and 0.9104 in accuracy, F1-score, ROC-AUC, precision, recall. A tabular representation of the predicted and actual cases is shown in Table 73, while Table 74 presents the comparison of the results per metric.

Table 74

Performance of C = 1 and Gamma = 0.01

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7846	0.7842	0.8474	0.7841	0.7846
ajeric_expert2	0.9201	0.9059	0.8443	0.9130	0.9201
ekamis_expert2	0.9181	0.8891	0.6946	0.9028	0.9181
iguted_expert1	0.9087	0.8850	0.7445	0.9030	0.9087
inefoh_expert1	0.9448	0.9294	0.7579	0.9364	0.9448
otafeh_expert1	0.9552	0.9401	0.7545	0.9493	0.9552
oxused_expert2	0.8879	0.8702	0.8276	0.8865	0.8879
pqbqpr_expert2	0.9414	0.9290	0.7498	0.9284	0.9414
uhepah_expert1	0.9385	0.9233	0.8318	0.9326	0.9385
ukudab_expert2	0.9042	0.8926	0.8340	0.8899	0.9042
	0.9104	**0.8949**	**0.7887**	**0.9026**	**0.9104**

Table 75

Confusion Matrix C = 1 and Gamma = 0.1

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5569	1586
	Artifact (1)	1201	4172
ajeric_expert2	*Non-Artifact (0)*	11199	171
	Artifact (1)	804	597
ekamis_expert2	*Non-Artifact (0)*	12369	59
	Artifact (1)	1025	166
iguted_expert1	*Non-Artifact (0)*	11565	90
	Artifact (1)	1059	421
inefoh_expert1	*Non-Artifact (0)*	12346	171
	Artifact (1)	641	219
otafeh_expert1	*Non-Artifact (0)*	11297	24
	Artifact (1)	488	113
oxused_expert2	*Non-Artifact (0)*	9762	148
	Artifact (1)	1146	898
pqbqpr_expert2	*Non-Artifact (0)*	13909	416
	Artifact (1)	708	282
uhepah_expert1	*Non-Artifact (0)*	11032	76
	Artifact (1)	625	311
ukudab_expert2	*Non-Artifact (0)*	12227	486
	Artifact (1)	909	675

Table 76

Performance of C =1 and Gamma = 0.1

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7775	0.7783	0.8255	0.7806	0.7775
ajeric_expert2	0.9237	0.9135	0.8479	0.9159	0.9237
ekamis_expert2	0.9204	0.8947	0.6779	0.9072	0.9204
iguted_expert1	0.9125	0.8930	0.7188	0.9057	0.9125
inefoh_expert1	0.9393	0.9284	0.7589	0.9256	0.9393
otafeh_expert1	0.9571	0.9440	0.6696	0.9518	0.9571
oxused_expert2	0.8918	0.8769	0.8153	0.8887	0.8918
pqbqpr_expert2	0.9266	0.9206	0.7094	0.9162	0.9266
uhepah_expert1	0.9418	0.9304	0.8015	0.9353	0.9418
ukudab_expert2	0.9024	0.8957	0.8134	0.8921	0.9024
	0.9093	**0.8976**	**0.7638**	**0.9019**	**0.9093**

Another hyperparameter tuning using the combination of parameters C with value of 1 and Gamma of 0.1 was applied to find the one configuration that produces the best results. This tuning resulted in accuracy, F1-score, ROC-AUC, precision, and recall with values of 0.9093, 0.8976, 0.7683, 0.9019, and 0.9093, as shown in Table 76.

Table 77

Confusion Matrix of C = 1 and Gamma = 1

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5603	1552
	Artifact (1)	2214	3159
ajeric_expert2	*Non-Artifact (0)*	11155	215
	Artifact (1)	829	572
ekamis_expert2	*Non-Artifact (0)*	12383	45
	Artifact (1)	1073	118
iguted_expert1	*Non-Artifact (0)*	11614	41
	Artifact (1)	1363	117
inefoh_expert1	*Non-Artifact (0)*	12482	35
	Artifact (1)	822	38
otafeh_expert1	*Non-Artifact (0)*	11309	12
	Artifact (1)	588	13
oxused_expert2	*Non-Artifact (0)*	9799	111
	Artifact (1)	1555	489
pqbqpr_expert2	*Non-Artifact (0)*	14261	64
	Artifact (1)	941	49
uhepah_expert1	*Non-Artifact (0)*	11003	105
	Artifact (1)	601	335
ukudab_expert2	*Non-Artifact (0)*	12110	603
	Artifact (1)	871	713

Table 78

Performance of C = 1 and Gamma = 1

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.6994	0.6962	0.7598	0.6970	0.6994
ajeric_expert2	0.9183	0.9079	0.8264	0.9084	0.9183
ekamis_expert2	0.9179	0.8884	0.6950	0.9031	0.9179
iguted_expert1	0.8931	0.8528	0.8021	0.8776	0.8931
inefoh_expert1	0.9359	0.9099	0.7548	0.9114	0.9359
otafeh_expert1	0.9497	0.9271	0.7757	0.9289	0.9497
oxused_expert2	0.8606	0.8273	0.8150	0.8548	0.8606
pqbqpr_expert2	0.9344	0.9093	0.7531	0.9055	0.9344
uhepah_expert1	0.9414	0.9315	0.7439	0.9337	0.9414
ukudab_expert2	0.8969	0.8927	0.7881	0.8896	0.8969
	0.8948	**0.8743**	**0.7714**	**0.8810**	**0.8948**

Hyperparameter tuning was performed using a combination of parameters: C set to 1, and Gamma to 1, to identify the optimal configuration for achieving the best results. This configuration yielded an accuracy of 0.8948, an F1-score of 0.8743, a ROC-AUC of 0.7714, a precision of 0.8810, and a recall of 0.8948, as summarized in Table 78.

Table 79

Confusion Matrix of C = 10 and Gamma = 0.001

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	6014	1141
	Artifact (1)	1583	3790
ajeric_expert2	*Non-Artifact (0)*	11277	93
	Artifact (1)	951	450
ekamis_expert2	*Non-Artifact (0)*	12394	34
	Artifact (1)	1087	104
iguted_expert1	*Non-Artifact (0)*	11606	49
	Artifact (1)	1171	309
inefoh_expert1	*Non-Artifact (0)*	12484	33
	Artifact (1)	734	126
otafeh_expert1	*Non-Artifact (0)*	11308	13
	Artifact (1)	528	73
oxused_expert2	*Non-Artifact (0)*	9806	104
	Artifact (1)	1274	770
pqbqpr_expert2	*Non-Artifact (0)*	14253	72
	Artifact (1)	794	196
uhepah_expert1	*Non-Artifact (0)*	11076	32
	Artifact (1)	748	188
ukudab_expert2	*Non-Artifact (0)*	12419	294
	Artifact (1)	1061	523

Table 80

Performance of C = 10 and Gamma = 0.001

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7826	0.7812	0.8505	0.7818	0.7826
ajeric_expert2	0.9183	0.9017	0.8415	0.9120	0.9183
ekamis_expert2	0.9177	0.8868	0.7250	0.9049	0.9177
iguted_expert1	0.9071	0.8809	0.8024	0.9033	0.9071
inefoh_expert1	0.9427	0.9237	0.7714	0.9347	0.9427
otafeh_expert1	0.9546	0.9381	0.7610	0.9500	0.9546
oxused_expert2	0.8847	0.8648	0.8645	0.8843	0.8847
pqbqpr_expert2	0.9435	0.9279	0.7814	0.9333	0.9435
uhepah_expert1	0.9352	0.9162	0.8360	0.9304	0.9352
ukudab_expert2	0.9052	0.8915	0.8400	0.8901	0.9052
	0.9092	**0.8913**	**0.8074**	**0.9025**	**0.9092**

A configuration of 10 and 0.001 for the C and Gamma, respectively, was experimented with to optimize the SVM model for automatic artifact detection. This generated performance of 0.9092, 0.8913, 0.8074, 0.9025, and 0.9092 in accuracy, F1-score, ROC-AUC, precision, recall. A tabular representation of the predicted and actual cases is shown in Table 79, while Table 80 presents the comparison of the results per metric.

Table 81

Confusion Matrix C = 10 and Gamma = 0.01

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5780	1375
	Artifact (1)	1324	4049
ajeric_expert2	*Non-Artifact (0)*	11237	133
	Artifact (1)	870	531
ekamis_expert2	*Non-Artifact (0)*	12379	49
	Artifact (1)	1041	150
iguted_expert1	*Non-Artifact (0)*	11581	74
	Artifact (1)	1089	391
inefoh_expert1	*Non-Artifact (0)*	12447	70
	Artifact (1)	671	189
otafeh_expert1	*Non-Artifact (0)*	11299	22
	Artifact (1)	502	99
oxused_expert2	*Non-Artifact (0)*	9779	131
	Artifact (1)	1192	852
pqbqpr_expert2	*Non-Artifact (0)*	14139	186
	Artifact (1)	755	235
uhepah_expert1	*Non-Artifact (0)*	11037	71
	Artifact (1)	652	284
ukudab_expert2	*Non-Artifact (0)*	12344	369
	Artifact (1)	992	592

Table 81 presents the confusion matrix generated from hyperparameter tuning using the parameters C = 10 and Gamma = 0.01. This setup achieved an accuracy of 0.9111, an F1-score of 0.8974, a ROC-AUC of 0.7847, a precision of 0.9033, and a recall of 0.9111, as shown in Table 82.

Table 82

Performance of C = 10 and Gamma = 0.01

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7846	0.7847	0.8483	0.7848	0.7846
ajeric_expert2	0.9215	0.9087	0.8436	0.9141	0.9215
ekamis_expert2	0.9200	0.8929	0.6620	0.9077	0.9200
iguted_expert1	0.9115	0.8902	0.7444	0.9058	0.9115
inefoh_expert1	0.9446	0.9304	0.7858	0.9348	0.9446
otafeh_expert1	0.9560	0.9419	0.7595	0.9504	0.9560
oxused_expert2	0.8893	0.8727	0.8247	0.8871	0.8893
pqbqpr_expert2	0.9386	0.9268	0.7384	0.9240	0.9386
uhepah_expert1	0.9400	0.9272	0.8100	0.9330	0.9400
ukudab_expert2	0.9048	0.8943	0.8308	0.8913	0.9048
	0.9111	**0.8970**	**0.7847**	**0.9033**	**0.9111**

Table 83

Confusion Matrix of C = 10 and Gamma = 0.1

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5542	1613
	Artifact (1)	1178	4195
ajeric_expert2	*Non-Artifact (0)*	11183	187
	Artifact (1)	786	615
ekamis_expert2	*Non-Artifact (0)*	12364	64
	Artifact (1)	1009	182
iguted_expert1	*Non-Artifact (0)*	11532	123
	Artifact (1)	1032	448
inefoh_expert1	*Non-Artifact (0)*	12260	257
	Artifact (1)	618	242
otafeh_expert1	*Non-Artifact (0)*	11289	32
	Artifact (1)	478	123
oxused_expert2	*Non-Artifact (0)*	9770	140
	Artifact (1)	1150	894
pqbqpr_expert2	*Non-Artifact (0)*	13829	496
	Artifact (1)	706	284
uhepah_expert1	*Non-Artifact (0)*	11024	84
	Artifact (1)	604	332
ukudab_expert2	*Non-Artifact (0)*	12195	518
	Artifact (1)	894	690

Table 84

Performance of C = 10 and Gamma = 0.1

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7772	0.7781	0.8312	0.7808	0.7772
ajeric_expert2	0.9238	0.9144	0.8516	0.9160	0.9238
ekamis_expert2	0.9212	0.8967	0.6477	0.9084	0.9212
iguted_expert1	0.9121	0.8942	0.7104	0.9028	0.9121
inefoh_expert1	0.9346	0.9264	0.7535	0.9220	0.9346
otafeh_expert1	0.9572	0.9450	0.6649	0.9510	0.9572
oxused_expert2	0.8921	0.8770	0.8045	0.8895	0.8921
pqbqpr_expert2	0.9215	0.9171	0.6992	0.9135	0.9215
uhepah_expert1	0.9429	0.9325	0.7884	0.9364	0.9429
ukudab_expert2	0.9012	0.8953	0.8159	0.8918	0.9012
	0.9084	**0.8977**	**0.7567**	**0.9012**	**0.9084**

Table 83 shows the confusion matrixes of the results produced by the hyperparameter tuning with C = 10 and Gamma 0.1. This configuration resulted in accuracy of 0.9084, F1-score of 0.8977, ROC-AUC of 0.7567, precision of 0.9012, and recall of 0.9084 as depicted in Table 84.

Table 85

Confusion Matrix of C = 100 and Gamma = 0.001

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5918	1237
	Artifact (1)	1475	3898
ajeric_expert2	*Non-Artifact (0)*	11263	107
	Artifact (1)	926	475
ekamis_expert2	*Non-Artifact (0)*	12388	40
	Artifact (1)	1068	123
iguted_expert1	*Non-Artifact (0)*	11597	58
	Artifact (1)	1136	344
inefoh_expert1	*Non-Artifact (0)*	12475	42
	Artifact (1)	711	149
otafeh_expert1	*Non-Artifact (0)*	11302	19
	Artifact (1)	520	81
oxused_expert2	*Non-Artifact (0)*	9799	111
	Artifact (1)	1236	808
pqbqpr_expert2	*Non-Artifact (0)*	14224	101
	Artifact (1)	771	219
uhepah_expert1	*Non-Artifact (0)*	11062	46
	Artifact (1)	709	227
ukudab_expert2	*Non-Artifact (0)*	12390	323
	Artifact (1)	1027	557

Table 86

Performance of C = 100 and Gamma = 0.001

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7835	0.7828	0.8531	0.7827	0.7835
ajeric_expert2	0.9191	0.9038	0.8456	0.9122	0.9191
ekamis_expert2	0.9186	0.8894	0.7241	0.9061	0.9186
iguted_expert1	0.9091	0.8851	0.7922	0.9046	0.9091
inefoh_expert1	0.9437	0.9265	0.7865	0.9354	0.9437
otafeh_expert1	0.9548	0.9391	0.7688	0.9487	0.9548
oxused_expert2	0.8873	0.8690	0.8605	0.8865	0.8873
pqbqpr_expert2	0.9431	0.9292	0.7838	0.9315	0.9431
uhepah_expert1	0.9373	0.9210	0.8398	0.9314	0.9373
ukudab_expert2	0.9056	0.8934	0.8418	0.8913	0.9056
	0.9102	**0.8939**	**0.8096**	**0.9030**	**0.9102**

Another hyperparameter tuning using the combination of parameters C with the value of 100, and Gamma with value of 0.001 was applied to find the one configuration that produces the best results. This tuning resulted in accuracy, F1-score, ROC-AUC, precision, and recall with values of 0.9102, 0.8939, 0.8096, 0.9030, and 0.9102, as shown in Table 86.

Table 87

Confusion Matrix of C = 100 and Gamma = 0.01

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5645	1510
	Artifact (1)	1244	4129
ajeric_expert2	*Non-Artifact (0)*	11215	155
	Artifact (1)	834	567
ekamis_expert2	*Non-Artifact (0)*	12371	57
	Artifact (1)	1023	168
iguted_expert1	*Non-Artifact (0)*	11571	84
	Artifact (1)	1067	413
inefoh_expert1	*Non-Artifact (0)*	12432	85
	Artifact (1)	651	209
otafeh_expert1	*Non-Artifact (0)*	11295	26
	Artifact (1)	491	110
oxused_expert2	*Non-Artifact (0)*	9775	135
	Artifact (1)	1179	865
pqbqpr_expert2	*Non-Artifact (0)*	14122	203
	Artifact (1)	749	241
uhepah_expert1	*Non-Artifact (0)*	11026	82
	Artifact (1)	622	314
ukudab_expert2	*Non-Artifact (0)*	12307	406
	Artifact (1)	962	622

Table 88

Performance of C = 100 and Gamma = 0.01

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7802	0.7807	0.8503	0.7820	0.7802
ajeric_expert2	0.9226	0.9113	0.8430	0.9148	0.9226
ekamis_expert2	0.9207	0.8951	0.6970	0.9081	0.9207
iguted_expert1	0.9124	0.8924	0.7704	0.9060	0.9124
inefoh_expert1	0.9450	0.9321	0.7976	0.9349	0.9450
otafeh_expert1	0.9566	0.9434	0.7630	0.9508	0.9566
oxused_expert2	0.8901	0.8740	0.8244	0.8877	0.8901
pqbqpr_expert2	0.9378	0.9266	0.7410	0.9233	0.9378
uhepah_expert1	0.9415	0.9304	0.8102	0.9347	0.9415
ukudab_expert2	0.9043	0.8952	0.8287	0.8918	0.9043
	0.9111	**0.8981**	**0.7926**	**0.9034**	**0.9111**

Table 87 represents the total number of actual and predicted instances in the SVM model that was optimized using C of 100 and Gamma of 0.01. These instances were calculated using the formula of accuracy, F1-score, ROC-AUC, precision, and recall to quantify its performance in detecting artifacts in an EDA signal. The configuration resulted in 0.9111, 0.8981, 0.7926, 0.9034, and 0.9111, in the same order with its respective metric.

Table 89

Confusion Matrix of C = 1000 and Gamma = 0.001

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5867	1288
	Artifact (1)	1400	3973
ajeric_expert2	*Non-Artifact (0)*	11257	113
	Artifact (1)	901	500
ekamis_expert2	*Non-Artifact (0)*	12377	51
	Artifact (1)	1050	141
iguted_expert1	*Non-Artifact (0)*	11588	67
	Artifact (1)	1091	389
inefoh_expert1	*Non-Artifact (0)*	12476	41
	Artifact (1)	678	182
otafeh_expert1	*Non-Artifact (0)*	11302	19
	Artifact (1)	514	87
oxused_expert2	*Non-Artifact (0)*	9799	111
	Artifact (1)	1209	835
pqbqpr_expert2	*Non-Artifact (0)*	14219	106
	Artifact (1)	780	210
uhepah_expert1	*Non-Artifact (0)*	11050	58
	Artifact (1)	676	260
ukudab_expert2	*Non-Artifact (0)*	12384	329
	Artifact (1)	1015	569

Table 89 represents the total number of actual and predicted instances in the SVM model that was optimized using C of 1000 and Gamma of 0.001. These instances were calculated using the formula of accuracy, F1-score, ROC-AUC, precision, and recall to quantify its performance in detecting artifacts in an EDA signal. The configuration resulted in 0.9115, 0.8964, 0.8050, 0.9044, and 0.9115, in the same order with its respective metric.

Table 90

Performance of C = 1000 and Gamma = 0.001

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7854	0.7851	0.8551	0.7850	0.7854
ajeric_expert2	0.9206	0.9064	0.8450	0.9138	0.9206
ekamis_expert2	0.9192	0.8915	0.7110	0.9054	0.9192
iguted_expert1	0.9118	0.8904	0.7843	0.9071	0.9118
inefoh_expert1	0.9463	0.9311	0.8074	0.9400	0.9463
otafeh_expert1	0.9553	0.9401	0.7588	0.9497	0.9553
oxused_expert2	0.8896	0.8722	0.8502	0.8889	0.8896
pqbqpr_expert2	0.9421	0.9279	0.7651	0.9297	0.9421
uhepah_expert1	0.9391	0.9249	0.8316	0.9327	0.9391
ukudab_expert2	0.9060	0.8942	0.8411	0.8920	0.9060
	0.9115	**0.8964**	**0.8050**	**0.9044**	**0.9115**

After the hyperparameter tuning and experimentation done to the SVM model of Hossain et. al. (2022) for automatic detection of artifacts in EDA signals, the results of each configuration were then compared to each other. The parameter combination that yielded the most promising result is the C = 100 and Gamma = 0.01. This combination produced 0.9111, 0.8981, 0.7926, 0.9034, and 0.9111 in terms of accuracy, F1-score, ROC-AUC, precision, and recall. This is the tuned model that will be analyzed alongside the other replicated models and the LSTM-SVM itself.

Linear Regression (Taylor et. al., 2015)

Taylor et. al. (2015) is one of the study's baseline models. To compare LSTM-SVM performance, the researchers reproduced Taylor et al.'s models. In their investigation, et al. (2015) used one of these methods: Linear Regression. The LR model's confusion matrix is shown in Table 91, and these findings were used to calculate the model's performance across measures.

Table 91

Confusion Matrix of C = 0.01

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	7148	7
	Artifact (1)	4366	1007
ajeric_expert2	*Non-Artifact (0)*	11360	10
	Artifact (1)	1330	71
ekamis_expert2	*Non-Artifact (0)*	12389	39
	Artifact (1)	1150	41
iguted_expert1	*Non-Artifact (0)*	11610	45
	Artifact (1)	1251	229
inefoh_expert1	*Non-Artifact (0)*	12360	157
	Artifact (1)	729	131
otafeh_expert1	*Non-Artifact (0)*	11319	2
	Artifact (1)	543	58
oxused_expert2	*Non-Artifact (0)*	9842	68
	Artifact (1)	1689	355
pqbqpr_expert2	*Non-Artifact (0)*	13818	507
	Artifact (1)	793	197
uhepah_expert1	*Non-Artifact (0)*	11108	0
	Artifact (1)	916	20
ukudab_expert2	*Non-Artifact (0)*	12679	34
	Artifact (1)	1561	23

Table 92

Performance of C = 0.01

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.6509	0.5726	0.8626	0.7805	0.6509
ajeric_expert2	0.8951	0.8512	0.7608	0.8931	0.8951
ekamis_expert2	0.9127	0.8764	0.6713	0.8799	0.9127
iguted_expert1	0.9013	0.8698	0.8623	0.8952	0.9013
inefoh_expert1	0.9338	0.9180	0.7984	0.9128	0.9338
otafeh_expert1	0.9543	0.9361	0.7611	0.9549	0.9543
oxused_expert2	0.8530	0.8103	0.8083	0.8511	0.8530
pqbqpr_expert2	0.9151	0.9084	0.7336	0.9027	0.9151
uhepah_expert1	0.9239	0.8890	0.8734	0.9297	0.9239
ukudab_expert2	0.8884	0.8397	0.7403	0.8364	0.8884
	0.8829	**0.8472**	**0.7872**	**0.8836**	**0.8829**

The detected true negative, true positive, false negative, and false positive examples yielded accuracy, F1-score, ROC-AUC, precision, and recall values of 0.8829, 0.8472, 0.7872, 0.8836, and 0.8829, respectively, as displayed in Table 92.

Table 93

Confusion Matrix of C = 0.1

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	7108	47
	Artifact (1)	3883	1490
ajeric_expert2	*Non-Artifact (0)*	11352	18
	Artifact (1)	1301	100
ekamis_expert2	*Non-Artifact (0)*	12382	46
	Artifact (1)	1124	67
iguted_expert1	*Non-Artifact (0)*	11578	77
	Artifact (1)	1149	331
inefoh_expert1	*Non-Artifact (0)*	12362	155
	Artifact (1)	675	185
otafeh_expert1	*Non-Artifact (0)*	11309	12
	Artifact (1)	521	80
oxused_expert2	*Non-Artifact (0)*	9822	88
	Artifact (1)	1540	504
pqbqpr_expert2	*Non-Artifact (0)*	13781	544
	Artifact (1)	729	261
uhepah_expert1	*Non-Artifact (0)*	11107	1
	Artifact (1)	880	56
ukudab_expert2	*Non-Artifact (0)*	12652	61
	Artifact (1)	1524	60

Table 94

Performance of C = 0.1

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.6863	0.6324	0.8609	0.7851	0.6863
ajeric_expert2	0.8967	0.8559	0.7853	0.8917	0.8967
ekamis_expert2	0.9141	0.8804	0.6862	0.8885	0.9141
iguted_expert1	0.9067	0.8822	0.8455	0.8986	0.9067
inefoh_expert1	0.9380	0.9251	0.7925	0.9222	0.9380
otafeh_expert1	0.9553	0.9394	0.7477	0.9516	0.9553
oxused_expert2	0.8638	0.8310	0.8263	0.8622	0.8638
pqbqpr_expert2	0.9169	0.9129	0.7543	0.9093	0.9169
uhepah_expert1	0.9269	0.8959	0.8617	0.9309	0.9269
ukudab_expert2	0.8891	0.8446	0.7639	0.8486	0.8891
	0.8894	**0.8600**	**0.7924**	**0.8889**	**0.8894**

The LR model's confusion matrix is shown in Table 93, and these findings were used to calculate the model's performance across measures. The detected true negative, true positive, false negative, and false positive examples yielded accuracy, F1-score, ROC-AUC, precision, and recall values of 0.8894, 0.8600, 0.7924, 0.8889, and 0.8894, respectively, as displayed in Table 94.

Table 95

Confusion Matrix of C = 1

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	7071	84
	Artifact (1)	3523	1850
ajeric_expert2	*Non-Artifact (0)*	11343	27
	Artifact (1)	1275	126
ekamis_expert2	*Non-Artifact (0)*	12383	45
	Artifact (1)	1099	92
iguted_expert1	*Non-Artifact (0)*	11572	83
	Artifact (1)	1117	363
inefoh_expert1	*Non-Artifact (0)*	12412	105
	Artifact (1)	670	190
otafeh_expert1	*Non-Artifact (0)*	11304	17
	Artifact (1)	514	87
oxused_expert2	*Non-Artifact (0)*	9790	120
	Artifact (1)	1380	664
pqbqpr_expert2	*Non-Artifact (0)*	13941	384
	Artifact (1)	722	268
uhepah_expert1	*Non-Artifact (0)*	11106	2
	Artifact (1)	847	89
ukudab_expert2	*Non-Artifact (0)*	12620	93
	Artifact (1)	1480	104

Table 96

Performance of C = 1

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7121	0.6722	0.8610	0.7914	0.7121
ajeric_expert2	0.8981	0.8598	0.8062	0.8907	0.8981
ekamis_expert2	0.9160	0.8844	0.6748	0.8969	0.9160
iguted_expert1	0.9086	0.8861	0.8102	0.9009	0.9086
inefoh_expert1	0.9421	0.9285	0.7628	0.9292	0.9421
otafeh_expert1	0.9555	0.9402	0.7109	0.9505	0.9555
oxused_expert2	0.8745	0.8503	0.8473	0.8714	0.8745
pqbqpr_expert2	0.9278	0.9208	0.7530	0.9159	0.9278
uhepah_expert1	0.9295	0.9018	0.8339	0.9329	0.9295
ukudab_expert2	0.8900	0.8500	0.7873	0.8544	0.8900
	0.8954	**0.8694**	**0.7847**	**0.8934**	**0.8954**

The LR model's confusion matrix is shown in Table 95, and these findings were used to calculate the model's performance across measures. The detected true negative, true positive, false negative, and false positive examples yielded accuracy, F1-score, ROC-AUC, precision, and recall values of 0.8954, 0.8694, 0.7847, 0.8934, and 0.8954, respectively, as displayed in Table 96.

Table 97

Confusion Matrix of C = 10

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	7044	111
	Artifact (1)	3430	1943
ajeric_expert2	*Non-Artifact (0)*	11340	30
	Artifact (1)	1262	139
ekamis_expert2	*Non-Artifact (0)*	12389	39
	Artifact (1)	1097	94
iguted_expert1	*Non-Artifact (0)*	11585	70
	Artifact (1)	1150	330
inefoh_expert1	*Non-Artifact (0)*	12455	62
	Artifact (1)	703	157
otafeh_expert1	*Non-Artifact (0)*	11303	18
	Artifact (1)	519	82
oxused_expert2	*Non-Artifact (0)*	9797	113
	Artifact (1)	1356	688
pqbqpr_expert2	*Non-Artifact (0)*	14121	204
	Artifact (1)	753	237
uhepah_expert1	*Non-Artifact (0)*	11102	6
	Artifact (1)	839	97
ukudab_expert2	*Non-Artifact (0)*	12620	93
	Artifact (1)	1460	124

The LR model's confusion matrix is shown in Table 97, and these findings were used to calculate the model's performance across measures. The detected true negative, true positive, false negative, and false positive examples yielded accuracy, F1-score, ROC-AUC, precision, and recall values of 0.8974, 0.8712, 0.7735, 0.8951, and 0.8974, respectively, as displayed in Table 98.

Table 98

Performance of C = 10

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7174	0.6808	0.8587	0.7898	0.7174
ajeric_expert2	0.8988	0.8617	0.8137	0.8914	0.8988
ekamis_expert2	0.9166	0.8850	0.6584	0.9001	0.9166
iguted_expert1	0.9071	0.8825	0.7791	0.9002	0.9071
inefoh_expert1	0.9428	0.9265	0.7357	0.9318	0.9428
otafeh_expert1	0.9550	0.9393	0.6777	0.9492	0.9550
oxused_expert2	0.8771	0.8539	0.8583	0.8751	0.8771
pqbqpr_expert2	0.9375	0.9261	0.7421	0.9227	0.9375
uhepah_expert1	0.9298	0.9030	0.8084	0.9307	0.9298
ukudab_expert2	0.8914	0.8529	0.8027	0.8603	0.8914
	0.8974	0.8712	0.7735	0.8951	0.8974

Table 99

Confusion Matrix of C = 100 Configuration

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	7039	116
	Artifact (1)	3409	1964
ajeric_expert2	*Non-Artifact (0)*	11339	31
	Artifact (1)	1254	147
ekamis_expert2	*Non-Artifact (0)*	12389	39
	Artifact (1)	1096	95
iguted_expert1	*Non-Artifact (0)*	11585	70
	Artifact (1)	1155	325
inefoh_expert1	*Non-Artifact (0)*	12456	61
	Artifact (1)	704	156
otafeh_expert1	*Non-Artifact (0)*	11301	20
	Artifact (1)	520	81
oxused_expert2	*Non-Artifact (0)*	9797	113
	Artifact (1)	1347	697
pqbqpr_expert2	*Non-Artifact (0)*	14140	185
	Artifact (1)	753	237
uhepah_expert1	*Non-Artifact (0)*	11102	6
	Artifact (1)	838	98
ukudab_expert2	*Non-Artifact (0)*	12619	94
	Artifact (1)	1454	130

The LR model's confusion matrix is shown in Table 99, and these findings were used to calculate the model's performance across measures. The detected true negative, true positive, false negative, and false positive examples yielded accuracy, F1-score, ROC-AUC, precision, and recall values of 0.8977, 0.8717, 0.7719, 0.8955, and 0.8977, respectively, as displayed in Table 100.

Table 100

Performance of C = 100

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7186	0.6828	0.8582	0.7897	0.7186
ajeric_expert2	0.8994	0.8630	0.8142	0.8922	0.8994
ekamis_expert2	0.9167	0.8851	0.6558	0.9004	0.9167
iguted_expert1	0.9067	0.8818	0.7755	0.8996	0.9067
inefoh_expert1	0.9428	0.9265	0.7325	0.9319	0.9428
otafeh_expert1	0.9547	0.9391	0.6741	0.9482	0.9547
oxused_expert2	0.8779	0.8550	0.8593	0.8759	0.8779
pqbqpr_expert2	0.9388	0.9270	0.7404	0.9244	0.9388
uhepah_expert1	0.9299	0.9032	0.8050	0.9308	0.9299
ukudab_expert2	0.8917	0.8538	0.8043	0.8616	0.8917
	0.8977	**0.8717**	**0.7719**	**0.8955**	**0.8977**

Random Forest (Taylor et. al., 2015)

Following the Linear Regression model, the Random Forest model by Taylor et al. (2015) was used to compare its performance with the proposed model of the current study's researchers.

Table 101

Confusion Matrix of N Estimators = 200, Max Depth = 10

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	6230	925
	Artifact (1)	1660	3713
ajeric_expert2	*Non-Artifact (0)*	11158	212
	Artifact (1)	932	469
ekamis_expert2	*Non-Artifact (0)*	12353	75
	Artifact (1)	1014	177
iguted_expert1	*Non-Artifact (0)*	11540	115
	Artifact (1)	1026	454
inefoh_expert1	*Non-Artifact (0)*	12409	108
	Artifact (1)	613	247
otafeh_expert1	*Non-Artifact (0)*	11287	34
	Artifact (1)	461	140
oxused_expert2	*Non-Artifact (0)*	9755	155
	Artifact (1)	1230	814
pqbqpr_expert2	*Non-Artifact (0)*	14086	239
	Artifact (1)	679	311
uhepah_expert1	*Non-Artifact (0)*	10983	125
	Artifact (1)	580	356
ukudab_expert2	*Non-Artifact (0)*	12280	433
	Artifact (1)	1105	479

Table 102

Performance of N Estimators = 200, Max Depth = 10

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7937	0.7911	0.8609	0.7943	0.7937
ajeric_expert2	0.9104	0.8963	0.8500	0.8972	0.9104
ekamis_expert2	0.9200	0.8955	0.6861	0.9047	0.9200
iguted_expert1	0.9131	0.8955	0.8636	0.9048	0.9131
inefoh_expert1	0.9461	0.9354	0.8206	0.9364	0.9461
otafeh_expert1	0.9585	0.9474	0.8567	0.9529	0.9585
oxused_expert2	0.8841	0.8665	0.8951	0.8798	0.8841
pqbqpr_expert2	0.9401	0.9319	0.7991	0.9289	0.9401
uhepah_expert1	0.9415	0.9327	0.8900	0.9335	0.9415
ukudab_expert2	0.8924	0.8793	0.7572	0.8740	0.8924
	0.9100	**0.8972**	**0.8279**	**0.9007**	**0.9100**

Table 101 displays the confusion matrixes for each test subject, with a configuration of N Estimators = 200, and Max Depth = 10, highlighting the occurrences

of true positives, true negatives, false positives, and false negatives. The results yielded an accuracy of 0.9100, an F1-score of 0.8972, a ROC-AUC of 0.8279, a precision of 0.9007, and a recall of 0.9100.

Table 103

Confusion Matrix of N Estimators = 200, Max Depth = 30

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5621	1534
	Artifact (1)	1265	4108
ajeric_expert2	*Non-Artifact (0)*	11049	321
	Artifact (1)	724	677
ekamis_expert2	*Non-Artifact (0)*	12319	109
	Artifact (1)	979	212
iguted_expert1	*Non-Artifact (0)*	11521	134
	Artifact (1)	1001	479
inefoh_expert1	*Non-Artifact (0)*	12384	133
	Artifact (1)	592	268
otafeh_expert1	*Non-Artifact (0)*	11281	40
	Artifact (1)	460	141
oxused_expert2	*Non-Artifact (0)*	9715	195
	Artifact (1)	1102	942
pqbqpr_expert2	*Non-Artifact (0)*	14027	298
	Artifact (1)	654	336
uhepah_expert1	*Non-Artifact (0)*	10922	186
	Artifact (1)	540	396
ukudab_expert2	*Non-Artifact (0)*	12017	696
	Artifact (1)	860	724

Table 104

Performance of N Estimators = 200, Max Depth = 30

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7766	0.7772	0.8514	0.7785	0.7766
ajeric_expert2	0.9182	0.9120	0.8806	0.9100	0.9182
ekamis_expert2	0.9201	0.8985	0.7209	0.9031	0.9201
iguted_expert1	0.9136	0.8972	0.8615	0.9044	0.9136
inefoh_expert1	0.9458	0.9364	0.8451	0.9360	0.9458
otafeh_expert1	0.9581	0.9472	0.8550	0.9517	0.9581
oxused_expert2	0.8915	0.8784	0.9013	0.8862	0.8915
pqbqpr_expert2	0.9378	0.9314	0.8019	0.9279	0.9378
uhepah_expert1	0.9397	0.9332	0.8797	0.9317	0.9397
ukudab_expert2	0.8912	0.8885	0.8109	0.8863	0.8912
	0.9093	**0.9000**	**0.8408**	**0.9016**	**0.9093**

Table 103 displays the confusion matrixes for each test subject, using the Random Forest model by Taylor et al. (2015) with a configuration of N Estimators = 200, and Max Depth = 30, highlighting the occurrences of true positives, true negatives, false

positives, and false negatives. The results yielded an accuracy of 0.9093, an F1-score of 0.9000, a ROC-AUC of 0.8408, a precision of 0.9016, and a recall of 0.9093.

Table 105

Confusion Matrix of N Estimators = 200, Max Depth = 50

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5625	1530
	Artifact (1)	1261	4112
ajeric_expert2	*Non-Artifact (0)*	11055	315
	Artifact (1)	736	665
ekamis_expert2	*Non-Artifact (0)*	12315	113
	Artifact (1)	973	218
iguted_expert1	*Non-Artifact (0)*	11526	129
	Artifact (1)	993	487
inefoh_expert1	*Non-Artifact (0)*	12393	124
	Artifact (1)	591	269
otafeh_expert1	*Non-Artifact (0)*	11283	38
	Artifact (1)	456	145
oxused_expert2	*Non-Artifact (0)*	9709	201
	Artifact (1)	1113	931
pqbqpr_expert2	*Non-Artifact (0)*	14017	308
	Artifact (1)	660	330
uhepah_expert1	*Non-Artifact (0)*	10919	189
	Artifact (1)	543	393
ukudab_expert2	*Non-Artifact (0)*	12013	700
	Artifact (1)	870	714

Table 106

Performance of N Estimators = 200, Max Depth = 50

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7772	0.7778	0.8517	0.7791	0.7772
ajeric_expert2	0.9177	0.9112	0.8812	0.9092	0.9177
ekamis_expert2	0.9203	0.8991	0.7239	0.9033	0.9203
iguted_expert1	0.9146	0.8985	0.8612	0.9060	0.9146
inefoh_expert1	0.9466	0.9371	0.8438	0.9371	0.9466
otafeh_expert1	0.9586	0.9479	0.8575	0.9526	0.9586
oxused_expert2	0.8901	0.8767	0.9017	0.8844	0.8901
pqbqpr_expert2	0.9368	0.9303	0.8012	0.9267	0.9368
uhepah_expert1	0.9392	0.9326	0.8804	0.9311	0.9392
ukudab_expert2	0.8902	0.8874	0.8085	0.8851	0.8902
	0.9091	**0.8999**	**0.8411**	**0.9015**	**0.9091**

Table 105 displays the confusion matrixes for each test subject, using the Random Forest model by Taylor et al. (2015) with a configuration of N Estimators = 200, and Max Depth = 50, highlighting the occurrences of true positives, true negatives, false positives, and false negatives. The results yielded an accuracy of 0.9091, an F1-score of 0.8999, a ROC-AUC of 0.8411, a precision of 0.9015, and a recall of 0.9091.

Table 107

Confusion Matrix of N Estimators = 400, Max Depth = 10

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	6234	921
	Artifact (1)	1661	3712
ajeric_expert2	*Non-Artifact (0)*	11152	218
	Artifact (1)	933	468
ekamis_expert2	*Non-Artifact (0)*	12354	74
	Artifact (1)	1016	175
iguted_expert1	*Non-Artifact (0)*	11538	117
	Artifact (1)	1019	461
inefoh_expert1	*Non-Artifact (0)*	12400	117
	Artifact (1)	609	251
otafeh_expert1	*Non-Artifact (0)*	11287	34
	Artifact (1)	460	141
oxused_expert2	*Non-Artifact (0)*	9754	156
	Artifact (1)	1227	817
pqbqpr_expert2	*Non-Artifact (0)*	14077	248
	Artifact (1)	679	311
uhepah_expert1	*Non-Artifact (0)*	10977	131
	Artifact (1)	575	361
ukudab_expert2	*Non-Artifact (0)*	12278	435
	Artifact (1)	1110	474

Table 108

Performance of N Estimators = 400, Max Depth = 10

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7939	0.7913	0.8609	0.7946	0.7939
ajeric_expert2	0.9099	0.8958	0.8489	0.8964	0.9099
ekamis_expert2	0.9200	0.8952	0.6952	0.9047	0.9200
iguted_expert1	0.9135	0.8962	0.8648	0.9052	0.9135
inefoh_expert1	0.9457	0.9354	0.8198	0.9358	0.9457
otafeh_expert1	0.9586	0.9476	0.8558	0.9530	0.9586
oxused_expert2	0.8843	0.8667	0.8951	0.8800	0.8843
pqbqpr_expert2	0.9395	0.9315	0.8029	0.9283	0.9395
uhepah_expert1	0.9414	0.9328	0.8915	0.9334	0.9414
ukudab_expert2	0.8919	0.8787	0.7563	0.8733	0.8919
	0.9099	**0.8971**	**0.8291**	**0.9005**	**0.9099**

Table 107 displays the confusion matrixes for each test subject, using the Random Forest model by Taylor et al. (2015) with a configuration of N Estimators = 400, and Max Depth = 10, highlighting the occurrences of true positives, true negatives, false positives, and false negatives. The results yielded an accuracy of 0.9099, an F1-score of 0.8971, a ROC-AUC of 0.8291, a precision of 0.9005, and a recall of 0.9099.

Table 109

Confusion Matrix of N Estimators = 400, Max Depth = 30

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5618	1537
	Artifact (1)	1258	4115
ajeric_expert2	*Non-Artifact (0)*	11050	320
	Artifact (1)	724	677
ekamis_expert2	*Non-Artifact (0)*	12316	112
	Artifact (1)	981	210
iguted_expert1	*Non-Artifact (0)*	11518	137
	Artifact (1)	993	487
inefoh_expert1	*Non-Artifact (0)*	12390	127
	Artifact (1)	595	265
otafeh_expert1	*Non-Artifact (0)*	11281	40
	Artifact (1)	458	143
oxused_expert2	*Non-Artifact (0)*	9714	196
	Artifact (1)	1113	931
pqbqpr_expert2	*Non-Artifact (0)*	14023	302
	Artifact (1)	660	330
uhepah_expert1	*Non-Artifact (0)*	10916	192
	Artifact (1)	540	396
ukudab_expert2	*Non-Artifact (0)*	12001	712
	Artifact (1)	865	719

Table 110

Performance of N Estimators = 400, Max Depth = 30

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7769	0.7775	0.8515	0.7789	0.7769
ajeric_expert2	0.9183	0.9121	0.8819	0.9100	0.9183
ekamis_expert2	0.9197	0.8981	0.7207	0.9023	0.9197
iguted_expert1	0.9140	0.8980	0.8641	0.9048	0.9140
inefoh_expert1	0.9460	0.9364	0.8455	0.9363	0.9460
otafeh_expert1	0.9582	0.9475	0.8635	0.9519	0.9582
oxused_expert2	0.8905	0.8771	0.9052	0.8850	0.8905
pqbqpr_expert2	0.9372	0.9306	0.8013	0.9271	0.9372
uhepah_expert1	0.9392	0.9328	0.8816	0.9312	0.9392
ukudab_expert2	0.8897	0.8872	0.8116	0.8851	0.8897
	0.9090	**0.8997**	**0.8427**	**0.9013**	**0.9090**

Table 109 displays the confusion matrixes for each test subject, using the Random Forest model by Taylor et al. (2015) with a configuration of N Estimators = 400, and Max Depth = 30, highlighting the occurrences of true positives, true negatives, false positives, and false negatives. The results yielded an accuracy of 0.9090, an F1-score of 0.8997, a ROC-AUC of 0.8427, a precision of 0.9013, and a recall of 0.9090.

Table 111

Confusion Matrix of N Estimators = 400, Max Depth = 50

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5587	1568
	Artifact (1)	1246	4127
ajeric_expert2	*Non-Artifact (0)*	11056	314
	Artifact (1)	721	680
ekamis_expert2	*Non-Artifact (0)*	12316	112
	Artifact (1)	975	216
iguted_expert1	*Non-Artifact (0)*	11522	133
	Artifact (1)	994	486
inefoh_expert1	*Non-Artifact (0)*	12395	122
	Artifact (1)	595	265
otafeh_expert1	*Non-Artifact (0)*	11283	38
	Artifact (1)	462	139
oxused_expert2	*Non-Artifact (0)*	9715	195
	Artifact (1)	1110	934
pqbqpr_expert2	*Non-Artifact (0)*	14030	295
	Artifact (1)	659	331
uhepah_expert1	*Non-Artifact (0)*	10928	180
	Artifact (1)	546	390
ukudab_expert2	*Non-Artifact (0)*	12003	710
	Artifact (1)	863	721

Table 112

Performance of N Estimators = 400, Max Depth = 50

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7754	0.7761	0.8508	0.7778	0.7754
ajeric_expert2	0.9190	0.9128	0.8817	0.9108	0.9190
ekamis_expert2	0.9202	0.8989	0.7191	0.9032	0.9202
iguted_expert1	0.9142	0.8981	0.8638	0.9053	0.9142
inefoh_expert1	0.9464	0.9367	0.8459	0.9369	0.9464
otafeh_expert1	0.9581	0.9470	0.8637	0.9518	0.9581
oxused_expert2	0.8908	0.8775	0.9047	0.8855	0.8908
pqbqpr_expert2	0.9377	0.9311	0.8022	0.9276	0.9377
uhepah_expert1	0.9397	0.9329	0.8807	0.9316	0.9397
ukudab_expert2	0.8900	0.8875	0.8119	0.8854	0.8900
	0.9091	**0.8999**	**0.8425**	**0.9016**	**0.9091**

Table 111 displays the confusion matrixes for each test subject, using the Random Forest model by Taylor et al. (2015) with a configuration of N Estimators = 400, and Max Depth = 50, highlighting the occurrences of true positives, true negatives, false positives, and false negatives. The results yielded an accuracy of 0.9091, an F1-score of 0.8999, a ROC-AUC of 0.8425, a precision of 0.9016, and a recall of 0.9091.

Table 113

Confusion Matrix of N Estimators = 600, Max Depth = 10

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	6228	927
	Artifact (1)	1669	3704
ajeric_expert2	*Non-Artifact (0)*	11160	210
	Artifact (1)	936	465
ekamis_expert2	*Non-Artifact (0)*	12354	74
	Artifact (1)	1015	176
iguted_expert1	*Non-Artifact (0)*	11540	115
	Artifact (1)	1018	462
inefoh_expert1	*Non-Artifact (0)*	12399	118
	Artifact (1)	612	248
otafeh_expert1	*Non-Artifact (0)*	11287	34
	Artifact (1)	461	140
oxused_expert2	*Non-Artifact (0)*	9757	153
	Artifact (1)	1229	815
pqbqpr_expert2	*Non-Artifact (0)*	14080	245
	Artifact (1)	678	312
uhepah_expert1	*Non-Artifact (0)*	10982	126
	Artifact (1)	580	356
ukudab_expert2	*Non-Artifact (0)*	12284	429
	Artifact (1)	1102	482

Table 114

Performance N Estimators = 600, Max Depth = 10

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7928	0.7902	0.8611	0.7934	0.7928
ajeric_expert2	0.9103	0.8960	0.8476	0.8970	0.9103
ekamis_expert2	0.9200	0.8954	0.6937	0.9048	0.9200
iguted_expert1	0.9137	0.8964	0.8643	0.9056	0.9137
inefoh_expert1	0.9454	0.9350	0.8193	0.9353	0.9454
otafeh_expert1	0.9585	0.9474	0.8567	0.9529	0.9585
oxused_expert2	0.8844	0.8667	0.8952	0.8802	0.8844
pqbqpr_expert2	0.9397	0.9317	0.8019	0.9286	0.9397
uhepah_expert1	0.9414	0.9326	0.8907	0.9334	0.9414
ukudab_expert2	0.8929	0.8799	0.7544	0.8746	0.8929
	0.9099	**0.8971**	**0.8285**	**0.9006**	**0.9099**

Table 113 displays the confusion matrixes for each test subject, using the Random Forest model by Taylor et al. (2015) with a configuration of N Estimators = 600, and Max Depth = 10, highlighting the occurrences of true positives, true negatives, false positives, and false negatives. The results yielded an accuracy of 0.9099, an F1-score of 0.8971, a ROC-AUC of 0.8285, a precision of 0.9006, and a recall of 0.9099

Table 115

Confusion Matrix of N Estimators = 600, Max Depth = 30

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5605	1550
	Artifact (1)	1246	4127
ajeric_expert2	*Non-Artifact (0)*	11059	311
	Artifact (1)	717	684
ekamis_expert2	*Non-Artifact (0)*	12325	103
	Artifact (1)	974	217
iguted_expert1	*Non-Artifact (0)*	11517	138
	Artifact (1)	995	485
inefoh_expert1	*Non-Artifact (0)*	12391	126
	Artifact (1)	596	264
otafeh_expert1	*Non-Artifact (0)*	11280	41
	Artifact (1)	457	144
oxused_expert2	*Non-Artifact (0)*	9708	202
	Artifact (1)	1111	933
pqbqpr_expert2	*Non-Artifact (0)*	14025	300
	Artifact (1)	657	333
uhepah_expert1	*Non-Artifact (0)*	10925	183
	Artifact (1)	542	394
ukudab_expert2	*Non-Artifact (0)*	11999	714
	Artifact (1)	866	718

Table 116

Performance of N Estimators = 600, Max Depth = 30

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7768	0.7775	0.8518	0.7790	0.7768
ajeric_expert2	0.9195	0.9134	0.8832	0.9115	0.9195
ekamis_expert2	0.9209	0.8995	0.7195	0.9050	0.9209
iguted_expert1	0.9137	0.8977	0.8659	0.9045	0.9137
inefoh_expert1	0.9460	0.9364	0.8481	0.9363	0.9460
otafeh_expert1	0.9582	0.9476	0.8656	0.9519	0.9582
oxused_expert2	0.8902	0.8769	0.9042	0.8844	0.8902
pqbqpr_expert2	0.9375	0.9310	0.7999	0.9275	0.9375
uhepah_expert1	0.9398	0.9331	0.8825	0.9318	0.9398
ukudab_expert2	0.8895	0.8870	0.8113	0.8849	0.8895
	0.9092	**0.9000**	**0.8432**	**0.9017**	**0.9092**

Table 115 displays the confusion matrixes for each test subject, using the Random Forest model by Taylor et al. (2015) with a configuration of N Estimators = 600, and Max Depth = 30, highlighting the occurrences of true positives, true negatives, false positives, and false negatives. The results yielded an accuracy of 0.9092, an F1-score of 0.9000, a ROC-AUC of 0.8432, a precision of 0.9017, and a recall of 0.9092.

Table 117

Confusion Matrix of N Estimators = 600, Max Depth = 50

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	5614	1541
	Artifact (1)	1229	4144
ajeric_expert2	*Non-Artifact (0)*	11061	309
	Artifact (1)	722	679
ekamis_expert2	*Non-Artifact (0)*	12322	106
	Artifact (1)	977	214
iguted_expert1	*Non-Artifact (0)*	11526	129
	Artifact (1)	994	486
inefoh_expert1	*Non-Artifact (0)*	12383	134
	Artifact (1)	588	272
otafeh_expert1	*Non-Artifact (0)*	11284	37
	Artifact (1)	458	143
oxused_expert2	*Non-Artifact (0)*	9713	197
	Artifact (1)	1116	928
pqbqpr_expert2	*Non-Artifact (0)*	14021	304
	Artifact (1)	661	329
uhepah_expert1	*Non-Artifact (0)*	10939	169
	Artifact (1)	545	391
ukudab_expert2	*Non-Artifact (0)*	12004	709
	Artifact (1)	872	712

Table 118

Performance of N Estimators = 600, Max Depth = 50

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7789	0.7796	0.8520	0.7812	0.7789
ajeric_expert2	0.9193	0.9130	0.8821	0.9111	0.9193
ekamis_expert2	0.9205	0.8989	0.7200	0.9040	0.9205
iguted_expert1	0.9145	0.8984	0.8638	0.9059	0.9145
inefoh_expert1	0.9460	0.9368	0.8465	0.9364	0.9460
otafeh_expert1	0.9585	0.9477	0.8613	0.9526	0.9585
oxused_expert2	0.8902	0.8767	0.9038	0.8846	0.8902
pqbqpr_expert2	0.9370	0.9304	0.8036	0.9268	0.9370
uhepah_expert1	0.9407	0.9338	0.8805	0.9328	0.9407
ukudab_expert2	0.8894	0.8868	0.8116	0.8845	0.8894
	0.9095	**0.9002**	**0.8425**	**0.9020**	**0.9095**

Table 117 displays the confusion matrixes for each test subject, using the Random Forest model by Taylor et al. (2015) with a configuration of N Estimators = 600, and Max Depth = 50, highlighting the occurrences of true positives, true negatives, false positives, and false negatives. The results yielded an accuracy of 0.9095, an F1-score of 0.9002, a ROC-AUC of 0.8425, a precision of 0.9020, and a recall of 0.9095.

Similar to the aforementioned Linear Regression model, hyperparameter optimization process was also done for the Random Forest classifier from the study of

Taylor et al. (2015). The researchers experimented also with various combinations of N Estimators and Max Depth values for the model. Based on the evaluation metrics of accuracy, precision, F1-score, ROC-AUC, and recall, the optimal configuration was handpicked by the researchers and identified as N Estimators = 600 and Max Depth = 30. This specific combination delivered the most favorable outcomes, outperforming other tested configurations, and would serve as the best representation of the RF to compare against the LSTM-SVM

Support Vector Machine (Taylor et. al., 2015)

Lastly, the third and final model of Taylor et. al. (2015) is the Support Vector Machine. Similar to other models with SVM, the study by Hossain et. al. (2022) and the LSTM-SVM model, the parameters C and Gamma were employed in the experimental process to figure out the best configuration or combination of hyperparameters that will achieve the finest results. The following is an in-depth discussion and analysis of the results gathered from the process.

Table 119

Confusion Matrix of C = 1 and Gamma = 0.001

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	7155	0
	Artifact (1)	4955	418
ajeric_expert2	*Non-Artifact (0)*	11368	2
	Artifact (1)	1358	43
ekamis_expert2	*Non-Artifact (0)*	12416	12
	Artifact (1)	1170	21
iguted_expert1	*Non-Artifact (0)*	11646	9
	Artifact (1)	1377	103
inefoh_expert1	*Non-Artifact (0)*	12511	6
	Artifact (1)	805	55
otafeh_expert1	*Non-Artifact (0)*	11321	0
	Artifact (1)	581	20
oxused_expert2	*Non-Artifact (0)*	9899	11
	Artifact (1)	1932	112
pqbqpr_expert2	*Non-Artifact (0)*	14297	28
	Artifact (1)	924	66
uhepah_expert1	*Non-Artifact (0)*	11108	0
	Artifact (1)	926	10
ukudab_expert2	*Non-Artifact (0)*	12712	1
	Artifact (1)	1580	4

Table 119 represents the total number of actual and predicted instances in the SVM model that was optimized using C of 1 and Gamma of 0.001. These instances were calculated using the formula of accuracy, F1-score, ROC-AUC, precision, and recall to quantify its performance in detecting artifacts in an EDA signal. The configuration resulted in 0.8784, 0.8312, 0.7905, 0.8926, and 0.8784, in the same order with its respective metric.

Table 120

Performance of C = 1 and Gamma = 0.001

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.6045	0.4861	0.8537	0.7663	0.6045
ajeric_expert2	0.8935	0.8466	0.7704	0.9001	0.8935
ekamis_expert2	0.9132	0.8741	0.6927	0.8896	0.9132
iguted_expert1	0.8945	0.8521	0.8331	0.8971	0.8945
inefoh_expert1	0.9394	0.9140	0.7803	0.9371	0.9394
otafeh_expert1	0.9513	0.9291	0.7722	0.9536	0.9513
oxused_expert2	0.8375	0.7726	0.8352	0.8493	0.8375
pqbqpr_expert2	0.9378	0.9131	0.7525	0.9240	0.9378
uhepah_expert1	0.9231	0.8870	0.8637	0.9290	0.9231
ukudab_expert2	0.8894	0.8377	0.7511	0.8795	0.8894
	0.8784	**0.8312**	**0.7905**	**0.8926**	**0.8784**

Table 121

Confusion Matrix of C = 1 and Gamma = 0.01

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	7116	39
	Artifact (1)	3941	1432
ajeric_expert2	*Non-Artifact (0)*	11354	16
	Artifact (1)	1312	89
ekamis_expert2	*Non-Artifact (0)*	12406	22
	Artifact (1)	1142	49
iguted_expert1	*Non-Artifact (0)*	11624	31
	Artifact (1)	1262	218
inefoh_expert1	*Non-Artifact (0)*	12501	16
	Artifact (1)	768	92
otafeh_expert1	*Non-Artifact (0)*	11314	7
	Artifact (1)	549	52
oxused_expert2	*Non-Artifact (0)*	9894	16
	Artifact (1)	1894	150
pqbqpr_expert2	*Non-Artifact (0)*	14263	62
	Artifact (1)	862	128
uhepah_expert1	*Non-Artifact (0)*	11108	0
	Artifact (1)	885	51
ukudab_expert2	*Non-Artifact (0)*	12684	29
	Artifact (1)	1548	36

Table 122

Performance of C = 1 and Gamma = 0.01

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.6823	0.6258	0.8581	0.7851	0.6823
ajeric_expert2	0.8960	0.8541	0.7867	0.8911	0.8960
ekamis_expert2	0.9145	0.8784	0.7016	0.8960	0.9145
iguted_expert1	0.9016	0.8690	0.8199	0.8991	0.9016
inefoh_expert1	0.9414	0.9195	0.7748	0.9363	0.9414
otafeh_expert1	0.9534	0.9348	0.7944	0.9501	0.9534
oxused_expert2	0.8402	0.7792	0.8665	0.8503	0.8402
pqbqpr_expert2	0.9397	0.9200	0.7807	0.9256	0.9397
uhepah_expert1	0.9265	0.8950	0.8724	0.9319	0.9265
ukudab_expert2	0.8897	0.8420	0.7658	0.8539	0.8897
	0.8885	**0.8518**	**0.8021**	**0.8919**	**0.8885**

A configuration of 1 and 0.01 for the C and Gamma parameters, respectively, to optimize the SVM model for automatic artifact detection. This generated performance of 0.8885, 0.8518, 0.8021, 0.8919, and 0.8885 in accuracy, F1-score, ROC-AUC, precision, recall. A tabular representation of the predicted and actual cases is shown in Table 121, while Table 122 presents the comparison of the results per metric.

Table 123

Confusion Matrix of C = 1 and Gamma = 0.1

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	6979	176
	Artifact (1)	3022	2351
ajeric_expert2	*Non-Artifact (0)*	11336	34
	Artifact (1)	1238	163
ekamis_expert2	*Non-Artifact (0)*	12402	26
	Artifact (1)	1132	59
iguted_expert1	*Non-Artifact (0)*	11608	47
	Artifact (1)	1200	280
inefoh_expert1	*Non-Artifact (0)*	12484	33
	Artifact (1)	732	128
otafeh_expert1	*Non-Artifact (0)*	11313	8
	Artifact (1)	537	64
oxused_expert2	*Non-Artifact (0)*	9878	32
	Artifact (1)	1829	215
pqbqpr_expert2	*Non-Artifact (0)*	14179	146
	Artifact (1)	803	187
uhepah_expert1	*Non-Artifact (0)*	11081	27
	Artifact (1)	723	213
ukudab_expert2	*Non-Artifact (0)*	12674	39
	Artifact (1)	1523	61

Table 124

Performance of C = 1 and Gamma = 0.1

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7447	0.7199	0.8533	0.7976	0.7447
ajeric_expert2	0.9004	0.8654	0.8358	0.8934	0.9004
ekamis_expert2	0.9150	0.8799	0.7033	0.8969	0.9150
iguted_expert1	0.9051	0.8770	0.7978	0.9007	0.9051
inefoh_expert1	0.9428	0.9240	0.7578	0.9350	0.9428
otafeh_expert1	0.9543	0.9368	0.7691	0.9514	0.9543
oxused_expert2	0.8443	0.7897	0.8580	0.8483	0.8443
pqbqpr_expert2	0.9380	0.9233	0.7905	0.9215	0.9380
uhepah_expert1	0.9377	0.9202	0.8679	0.9348	0.9377
ukudab_expert2	0.8907	0.8456	0.7681	0.8614	0.8907
	0.8973	**0.8682**	**0.8002**	**0.8941**	**0.8973**

Table 123 presents the confusion matrixes generated from hyperparameter tuning using the parameters C = 1, and Gamma 0.1. This setup achieved an accuracy of 0.9099, an F1-score of 0.8682, a ROC-AUC of 0.8002, a precision of 0.8941, and a recall of 0.8973, as shown in Table 124.

Table 125

Confusion Matrix of C = 1 and Gamma = 1

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	6902	253
	Artifact (1)	2835	2538
ajeric_expert2	*Non-Artifact (0)*	11321	49
	Artifact (1)	1198	203
ekamis_expert2	*Non-Artifact (0)*	12387	41
	Artifact (1)	1101	90
iguted_expert1	*Non-Artifact (0)*	11572	83
	Artifact (1)	1109	371
inefoh_expert1	*Non-Artifact (0)*	12423	94
	Artifact (1)	658	202
otafeh_expert1	*Non-Artifact (0)*	11306	15
	Artifact (1)	513	88
oxused_expert2	*Non-Artifact (0)*	9855	55
	Artifact (1)	1709	335
pqbqpr_expert2	*Non-Artifact (0)*	14035	290
	Artifact (1)	749	241
uhepah_expert1	*Non-Artifact (0)*	11064	44
	Artifact (1)	696	240
ukudab_expert2	*Non-Artifact (0)*	12660	53
	Artifact (1)	1486	98

Table 126

Performance of C = 1 and Gamma = 1

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7535	0.7334	0.8497	0.7948	0.7535
ajeric_expert2	0.9024	0.8708	0.8194	0.8935	0.9024
ekamis_expert2	0.9161	0.8842	0.7697	0.8981	0.9161
iguted_expert1	0.9093	0.8871	0.7799	0.9018	0.9093
inefoh_expert1	0.9438	0.9307	0.7980	0.9325	0.9438
otafeh_expert1	0.9557	0.9405	0.7562	0.9514	0.9557
oxused_expert2	0.8524	0.8080	0.8272	0.8534	0.8524
pqbqpr_expert2	0.9322	0.9225	0.8033	0.9173	0.9322
uhepah_expert1	0.9386	0.9230	0.8666	0.9334	0.9386
ukudab_expert2	0.8924	0.8508	0.7380	0.8677	0.8924
	0.8996	**0.8751**	**0.8008**	**0.8944**	**0.8996**

Hyperparameter tuning was performed using a combination of parameters: C set to 1 and Gamma to 1, to identify the optimal configuration for achieving the best results. This configuration yielded an accuracy of 0.8996, an F1-score of 0.8751, a ROC-AUC of 0.8008, a precision of 0.8944, and a recall of 0.8996, as summarized in Table 126.

Table 127

Confusion Matrix of C = 10 and Gamma = 0.001

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	7128	27
	Artifact (1)	4122	1251
ajeric_expert2	*Non-Artifact (0)*	11357	13
	Artifact (1)	1324	77
ekamis_expert2	*Non-Artifact (0)*	12408	20
	Artifact (1)	1142	49
iguted_expert1	*Non-Artifact (0)*	11619	36
	Artifact (1)	1257	223
inefoh_expert1	*Non-Artifact (0)*	12498	19
	Artifact (1)	765	95
otafeh_expert1	*Non-Artifact (0)*	11313	8
	Artifact (1)	543	58
oxused_expert2	*Non-Artifact (0)*	9889	21
	Artifact (1)	1828	216
pqbqpr_expert2	*Non-Artifact (0)*	14271	54
	Artifact (1)	859	131
uhepah_expert1	*Non-Artifact (0)*	11108	0
	Artifact (1)	893	43
ukudab_expert2	*Non-Artifact (0)*	12691	22
	Artifact (1)	1551	33

Table 128

Performance of C = 10 and Gamma = 0.001

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.6688	0.6037	0.8542	0.7817	0.6688
ajeric_expert2	0.8953	0.8521	0.7933	0.8912	0.8953
ekamis_expert2	0.9147	0.8785	0.7078	0.8977	0.9147
iguted_expert1	0.9016	0.8695	0.8261	0.8977	0.9016
inefoh_expert1	0.9414	0.9198	0.7870	0.9353	0.9414
otafeh_expert1	0.9538	0.9358	0.7731	0.9504	0.9538
oxused_expert2	0.8453	0.7905	0.8711	0.8555	0.8453
pqbqpr_expert2	0.9404	0.9208	0.7824	0.9280	0.9404
uhepah_expert1	0.9259	0.8935	0.8639	0.9314	0.9259
ukudab_expert2	0.8900	0.8418	0.7723	0.8588	0.8900
	0.8877	**0.8506**	**0.8031**	**0.8928**	**0.8877**

Table 127 represents the total number of actual and predicted instances in the GBT model that was optimized using C of 10 and Gamma of 0.001. These instances were calculated using the formula of accuracy, F1-score, ROC-AUC, precision, and recall to quantify its performance in detecting artifacts in an EDA signal. The configuration resulted in 0.8877, 0.8506, 0.8031, 0.8928, and 0.8877, in the same order with its respective metrics.

Table 129

Confusion Matrix of C = 10 and Gamma = 0.01

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	6980	175
	Artifact (1)	3017	2356
ajeric_expert2	*Non-Artifact (0)*	11326	44
	Artifact (1)	1228	173
ekamis_expert2	*Non-Artifact (0)*	12385	43
	Artifact (1)	1095	96
iguted_expert1	*Non-Artifact (0)*	11598	57
	Artifact (1)	1169	311
inefoh_expert1	*Non-Artifact (0)*	12488	29
	Artifact (1)	718	142
otafeh_expert1	*Non-Artifact (0)*	11308	13
	Artifact (1)	526	75
oxused_expert2	*Non-Artifact (0)*	9874	36
	Artifact (1)	1785	259
pqbqpr_expert2	*Non-Artifact (0)*	14243	82
	Artifact (1)	800	190
uhepah_expert1	*Non-Artifact (0)*	11092	16
	Artifact (1)	767	169
ukudab_expert2	*Non-Artifact (0)*	12646	67
	Artifact (1)	1475	109

Table 130

Performance of C = 10 and Gamma = 0.01

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7452	0.7205	0.8632	0.7980	0.7452
ajeric_expert2	0.9004	0.8664	0.8189	0.8907	0.9004
ekamis_expert2	0.9164	0.8851	0.7089	0.8988	0.9164
iguted_expert1	0.9067	0.8807	0.7939	0.9013	0.9067
inefoh_expert1	0.9442	0.9262	0.7653	0.9382	0.9442
otafeh_expert1	0.9548	0.9385	0.7659	0.9503	0.9548
oxused_expert2	0.8477	0.7969	0.8767	0.8522	0.8477
pqbqpr_expert2	0.9424	0.9267	0.7931	0.9308	0.9424
uhepah_expert1	0.9350	0.9143	0.8760	0.9336	0.9350
ukudab_expert2	0.8921	0.8518	0.7786	0.8649	0.8921
	0.8985	**0.8707**	**0.8040**	**0.8959**	**0.8985**

The hyperparameter tuning was performed using a combination of parameters: C set to 10, and Gamma set to 0.01, to identify the optimal configuration for achieving the best results. This configuration produced an accuracy of 0.8985, an F1-score of 0.8707, a ROC-AUC of 0.8040, a precision of 0.8959, and a recall of 0.8985, as summarized in Table 130.

Table 131

Confusion matrixes of C = 10 and Gamma = 0.01

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	6803	352
	Artifact (1)	2561	2812
ajeric_expert2	*Non-Artifact (0)*	11316	54
	Artifact (1)	1162	239
ekamis_expert2	*Non-Artifact (0)*	12380	48
	Artifact (1)	1073	118
iguted_expert1	*Non-Artifact (0)*	11577	78
	Artifact (1)	1120	360
inefoh_expert1	*Non-Artifact (0)*	12461	56
	Artifact (1)	689	171
otafeh_expert1	*Non-Artifact (0)*	11305	16
	Artifact (1)	521	80
oxused_expert2	*Non-Artifact (0)*	9860	50
	Artifact (1)	1695	349
pqbqpr_expert2	*Non-Artifact (0)*	14156	169
	Artifact (1)	756	234
uhepah_expert1	*Non-Artifact (0)*	11029	79
	Artifact (1)	627	309
ukudab_expert2	*Non-Artifact (0)*	12638	75
	Artifact (1)	1434	150

Table 131 presents the outcome of the hyperparameter tuning of Taylor et. al. (2015) SVM model, using the values 10 and 0.01 for its C and Gamma parameter,

respectively. It produced accuracy of 0.9027, f1-score of 0.8796, ROC-AUC of 0.8065, precision of 0.8971, and recall of 0.9027.

Table 132

Performance of C = 10 and Gamma = 0.1

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7675	0.7529	0.8467	0.7961	0.7675
ajeric_expert2	0.9048	0.8759	0.8435	0.8969	0.9048
ekamis_expert2	0.9177	0.8882	0.7323	0.9019	0.9177
iguted_expert1	0.9088	0.8860	0.7801	0.9017	0.9088
inefoh_expert1	0.9443	0.9288	0.7746	0.9351	0.9443
otafeh_expert1	0.9550	0.9391	0.7517	0.9498	0.9550
oxused_expert2	0.8540	0.8105	0.8751	0.8570	0.8540
pqbqpr_expert2	0.9396	0.9275	0.8028	0.9255	0.9396
uhepah_expert1	0.9414	0.9300	0.8729	0.9346	0.9414
ukudab_expert2	0.8945	0.8575	0.7857	0.8725	0.8945
	0.9027	**0.8796**	**0.8065**	**0.8971**	**0.9027**

Table 133

Confusion Matrix of C = 100 and Gamma = 0.001

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	7042	113
	Artifact (1)	3418	1955
ajeric_expert2	*Non-Artifact (0)*	11346	24
	Artifact (1)	1257	144
ekamis_expert2	*Non-Artifact (0)*	12389	39
	Artifact (1)	1098	93
iguted_expert1	*Non-Artifact (0)*	11588	67
	Artifact (1)	1153	327
inefoh_expert1	*Non-Artifact (0)*	12493	24
	Artifact (1)	722	138
otafeh_expert1	*Non-Artifact (0)*	11307	14
	Artifact (1)	524	77
oxused_expert2	*Non-Artifact (0)*	9874	36
	Artifact (1)	1699	345
pqbqpr_expert2	*Non-Artifact (0)*	14263	62
	Artifact (1)	802	188
uhepah_expert1	*Non-Artifact (0)*	11104	4
	Artifact (1)	830	106
ukudab_expert2	*Non-Artifact (0)*	12649	64
	Artifact (1)	1495	89

Table 133 shows the results from the hyperparameter tuning of Taylor et. al. (2015) SVM model, using the values 10 and 0.001 for its C and Gamma parameter,

respectively. It produced accuracy of 0.8961, f1-score of 0.8668, ROC-AUC of 0.8011, precision of 0.8964, and recall of 0.8961.

Table 134

Performance of C = 100 and Gamma = 0.001

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7182	0.6820	0.8552	0.7899	0.7182
ajeric_expert2	0.8997	0.8629	0.8159	0.8955	0.8997
ekamis_expert2	0.9165	0.8848	0.7146	0.8999	0.9165
iguted_expert1	0.9071	0.8823	0.7998	0.9005	0.9071
inefoh_expert1	0.9442	0.9259	0.7727	0.9394	0.9442
otafeh_expert1	0.9549	0.9387	0.7343	0.9502	0.9549
oxused_expert2	0.8549	0.8107	0.8877	0.8621	0.8549
pqbqpr_expert2	0.9436	0.9275	0.7806	0.9342	0.9436
uhepah_expert1	0.9308	0.9047	0.8550	0.9330	0.9308
ukudab_expert2	0.8910	0.8489	0.7955	0.8597	0.8910
	0.8961	**0.8668**	**0.8011**	**0.8964**	**0.8961**

Table 135

Confusion Matrix of C = 100 and Gamma = 0.001

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	6757	398
	Artifact (1)	2430	2943
ajeric_expert2	*Non-Artifact (0)*	11299	71
	Artifact (1)	1131	270
ekamis_expert2	*Non-Artifact (0)*	12369	59
	Artifact (1)	1045	146
iguted_expert1	*Non-Artifact (0)*	11580	75
	Artifact (1)	1125	355
inefoh_expert1	*Non-Artifact (0)*	12478	39
	Artifact (1)	689	171
otafeh_expert1	*Non-Artifact (0)*	11304	17
	Artifact (1)	517	84
oxused_expert2	*Non-Artifact (0)*	9861	49
	Artifact (1)	1672	372
pqbqpr_expert2	*Non-Artifact (0)*	14229	96
	Artifact (1)	767	223
uhepah_expert1	*Non-Artifact (0)*	11052	56
	Artifact (1)	671	265
ukudab_expert2	*Non-Artifact (0)*	12606	107
	Artifact (1)	1381	203

Table 135 and 136 illustrates the findings of the hyperparameter tuning of Taylor et. al. (2015) SVM model, using the values 10 and 0.01 for its C and Gamma parameter, respectively. It produced accuracy of 0.9044, f1-score of 0.8821, ROC-AUC of 0.8061, precision of 0.8989, and recall of 0.9044.

Table 136

Performance of C = 100 and Gamma = 0.01

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7743	0.7620	0.8641	0.7978	0.7743
ajeric_expert2	0.9059	0.8793	0.8418	0.8962	0.9059
ekamis_expert2	0.9189	0.8919	0.7044	0.9037	0.9189
iguted_expert1	0.9086	0.8855	0.7796	0.9018	0.9086
inefoh_expert1	0.9456	0.9297	0.7606	0.9391	0.9456
otafeh_expert1	0.9552	0.9397	0.7640	0.9500	0.9552
oxused_expert2	0.8560	0.8141	0.8858	0.8599	0.8560
pqbqpr_expert2	0.9437	0.9299	0.7920	0.9327	0.9437
uhepah_expert1	0.9396	0.9257	0.8703	0.9337	0.9396
ukudab_expert2	0.8959	0.8634	0.7989	0.8740	0.8959
	0.9044	**0.8821**	**0.8061**	**0.8989**	**0.9044**

Table 137

Confusion Matrix of C = 100 and Gamma = 0.1

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	6688	467
	Artifact (1)	2295	3078
ajeric_expert2	*Non-Artifact (0)*	11299	71
	Artifact (1)	1112	289
ekamis_expert2	*Non-Artifact (0)*	12366	62
	Artifact (1)	1032	159
iguted_expert1	*Non-Artifact (0)*	11564	91
	Artifact (1)	1075	405
inefoh_expert1	*Non-Artifact (0)*	12453	64
	Artifact (1)	665	195
otafeh_expert1	*Non-Artifact (0)*	11297	24
	Artifact (1)	502	99
oxused_expert2	*Non-Artifact (0)*	9845	65
	Artifact (1)	1592	452
pqbqpr_expert2	*Non-Artifact (0)*	14142	183
	Artifact (1)	738	252
uhepah_expert1	*Non-Artifact (0)*	11019	89
	Artifact (1)	585	351
ukudab_expert2	*Non-Artifact (0)*	12599	114
	Artifact (1)	1366	218

A configuration of 100 and 0.1 for the C and Gamma,, respectively, to optimize the SVM model for automatic artifact detection. This generated performance of 0.9061, 0.8863, 0.8117, 0.8994, and 0.9061 in accuracy, F1-score, ROC-AUC, precision, recall. A tabular representation of the predicted and actual cases is shown in Table 137, while Table 138 presents the comparison of the results per metric.

Table 138

Performance of C = 100 and Gamma = 0.1

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7795	0.7694	0.8465	0.7976	0.7795
ajeric_expert2	0.9074	0.8820	0.8419	0.8986	0.9074
ekamis_expert2	0.9197	0.8936	0.7406	0.9052	0.9197
iguted_expert1	0.9112	0.8909	0.7745	0.9039	0.9112
inefoh_expert1	0.9455	0.9315	0.8066	0.9367	0.9455
otafeh_expert1	0.9559	0.9418	0.7466	0.9498	0.9559
oxused_expert2	0.8614	0.8250	0.8816	0.8631	0.8614
pqbqpr_expert2	0.9399	0.9287	0.8038	0.9264	0.9399
uhepah_expert1	0.9440	0.9346	0.8731	0.9378	0.9440
ukudab_expert2	0.8965	0.8651	0.8019	0.8750	0.8965
	0.9061	**0.8863**	**0.8117**	**0.8994**	**0.9061**

Table 139

Confusion Matrix of C = 1000 and Gamma = 0.001

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	6880	275
	Artifact (1)	2827	2546
ajeric_expert2	*Non-Artifact (0)*	11322	48
	Artifact (1)	1178	223
ekamis_expert2	*Non-Artifact (0)*	12372	56
	Artifact (1)	1041	150
iguted_expert1	*Non-Artifact (0)*	11572	83
	Artifact (1)	1115	365
inefoh_expert1	*Non-Artifact (0)*	12485	32
	Artifact (1)	706	154
otafeh_expert1	*Non-Artifact (0)*	11303	18
	Artifact (1)	518	83
oxused_expert2	*Non-Artifact (0)*	9864	46
	Artifact (1)	1635	409
pqbqpr_expert2	*Non-Artifact (0)*	14260	65
	Artifact (1)	778	212
uhepah_expert1	*Non-Artifact (0)*	11086	22
	Artifact (1)	756	180
ukudab_expert2	*Non-Artifact (0)*	12607	106
	Artifact (1)	1395	189

Another hyperparameter tuning using the combination of parameters C with the value of 1000 and Gamma of 0.001 was applied to find the one configuration that produces the best results. This tuning resulted in accuracy, F1-score, ROC-AUC, precision, and recall with values of 0.9019, 0.8780, 0.8004, 0.8929, and 0.9019, as shown in Table 140.

Table 140

Performance of C = 1000 and Gamma = 0.001

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7524	0.7326	0.8571	0.7919	0.7524
ajeric_expert2	0.9040	0.8738	0.8306	0.8967	0.9040
ekamis_expert2	0.9195	0.8926	0.7188	0.9054	0.9195
iguted_expert1	0.9088	0.8863	0.7817	0.9011	0.9088
inefoh_expert1	0.9448	0.9278	0.7493	0.9389	0.9448
otafeh_expert1	0.9550	0.9395	0.7280	0.9494	0.9550
oxused_expert2	0.8594	0.8199	0.8921	0.8648	0.8594
pqbqpr_expert2	0.9450	0.9301	0.7870	0.9364	0.9450
uhepah_expert1	0.9354	0.9156	0.8519	0.9327	0.9354
ukudab_expert2	0.8950	0.8615	0.8079	0.8716	0.8950
	0.9019	**0.8780**	**0.8004**	**0.8989**	**0.9019**

Table 141

Confusion Matrix of C = 1000 and Gamma = 0.1

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	6624	531
	Artifact (1)	2209	3164
ajeric_expert2	*Non-Artifact (0)*	11295	75
	Artifact (1)	1082	319
ekamis_expert2	*Non-Artifact (0)*	12361	67
	Artifact (1)	1015	176
iguted_expert1	*Non-Artifact (0)*	11564	91
	Artifact (1)	1034	446
inefoh_expert1	*Non-Artifact (0)*	12432	85
	Artifact (1)	651	209
otafeh_expert1	*Non-Artifact (0)*	11290	31
	Artifact (1)	493	108
oxused_expert2	*Non-Artifact (0)*	9840	70
	Artifact (1)	1507	537
pqbqpr_expert2	*Non-Artifact (0)*	14131	194
	Artifact (1)	727	263
uhepah_expert1	*Non-Artifact (0)*	11021	87
	Artifact (1)	595	341
ukudab_expert2	*Non-Artifact (0)*	12574	139
	Artifact (1)	1321	263

Another hyperparameter tuning using the combination of parameters C with the value of 1000 and Gamma of 0.1 was applied to find the one configuration that produces the best results. This tuning resulted in accuracy, F1-score, ROC-AUC, precision, and recall with values of 0.9076, 0.8894, 0.8195, 0.9007, and 0.9076, as shown in Table 142.

Table 142

Performance of C = 1000 and Gamma = 0.1

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.7813	0.7725	0.8491	0.7955	0.7813
ajeric_expert2	0.9094	0.8859	0.8338	0.9013	0.9094
ekamis_expert2	0.9206	0.8958	0.7713	0.9066	0.9206
iguted_expert1	0.9144	0.8960	0.7809	0.9081	0.9144
inefoh_expert1	0.9450	0.9321	0.8354	0.9349	0.9450
otafeh_expert1	0.9560	0.9428	0.7585	0.9490	0.9560
oxused_expert2	0.8681	0.8368	0.8773	0.8702	0.8681
pqbqpr_expert2	0.9399	0.9293	0.8066	0.9268	0.9399
uhepah_expert1	0.9434	0.9335	0.8730	0.9370	0.9434
ukudab_expert2	0.8979	0.8698	0.8096	0.8772	0.8979
	0.9076	**0.8894**	**0.8195**	**0.9007**	**0.9076**

LSTM-CNN (Llanes-Jurado et. al., 2023)

Finally, this study compares the results with those from the benchmark research by Llanes-Jurado et al. (2023), which utilized an LSTM-CNN model, forming the foundation of the current investigation. The LSTM-CNN model demonstrated strong performance across various metrics, achieving scores of 0.8832 for accuracy, 0.8845 for F1-score, 0.8838 for ROC-AUC, 0.9054 for precision, and 0.8832 for recall.

Table 143

Confusion Matrix of Every Subject

Subject	Actual	Predicted	
		Non-Artifact (0)	*Artifact (1)*
afegip_expert1	*Non-Artifact (0)*	2820	4325
	Artifact (1)	299	5074
ajeric_expert2	*Non-Artifact (0)*	10358	1002
	Artifact (1)	437	964
ekamis_expert2	*Non-Artifact (0)*	11904	514
	Artifact (1)	650	541
iguted_expert1	*Non-Artifact (0)*	10939	706
	Artifact (1)	544	936
inefoh_expert1	*Non-Artifact (0)*	12247	260
	Artifact (1)	390	470
otafeh_expert1	*Non-Artifact (0)*	11183	128
	Artifact (1)	301	300
oxused_expert2	*Non-Artifact (0)*	8829	1070
	Artifact (1)	340	1705
pqbqpr_expert2	*Non-Artifact (0)*	13847	467
	Artifact (1)	489	502
uhepah_expert1	*Non-Artifact (0)*	10246	852
	Artifact (1)	260	676
ukudab_expert2	*Non-Artifact (0)*	11079	1624
	Artifact (1)	489	1095

Table 144

Performance of the LSTM-CNN Model

Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall
afegip_expert1	0.6306	0.6085	0.7938	0.7478	0.6306
ajeric_expert2	0.8872	0.8953	0.8889	0.9080	0.8872
ekamis_expert2	0.9145	0.9121	0.8364	0.9101	0.9145
iguted_expert1	0.9048	0.9069	0.8875	0.9095	0.9048
inefoh_expert1	0.9514	0.9495	0.9129	0.9482	0.9514
otafeh_expert1	0.9640	0.9611	0.9113	0.9600	0.9640
oxused_expert2	0.8819	0.8886	0.9281	0.9032	0.8819
pqbqpr_expert2	0.9375	0.9372	0.8793	0.9369	0.9375
uhepah_expert1	0.9076	0.9174	0.9258	0.9338	0.9076
ukudab_expert2	0.8521	0.8682	0.8739	0.8962	0.8521
	0.8832	**0.8845**	**0.8838**	**0.9054**	**0.8832**

These results provide a comprehensive overview of the model's effectiveness, with each metric highlighting different aspects of its performance, such as its ability to correctly classify instances, balance precision and recall, and distinguish between classes. The comparison underscores the similarities and differences between the LSTM-CNN model and the proposed model in this study, offering insights into the relative strengths and areas for improvement in the context of artifact detection in EDA data.

With the use of confusion matrices that represent results from each subject in the test set, the performance of the LSTM-SVM compared to other models can be seen, in terms of the metrics which are accuracy, precision, F1-score, ROC-AUC, and recall.

Performance of the LSTM-SVM Model Compared to other Existing Models

Table 145

Performance of the models during testing phase in accordance to ROC-AUC, Recall, Precision, F1-Score and Accuracy

Artifact Detection Models	Accuracy	F1-Score	ROC-AUC	Precision	Recall
LSTM-SVM (1, 0.1)	**0.8953**	**0.8840**	**0.7836**	**0.8867**	**0.8953**
Taylor RF (600, 30)	0.9092	0.9000	0.8432	0.9017	0.9092
Taylor SVM (1000, 0.1)	0.9076	0.8894	0.8195	0.9007	0.9076
Taylor LR (100)	0.8977	**0.8717**	**0.7719**	0.8955	0.8977
Hossain LR (10)	0.9072	0.8913	0.8201	0.8996	0.9072
Hossain GBT (600, 0.01, 10)	0.9105	0.9011	0.8434	0.9039	0.9105
Hossain SVM (100, 0.01)	0.9111	0.8981	0.7926	0.9034	0.9111
Llanes-Jurado LSTM-CNN	**0.8832**	0.8845	0.8838	0.9054	**0.8832**

Table 145 shows the following results gathered during the testing period of the study with the same metrics used to evaluate and measure the performance of the automatic detection of artifacts in EDA signals. The proposed LSTM-SVM model achieved a ROC-AUC of 0.7836, precision of 0.8867, F1-score of 0.8840, accuracy of 0.8953, and recall of 0.8953. While these results demonstrate balanced performance, they are slightly outperformed by the other models in certain metrics. For instance, the SVM model by Hossain et al. (2022) achieved a higher Accuracy than all other models of 0.9111, and a recall also of 0.9111 indicating its ability to detect positive labels in the signal which are the Artifacts. Similarly, the GBT model by Hossain et al. (2022) was recorded as the superior model in terms of F1-Score of 0.9011, in that a relatively high F1-Score indicates that the model can classify between artifacts and non-artifacts especially if the dataset is imbalance.

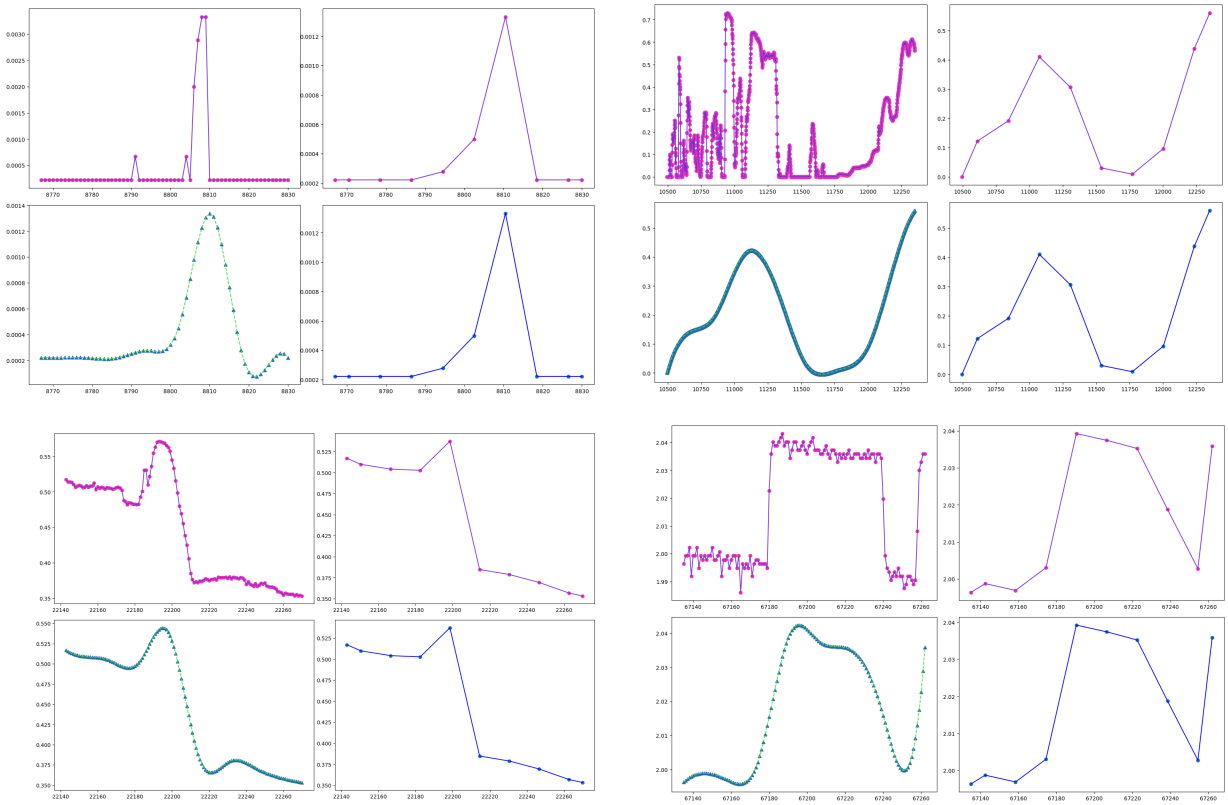
On the other hand, the LSTM-CNN model by Llanes-Jurado et al. (2023) exhibited the highest ROC-AUC of 0.8838 and a precision of 0.9054, reflecting its strength in minimizing false positives. However, its F1-Score is 0.8845 and its accuracy

is 0.8832 were comparable to the proposed LSTM-SVM. Compared to the traditional methods such as the Logistic Regression and Support Vector Machines across different studies, the LSTM-SVM model consistently demonstrated better ROC-AUC and precision. For instance, Logistic Regression by Taylor et al. (2015) achieved a lower ROC-AUC of 0.8717 and F1-Score 0.7719 compared to that of the LSTM-SVM, similarly the SVM also by Taylor et al. (2015) also achieved a lower F1-Score of 0.8796 than the LSTM-SVM despite having competitive precision and accuracy.

Overall, while the proposed LSTM-SVM model performs stable and achieves a good balance between metrics, models like Random Forest and Gradient Boost outperform it in terms of F1-Score and accuracy. The LSTM-SVM, however, remains a competitive choice, particularly in scenarios where high precision is crucial.

Visualization of the EDA Signal Detection and Correction

Figure 17. EDA signals with detected artifacts along with smoothed and corrected signal



The figures illustrate the four phases of electrodermal activity (EDA) signal processing. In the top-left, the raw signal is shown with detected artifacts or noise, highlighted by pink dots, exhibiting irregularities and a sharp drop that indicate the presence of artifacts. The top-right panel presents a smoothed version of this unclean signal, where some irregularities are partially reduced. Moving to the bottom-left, the signal appears more refined, with most abrupt changes and noise removed, resulting in a smoother representation. Finally, the bottom-right panel shows the fully corrected signal, which has been simplified and filtered to produce a clean version of the original data.

CHAPTER 5

SUMMARY OF FINDINGS, CONCLUSIONS, AND RECOMMENDATIONS

This chapter presents an overview of the findings and summary of the result presented in the previous chapter. It evaluates whether the model has significant performance improvements compared to the other existing models and discusses the implication of these findings. Furthermore, conclusions of the study and recommendations are discussed for future research and studies.

Summary of Findings

This study presents a comprehensive discussion of the results from different models, including the LSTM-SVM, in its performance and efficiency in detecting artifacts or noises in EDA signals. Experimentation was done to find the best result from hyperparameter tuning of C and Gamma hyperparameters. Afterwards, the best result was compared to the best outcome of tuning or optimization of the existing models. The following provides the summary of the results, aligned with the statement of the problem of the study:

1. The study explored various combinations of hyperparameters C and Gamma for SVM-based model to optimize artifact detection in electrodermal activity (EDA) signals. The testing values of C (1, 10, 100) and Gamma (0.1, 0.5, and 1), but the configuration C = 1 and Gamma = 0.1 emerged as the best one. This setup achieved a balanced performance across metrics, including an accuracy of 0.8953, an F1-score of 0.8840, a ROC-AUC of 0.7836, precision of 0.8867 and recall of 0.8953. Although these demonstrated slightly higher results but not significant when compared for instance to the LSTM-CNN and Taylor et al. (2015) these results still demonstrated that relative to other models LSTM-SVM elicits

still an ability to distinguish artifacts from non-artifacts in electrodermal activity (EDA) signals even if lower by a small margin (e.g. 1% - 2%) than most models in terms of most performance metrics.

The optimal configuration was benchmarked against state-of-the-art methods, including Random Forest, Logistic Regression, Gradient Boosted Trees, and hybrid LSTM-CNN models. While the performance metrics of the LSTM-SVM model were competitive, other state-of-the-art methods, like RF and GBT, slightly surpassed it in accuracy and precision. The study underscores the significance of fine-tuning hyperparameters to enhance model performance and presents C = 1, Gamma = 0.1 as a baseline for future comparative studies in automatic artifact detection.

2. The experimentation results underscore the superior performance of the LSTM-SVM model across key metrics, including ROC-AUC, precision, F1-Score, accuracy, and recall. During the testing phase, the LSTM-SVM model achieved a ROC-AUC of 0.7836, precision of 0.8867, F1-score of 0.8840, accuracy of 0.8953, and recall of 0.8953, while models such as Support Vector Machine (Hossain et al., 2022) and Gradient Boosted Tree (Hossain et al., 2022) outperformed the all other models and the LSTM-SVM in specific metrics, such as F1-Score, Accuracy, and Recall even though these models lack the flexibility and reliability required for processing sequential data. Similarly, the LSTM-CNN model (Llanes-Jurado et al., 2023) achieved the highest ROC-AUC of 0.8838 and the highest precision of 0.9054, but its performance in F1-Score and accuracy did not surpass that of the SVM and or GBT models by Hossain et al. (2022). Moreover, traditional models like Logistic Regression from the study of Taylor et al. (2015) slightly underperformed when compared to the LSTM-SVM, with lower

ROC-AUC and F1-score indicating its less adaptability to the complex temporal nature of EDA signals.

Overall, the summary of findings suggests the positive effect of using hybrid techniques like LSTM-SVM in solving the gaps of the existing models in terms of artifact detection. The LSTM-SVM albeit it underperforms relative to other models like RF, GBT it outperforms slightly typical models used such as LogisticRegression showing that it addressed the gap of traditional machine learning and moreover opened a path to how utilizing hybridized sequence models and traditional ML models can help address complexities of deep learning techniques that are usually utilized in classifying artifacts and non-artifacts in an electrodermal activity signal. The model provided a balanced performance across the different metrics, although a small margin has been slightly outperformed by the other existing models. The outcome highlights the performance of the LSTM-SVM model compared to existing models and may also serve as a guide for using hybrid architectures in studies concerning time-series data.

Conclusions

The study aimed to solve the gap in detecting the noises, or "artifacts" found in electrodermal activity signals in order to provide reliable data in detecting stress using this type of physiological signals. The researchers hybridized LSTM and SVM to address the limitation in existing artifact detection models. These techniques incorporated the strengths of machine learning and deep learning, in order to present a much efficient and effective classification of the signal. The research focused on answering the questions related to the best C and Gamma values that provided the best results and the comparison of the existing detection models to LSTM-SVM, in terms of its performance. Considering all these, the researchers come up with the following conclusion:

1. In order to provide the best setup of the model that would aid in producing the best outcome, hyperparameter optimization or tuning helps the model to provide the optimal outcome. Hyperparameter tuning using different values of C and Gamma parameters helped the SVM component of the hybridized LSTM-SVM to deliver the best results, in terms of its performance in detecting artifacts in electrodermal activity signals.
2. LSTM-SVM, in terms of performance according to its metrics, which are accuracy, precision, F1-score, recall, and ROC-AUC, shows promising results in detecting artifacts from non-artifacts in an electrodermal activity signal. It shows a higher accuracy and recall from the baseline LSTM-CNN model of Llanes-Jurado et. al. (2023) and higher F1-score and ROC-AUC from Logistic Regression of Taylor et. al. (2015). These results illustrate that the hybridized model outlines positive results in classifying EDA signals into artifacts and non-artifacts, for the purpose of providing a clean signal to properly detect stress from.

In conclusion, although the LSTM-SVM underperforms relative to other models in terms of ROC-AUC, F1-Score, but outperforms relative to other models slightly in terms of Accuracy, Recall it still provides great outcomes in detecting artifacts in EDA signals, but there remains a room to improve the system, in order to create a better system for classification and correction of the mentioned signals. This study emphasizes the prospect of using hybridized machine learning and deep learning models in order to cater to the limitations and challenges of detecting noises in electrodermal activity signals.

Recommendations

Based on the findings of this study, several recommendations are proposed to further enhance the hybrid LSTM-SVM model and its application in detecting artifacts in Electrodermal Activity (EDA) signals:

- SVM component of the LSTM-SVM can be further improved by providing a larger pool of hyperparameter configuration values to use in tuning other than the utilized values in this study (e.g. 1, 10, 100, and 1000 & 0.001, 0.01, 0.1, and 1 for C and gamma hyperparameters as opposed to 1, 10, and 100 & 0.001, 0.01, 0.1, and 1). This could aid in allowing more evaluation and representation, and could help produce the best results from the model.
- It is also recommended to test the model on larger, more diverse, and more balanced dataset to ensure its reliability and applicability, while examining its adaptability across different scenarios.
- Other recommendations in the domain of model optimization would include using a much powerful machine for these computationally intensive tasks of training, as SVM based models for instance can entail long and intensive training time i.e. 3 days for using one hyperparameter configuration or permutation.
- Physical and real-time implementation of the model, specifically in using wearable devices to gather electrodermal activity signals is another area for development. Physically-gathered signals could provide data significantly different from datasets, and it could further develop the model in its detecting and classifying tasks.

By addressing these recommendations, future research can advance the field of biomedical signal analysis and contribute to the development of more accurate, reliable, and practical stress detection systems.

REFERENCES

Agarap, A. F. (2018). *A Neural Network Architecture Combining Gated Recurrent Unit(GRU) and Support Vector Machine (SVM) for Intrusion Detection in Network Traffic Data*. https://doi.org/10.1145/3195106.3195117

Agarwal, C., Kaur, I., & Yadav, S. (2022). Hybrid CNN-SVM Model for Face Mask Detector to Protect from COVID-19. In *Lecture notes in computational vision and biomechanics* (pp. 419–426). https://doi.org/10.1007/978-981-19-0151-5_35

Agrawal, J., Gupta, M., & Garg, H. (2021). Early Stress Detection and Analysis using EEG signals in Machine Learning Framework. *IOP Conference Series. Materials Science and Engineering*, *1116*(1), 012134. https://doi.org/10.1088/1757-899x/1116/1/012134

Ahuja, R., & Banga, A. (2019). Mental Stress Detection in University Students using Machine Learning Algorithms. *Procedia Computer Science*, *152*, 349–353. https://doi.org/10.1016/j.procs.2019.05.007

Albertetti, F., Simalastar, A., & Rizzotti-Kaddouri, A. (2021). Stress Detection with Deep Learning Approaches Using Physiological Signals. In *Springer eBooks* (pp. 95–111). https://doi.org/10.1007/978-3-030-69963-5_7

Alfaras, M., Primett, W., Umair, M., Windlin, C., Karpashevich, P., Chalabianloo, N., Bowie, D., Sas, C., Sanches, P., Höök, K., Ersoy, C., & Gamboa, H. (2020). Biosensing and Actuation—Platforms coupling body Input-Output modalities for affective technologies. *Sensors*, *20*(21), 5968. https://doi.org/10.3390/s20215968

Alnazly, E., Khraisat, O. M., Al-Bashaireh, A. M., & Bryant, C. L. (2021).Anxiety,depression, stress, fear and social support during COVID-19 pandemic among Jordanian healthcare workers. *PloS One*, *16*(3), e0247679. https://doi.org/10.1371/journal.pone.0247679

AlShorman, O., Masadeh, M., Heyat, M. B. B., Akhtar, F., Almahasneh, H., Ashraf, G. M., & Alexiou, A. (2022). Frontal lobe real-time EEG analysis using machine learning techniques for mental stress detection. *Journal of Integrative Neuroscience*, *21*(1), 020. https://doi.org/10.31083/j.jin2101020

Archana, V. R., & Devaraju, B. M. (2020, August). *View of stress detection using machine learning algorithms*. https://journal.ijresm.com/index.php/ijresm/article/view/171/154

Bach, D. R., & Friston, K. J. (2012). Model-based analysis of skin conductance responses: Towards causal models in psychophysiology. *Psychophysiology*, *50*(1), 15–22. https://doi.org/10.1111/j.1469-8986.2012.01483.x

Bakker, J., Pechenizkiy, M., & Sidorova, N. (2011). *What's Your Current Stress Level? Detection of Stress Patterns from GSR Sensor Data. 2011 IEEE 11th International Conference on Data Mining Workshops.* doi:10.1109/icdmw.2011.178

Banganho, A., Santos, M., & Da Silva, H. P. (2022). Electrodermal activity: Fundamental principles, measurement, and application. *IEEE Potentials*, *41*(5), 35–43. https://doi.org/10.1109/mpot.2020.2983381

Bartolomé-Tomás, A., Sánchez-Reolid, R., Fernández-Sotos, A., Latorre, J. M., & Fernández-Caballero, A. (2020). Arousal Detection in Elderly People from Electrodermal Activity Using Musical Stimuli. *Sensors*, *20*(17), 4788. https://doi.org/10.3390/s20174788

Benedek, M., & Kaernbach, C. (2010). A continuous measure of phasic electrodermal activity. *Journal of Neuroscience Methods*, *190*(1), 80–91. https://doi.org/10.1016/j.jneumeth.2010.04.028

Boucsein, W. (2012). Electrodermal activity. In *Springer eBooks*. https://doi.org/10.1007/978-1-4614-1126-0

Chen, N. W., Jaques, N., Taylor, S., Sano, A., Fedor, S., & Picard, R. W. (2015). Wavelet-based motion artifact removal for electrodermal activity. . https://doi.org/10.1109/embc.2015.7319814

Cimino, A., & Dell'Orletta, F. (2016). Tandem LSTM-SVM approach for Sentiment Analysis. In *Accademia University Press eBooks* (pp. 172–177). https://doi.org/10.4000/books.aaccademia.2003

Chen, F., & Chen, Z. D. (2019). Application of weighted combination model based on XGBoost and LSTM in sales forecasting. *Journal of Computer System Applications, 28*(10), 226–232.https://doi.org/10.15888/j.cnki.csa.007091

Chu, B., Marwaha, K., Sanvictores, T., & Ayers, D. (2019). Physiology, stress reaction. *StatPearls*. https://pubmed.ncbi.nlm.nih.gov/31082164/

Di Lascio, E., Gashi, S., & Santini, S. (2018). Unobtrusive Assessment of Students' Emotional Engagement during Lectures Using Electrodermal Activity Sensors. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, *2*(3), 1–21. https://doi.org/10.1145/3264913

Di Lascio, E., Gashi, S., & Santini, S. (2019). *Laughter Recognition Using Non invasive Wearable Devices*. https://doi.org/10.1145/3329189.3329216

Dzedzickis, A., Kaklauskas, A., & Bucinskas, V. (2020, January 21). *Human Emotion Recognition: Review of Sensors and Methods*. Sensors. https://doi.org/10.3390/s20030592

Fevre, M. L., Matheny, J., & Kolt, G. S. (2003). Eustress, distress, and interpretation in occupational stress. *Journal of Managerial Psychology*, *18*(7), 726–744. https://doi.org/10.1108/02683940310502412

Gashi, S., Di Lascio, E., Stancu, B., Swain, V. D., Mishra, V., Gjoreski, M., & Santini, S. (2020). *Detection of Artifacts in Ambulatory Electrodermal Activity Data. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 4(2), 1–31.* doi:10.1145/3397316

Gers, F. A., Schmidhuber, J., & Cummins, F. (2000). Learning to Forget: Continual Prediction with LSTM. *Neural Computation*, *12*(10), 2451–2471. https://doi.org/10.1162/089976600300015015

Gers, F. A., Schraudolph, N., & Schmidhuber, J. (2003). *Learning Precise Timing with LSTM Recurrent Networks*. https://www.semanticscholar.org/paper/Learning-Precise-Timing-with-LSTM-Recurrent-Gers-Schraudolph/047655e733a9eed9a500afd916efa566915b9110

Giuste, F., Shi, W., Zhu, Y., Naren, T., Isgut, M., Sha, Y., Tong, L., Gupte, M., & Wang, M. D. (2023). Explainable Artificial Intelligence Methods in Combating Pandemics: A Systematic review. *IEEE Reviews in Biomedical Engineering*, *16*, 5–21. https://doi.org/10.1109/rbme.2022.3185953

Gjoreski, M., Luštrek, M., Gams, M., & Gjoreski, H. (2017). Monitoring stress with a wrist device using context. *Journal of Biomedical Informatics*, *73*, 159–170. https://doi.org/10.1016/j.jbi.2017.08.006

Greco, A., Valenza, G., Lanata, A., Scilingo, E., & Citi, L. (2016). cvxEDA: a Convex Optimization Approach to Electrodermal Activity Processing. *IEEE Transactions on Bio-medical Engineering/IEEE Transactions on Biomedical Engineering*, 1. https://doi.org/10.1109/tbme.2015.2474131

Guo, X., Yin, Y., Dong, C., Yang, G., & Zhou, G. (2008). On the Class Imbalance Problem. *2008 Fourth International Conference on Natural Computation, Jinan, China*. https://doi.org/10.1109/icnc.2008.871

Hemakom, A., Atiwiwat, D., & Israsena, P. (2023). ECG and EEG based detection and

multilevel classification of stress using machine learning for specified genders: A preliminary study. *PloS One*, *18*(9), e0291070. https://doi.org/10.1371/journal.pone.0291070

Hernandez, J., Morris, R. R., & Picard, R. W. (2011). Call Center Stress Recognition with Person-Specific Models. In *Lecture notes in computer science* (pp. 125–134). https://doi.org/10.1007/978-3-642-24600-5_16

Heyat, M. B. B., Akhtar, F., Abbas, S. J., Al-Sarem, M., Alqarafi, A., Stalin, A., Abbasi, R., Muaad, A. Y., Lai, D., & Wu, K. (2022). Wearable flexible electronics based cardiac electrode for researcher mental stress detection system using machine learning models on single lead electrocardiogram signal. *Biosensors*, *12*(6), 427. https://doi.org/10.3390/bios12060427

Hernandez, J., Morris, R. R., & Picard, R. W. (2011). Call Center Stress Recognition with Person-Specific Models. In *Lecture notes in computer science* (pp. 125–134). https://doi.org/10.1007/978-3-642-24600-5_16

Hicks, S., Riegler, M., Pogorelov, K., Anonsen, K. V., De Lange, T., Johansen, D., Jeppsson, M., Randel, K. R., Eskeland, S. L., & Halvorsen, P. (2018). *Dissecting Deep Neural Networks for Better Medical Image Classification and Classification Understanding*. https://doi.org/10.1109/cbms.2018.00070

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Hossain, M. B., Posada-Quintero, H. F., Kong, Y., McNaboe, R., & Chon, K. H. (2022, April 1). *Automatic motion artifact detection in electrodermal activity data using machine learning*. Biomedical Signal Processing and Control. https://doi.org/10.1016/j.bspc.2022.103483

Hussein, R., Palangi, H., Ward, R. K., & Wang, Z. J. (2019). Optimized deep neural network architecture for robust detection of epileptic seizures using EEG signals. *Clinical Neurophysiology*, *130*(1), 25–37. https://doi.org/10.1016/j.clinph.2018.10.010

Jamshidzadeh, Z., Ehteram, M., & Shabanian, H. (2024). Bidirectional Long Short Term Memory (BILSTM) - Support Vector Machine: A new machine learning model for predicting water quality parameters. *Ain Shams Engineering Journal/Ain Shams Engineering Journal*, *15*(3), 102510. https://doi.org/10.1016/j.asej.2023.102510

Jaseena, K., & Kovoor, B. C. (2021). Decomposition-based hybrid wind speed forecasting model using deep bidirectional LSTM networks. *Energy Conversion and Management*, *234*, 113944. https://doi.org/10.1016/j.enconman.2021.113944

Jose, J. (2022). INTRODUCTION TO TIME SERIES ANALYSIS AND ITS APPLICATIONS.*ResearchGate*.https://www.researchgate.net/publication/362389180_INTRODUCTION_TO_TIME_SERIES_ANALYSIS_AND_ITS_APPLICATIONS

Joseph, R. V., Mohanty, A., Tyagi, S., Mishra, S., Satapathy, S. K., & Mohanty, S. N. (2022). A hybrid deep learning framework with CNN and Bi-directional LSTM for store item demand forecasting. *Computers & Electrical Engineering*, *103*, 108358. https://doi.org/10.1016/j.compeleceng.2022.108358

Juneau, P., Baddour, N., Burger, H., Bavec, A., & Lemaire, E. D. (2021). Comparison of Decision Tree and Long Short-Term memory approaches for automated foot strike detection in lower extremity amputee populations. *Sensors*, *21*(21), 6974. https://doi.org/10.3390/s21216974

Khorram, A., Khalooei, M., & Rezghi, M. (2020). End-to-end CNN + LSTM deep learning approach for bearing fault diagnosis. *Applied Intelligence*, *51*(2), 736–751. https://doi.org/10.1007/s10489-020-01859-1

Kleckner, I. R., Jones, R. M., Wilder-Smith, O., Wormwood, J. B., Akcakaya, M., Quigley, K. S., Lord, C., & Goodwin, M. S. (2018). Simple, transparent, and flexible automated quality assessment procedures for ambulatory electrodermal activity data. *IEEE Transactions on Bio-medical Engineering/IEEE Transactions on Biomedical Engineering*, *65*(7), 1460–1467. https://doi.org/10.1109/tbme.2017.2758643

Kocielnik, R., Sidorova, N., Maggi, F. M., Ouwerkerk, M., & Westerink, J. H. D. M. (2013). *Smart technologies for long-term stress monitoring at work. Proceedings of the 26th IEEE International Symposium on Computer-Based Medical Systems.* doi:10.1109/cbms.2013.6627764

Koklu, M., Unlersen, M. F., Ozkan, I. A., Aslan, M. F., & Sabanci, K. (2022). A CNN SVM study based on selected deep features for grapevine leaves classification. *Measurement*, *188*, 110425. https://doi.org/10.1016/j.measurement.2021.110425

Kortum, E., Leka, S., & Cox, T. (2010). Psychosocial risks and work-related stress in developing countries: health impact, priorities, barriers and solutions. *International Journal of Occupational Medicine and Environmental Health*, *23*(3). https://doi.org/10.2478/v10001-010-0024-5

Lang, P. J., Bradley, M. M., & Cuthbert, B. N. (2005). International Affective Picture System (IAPS) Affective Ratings of Pictures and Instruction Manual. Technical Report A-6, Gainesville, FL University of Florida. - References - Scientific Research Publishing. (n.d.). https://www.scirp.org/reference/referencespapers?referenceid=2323896

Lee, G., Choi, B., Jebelli, H., Ahn, C. R., & Lee, S. (2020). Noise reference Signal Based denoising method for EDA collected by multimodal biosensor wearable in the field. *Journal of Computing in Civil Engineering*, *34*(6). https://doi.org/10.1061/(asce)cp.1943-5487.0000927

Li, S., Zhao, X., Stankovic, L., & Mandic, D. (2022). Demystifying CNNs for imagesby matched filters. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2210.08521

Lin, X., Lei, Y., Chen, J., Xing, Z., Yang, T., Wang, Q., & Wang, C. (2023). A Case Finding Clinical Decision Support System to Identify Subjects with Chronic Obstructive Pulmonary Disease Based on Public Health Data. *Tsinghua Science and Technology/Tsinghua Science and Technology*, *28*(3), 525–540. https://doi.org/10.26599/tst.2022.9010010

Liu, Y., & Du, S. (2018). Psychological stress level detection based on electrodermal activity. *Behavioural Brain Research*, *341*, 50–53. https://doi.org/10.1016/j.bbr.2017.12.021

Llanes-Jurado, J., Carrasco-Ribelles, L. A., Alcañiz, M., & Marín-Morales, J. (2021). Automatic artifact recognition and correction for electrodermal activity in uncontrolled environments. *Research Square (Research Square)*. https://doi.org/10.21203/rs.3.rs-717360/v

Llanes-Jurado, J., Carrasco-Ribelles, L. A., Alcañiz, M., Soria-Olivas, E., & Marín Morales, J. (2023, November 1). *Automatic artifact recognition and correction for electrodermal activity based on LSTM-CNN models*. Expert Systems With Applications. https://doi.org/10.1016/j.eswa.2023.120581

Mordensky, S. P., Lipor, J. J., DeAngelo, J., Burns, E. R., & Lindsey, C. R. (2023). When less is more: How increasing the complexity of machine learning strategies for geothermal energy assessments may not lead toward better estimates. *Geothermics*, *110*, 102662. https://doi.org/10.1016/j.geothermics.2023.102662

Moruzzi, G., & Magoun, H. (1949). Brain stem reticular formation and activation of the EEG. *Electroencephalography and Clinical Neurophysiology*, *1*(1–4), 455–473. https://doi.org/10.1016/0013-4694(49)90219-9

Mohsen, S., Ghoneim, S. S. M., Alzaidi, M. S., Alzahrani, A., & Hassan, A. M. A. (2023). Classification of electroencephalogram signals using LSTM and SVM based on Fast Walsh-Hadamard Transform. *Computers, Materials & Continua/Computers, Materials & Continua (Print)*, *75*(3), 5271–5286. https://doi.org/10.32604/cmc.2023.038758

Mozer, M. C. (1997). Induction of multiscale temporal structure. *ResearchGate*. https://www.researchgate.net/publication/2926596_Induction_of_Multiscale_Temporal_Structure

Munir, M., Siddiqui, S. A., Küsters, F., Mercier, D., Dengel, A., & Ahmed, S. (2019). TSXPlain: Demystification of DNN decisions for Time-Series using natural language and statistical features. In *Lecture notes in computer science* (pp. 426–439). https://doi.org/10.1007/978-3-030-30493-5_43

Mordensky, S. P., Lipor, J. J., DeAngelo, J., Burns, E. R., & Lindsey, C. R. (2023). When less is more: How increasing the complexity of machine learning strategies for geothermal energy assessments may not lead toward better estimates. *Geothermics*, *110*, 102662. https://doi.org/10.1016/j.geothermics.2023.102662

Moruzzi, G., & Magoun, H. (1949). Brain stem reticular formation and activation of the EEG. *Electroencephalography and Clinical Neurophysiology*, *1*(1–4), 455–473. https://doi.org/10.1016/0013-4694(49)90219-9

Nagabushanam, P., George, S. T., & Radha, S. (2019). EEG signal classification using LSTM and improved neural network algorithms. *Soft Computing*, *24*(13), 9981–10003. https://doi.org/10.1007/s00500-019-04515-0

Orlanda-Ventayen, C., & Ventayen, R. J. (2022). Stress and depression in the workplace of educators in the Philippines. *International Journal of Occupational Safety and Health*, *12*(4), 325–335. https://doi.org/10.3126/ijosh.v12i4.43226

O'Reilly, R. C., & Frank, M. J. (2006). Making working Memory work: a computational model of learning in the prefrontal cortex and basal ganglia. *Neural Computation*, *18*(2), 283–328. https://doi.org/10.1162/089976606775093909

Park, Y., & Yang, H. S. (2019). Convolutional neural network based on an extreme learning machine for image classification. *Neurocomputing*, *339*, 66–76. https://doi.org/10.1016/j.neucom.2018.12.080

Pérez-Ortiz, J. A., Gers, F. A., Eck, D., & Schmidhuber, J. (2003). Kalman filters improve LSTM network performance in problems unsolvable by traditional recurrent nets. *Neural Networks*, *16*(2), 241–250. https://doi.org/10.1016/s0893-6080(02)00219-8

Posada-Quintero, H. F., & Chon, K. H. (2020). Innovations in Electrodermal Activity Data Collection and Signal Processing: A Systematic review. *Sensors*, *20*(2), 479. https://doi.org/10.3390/s20020479

Posner, J., Russell, J. A., & Peterson, B. S. (2005). The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology. *Development and Psychopathology*, *17*(03). https://doi.org/10.1017/s0954579405050340

Saeb, S., Lonini, L., Jayaraman, A., Mohr, D. C., & Kording, K. P. (2016). Voodoo machine learning for clinical predictions. *bioRxiv (Cold Spring Harbor Laboratory)*. https://doi.org/10.1101/059774

Sarker, H., Sharmin, M., Ali, A. A., Rahman, M. M., Bari, R., Hossain, S. M., & Kumar, S. (2014). *Assessing the availability of users to engage in just-in-time*

intervention in the natural environment. https://doi.org/10.1145/2632048.2636082

Sánchez-Reolid, R., De La Rosa, F. L., Sánchez-Reolid, D., López, M. T., & Fernández-Caballero, A. (2022). Machine Learning Techniques for Arousal Classification from Electrodermal Activity: A Systematic Review. *Sensors*, *22*(22), 8886. https://doi.org/10.3390/s22228886

Sánchez-Reolid, R., Martínez-Rodrigo, A., & Fernández-Caballero, A. (2019). Stress Identification from Electrodermal Activity by Support Vector Machines. In *Lecture notes in computer science* (pp. 202–211). https://doi.org/10.1007/978-3-030-19591-5_21

Sánchez-Reolid, R., Martínez-Rodrigo, A., López, M. T., & Fernández-Caballero, A. (2020). Deep Support Vector Machines for the Identification of Stress Condition from Electrodermal Activity. *International Journal of Neural Systems*, *30*(07), 2050031. https://doi.org/10.1142/s0129065720500318

Semmlow, J. L. (2004). *Biosignal and medical image processing*. CRC Press.

Sethi, Amit. (2006). INTERACTION BETWEEN MODULES IN LEARNING SYSTEMS FOR VISION APPLICATIONS. https://www.researchgate.net/publication/249862831_INTERACTION_BETWEEN_MODULES_IN_LEARNING_SYSTEMS_FOR_VISION_APPLICATIONS

Schmidt, P., Reiss, A., Duerichen, R., Marberger, C., & Van Laerhoven, K. (2018). Introducing WESAD, a Multimodal Dataset for Wearable Stress and Affect Detection. *International Conference on Multimodal Interaction*. https://doi.org/10.1145/3242969.3242985

Schmidt, P., Dürichen, R., Reiss, A., Van Laerhoven, K., & Plötz, T. (2019). Multi target affect detection in the wild:An Exploratory Study. *Conference: The 23rd International Symposium*. https://doi.org/10.1145/3341163.3347741 *Stress*. (2022, June 17). https://www.who.int/news-room/questions-and-answers/item/stress#:~:text=Stress%20can%20be%20defined%20as,experiences%20stress%20to%20some%20degree

Stress. (2022, June 17). https://www.who.int/news-room/questions-and-answers/item/stress

Sun, L., Jin, B., Yang, H., Tong, J., Liu, C., & Xiong, H. (2018). Unsupervised EEG feature extraction based on echo state network. *Information Sciences*, *475*, 1–17. https://doi.org/10.1016/j.ins.2018.09.057

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). *Rethinking the Inception Architecture for Computer Vision*. https://doi.org/10.1109/cvpr.2016.308

Tang, Y. (2013). Deep Learning using Linear Support Vector Machines. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.1306.0239

Taylor, S., Jaques, N., Chen, W., Fedor, S., Sano, A., Picard, R. (2015). Automatic identification of artifacts in electrodermal activity data. Annu Int Conf IEEE Eng Med Biol Soc. 2015;2015:1934-7. doi: 10.1109/EMBC.2015.7318762. PMID: 26736662; PMCID: PMC5413200.

Thammasan, N., Stuldreher, I. V., Schreuders, E., Giletta, M., & Brouwer, A. (2020). A usability study of physiological measurement in school using wearable sensors. *Sensors*, *20*(18), 5380. https://doi.org/10.3390/s20185380

Vapnik, V. N. (1995). The nature of statistical learning theory. In *Springer eBooks*. https://doi.org/10.1007/978-1-4757-2440-0

Wan, H., Xu, R., Zhang, M., Cai, Y., Li, J., & Shen, X. (2022). A novel model for water

quality prediction caused by non-point sources pollution based on deep learning and feature extraction methods. *Journal of Hydrology*, *612*, 128081. https://doi.org/10.1016/j.jhydrol.2022.128081

Wang, B., Ma, R., Kuang, J., & Zhang, Y. (2020). How decisions are made in brains: Unpack “Black Box” of CNN with Ms. Pac-Man video game. *IEEE Access*, *8*, 142446–142458. https://doi.org/10.1109/access.2020.3013645

Wang, Q.S.. (2019). Research and application of timing optimization of short-term traffic flow at urban intersections. University of Science and Technology of China, Hefei, China.

Weerakody, P. B., Wong, K. W., Wang, G., & Ela, W. (2021). A review of irregular time series data handling with gated recurrent neural networks. *Neurocomputing*, *441*, 161–178. https://doi.org/10.1016/j.neucom.2021.02.046

What is Python? Executive Summary. (n.d.). Python.org. https://www.python.org/doc/essays/blurb/

What is sklearn? | Domino Data Lab. (n.d.). Domino Data Lab. https://domino.ai/data-science-dictionary/sklearn

What is TensorFlow? (n.d.). NVIDIA Data Science Glossary. https://www.nvidia.com/en-us/glossary/tensorflow/

Wu, Z., Rincon, D., & Christofides, P. D. (2020). Process structure-based recurrent neural network modeling for model predictive control of nonlinear processes. *Journal of Process Control*, *89*, 74–84. https://doi.org/10.1016/j.jprocont.2020.03.013

Yanase, J., & Triantaphyllou, E. (2019). The seven key challenges for the future of computer-aided diagnosis in medicine. *International Journal of Medical Informatics*, *129*, 413–422. https://doi.org/10.1016/j.ijmedinf.2019.06.017

Zhang, Q., Wang, R., Qi, Y., & Wen, F. (2022). A watershed water quality prediction model based on attention mechanism and Bi-LSTM. *Environmental Science and Pollution Research International*, *29*(50), 75664–75680. https://doi.org/10.1007/s11356-022-21115-y

Zhang, Y., Haghdan, M., & Xu, K. S. (2017). Unsupervised motion artifact detection in Wrist-Measured Electrodermal Activity data. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.1707.08287

APPENDICES

Appendix 1. Research Instruments

CONFUSION MATRIX

	Actual Values	Predicted Values	
		Non-Artifact (0)	**Artifact (1)**
Subjects	**Non-Artifact (0)**	True Negative (TN)	False Positive (FP)
	Artifact (1)	False Negative (FN)	True Positive (TP)

CONFUSION MATRIX PER TEST SUBJECT

Subject	**Actual**	**Predicted**	
		Non-Artifact (0)	*Artifact (1)*
otafeh_expert1	*Non-Artifact (0)*		
	Artifact (1)		
ajeric_expert2	*Non-Artifact (0)*		
	Artifact (1)		
pqbqpr_expert2	*Non-Artifact (0)*		
	Artifact (1)		
afegip_expert1	*Non-Artifact (0)*		
	Artifact (1)		
uhepah_expert1	*Non-Artifact (0)*		
	Artifact (1)		
inefoh_expert1	*Non-Artifact (0)*		
	Artifact (1)		
oxused_expert2	*Non-Artifact (0)*		
	Artifact (1)		
ukudab_expert2	*Non-Artifact (0)*		
	Artifact (1)		
iguted_expert1	*Non-Artifact (0)*		
	Artifact (1)		
ekamis_expert2	*Non-Artifact (0)*		
	Artifact (1)		

AVERAGE PERFORMANCE OF TEST SUBJECTS PER METRIC

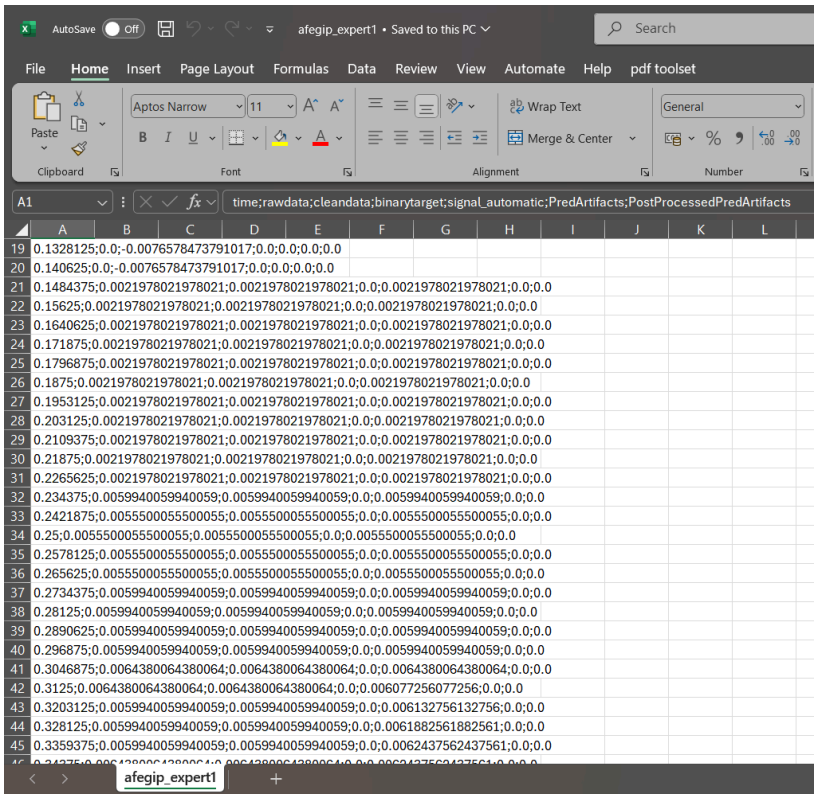
Subject	Accuracy	F1-Score	ROC-AUC	Precision	Recall/TPR
oxused_expert2					
ukudab_expert2					
otafeh_expert1					
uhepah_expert1					
ajeric_expert2					
afegip_expert1					
ekamis_expert2					
pqbqpr_expert2					
iguted_expert1					
inefoh_expert1					

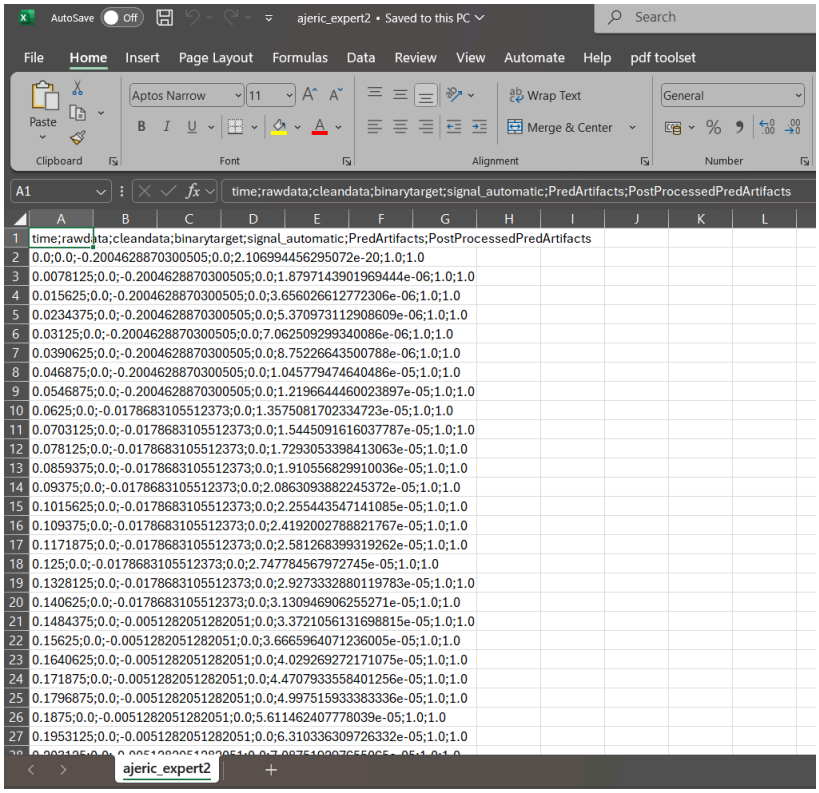
PERFORMANCE OF ARTIFACT DETECTION MODELS

Artifact Detection Models	ROC-AUC	Recall/TPR	Precision	F1-Score	Accuracy
Proposed LSTM-SVM					
Logistic Regression (Taylor et. al., 2015)					
Random Forest (Taylor et. al., 2015)					
Support Vector Machine (Taylor et. al., 2015)					
Logistic Regression (Hossain et. al., 2022)					
Gradient Boost (Hossain et. al., 2022)					
Support Vector Machine (Hossain et. al., 2022)					
LSTM-CNN (Llanes-Jurado et. al., 2023)					

Appendix 2. Raw Data

Snippets of Raw Data from Electrodermal Activity Artifact Correction Benchmark Dataset (EDABE)





Appendix 3. Correspondence

21 August 2024

PETER ANTONIO B. BANZON
Chief Science Research Specialist
COARE Facility, DOST-ASTI, C.P. Garcia Ave.,
U.P. Technopark, Diliman, Quezon City

Dear Mr. Banzon:

This is to endorse **Johana Benolirao, Christaline Calubayan, Larry Miguel Cueva, and Deseree Quiray, students** of **Polytechnic University of the Philippines** to have access to the DOST-ASTI COARE Facility and avail the **HPC** service.

Johana Benolirao, Christaline Calubayan, Larry Miguel Cueva, and Deseree Quiray are conducting the **thesis** titled **Application of Hybridized LSTM-SVM in the Detection of Artifacts in Electrodermal Activity Signals for Stress Detection**.

This thesis aims to develop a hybrid LSTM-SVM model for recognizing artifacts in Electrodermal Activity (EDA) signals, which are used to detect stress. EDA, a psychophysiological signal, is easier to use for long-term emotional state monitoring compared to EEG and ECG, thanks to its simplicity and lower equipment cost. However, like EEG and ECG, EDA signals are prone to artifacts caused by factors such as unstable electrode contact, temperature changes, and movement.

Previous research by Llanes-Jurado et al. (2023) used a hybrid LSTM-CNN model to automatically detect these artifacts, achieving 88% accuracy, 57% F1-score, and 76% Area Under Curve (AUC). While effective, CNNs can be difficult to interpret. To address this, our study proposes a hybrid LSTM-SVM model that combines the strengths of LSTM with the interpretability of SVM.

We will compare our model's performance to SVM, Gradient Boosting, Random Forest, and LSTM-CNN in terms of accuracy, precision, recall, F1-score, and AUC using the Electrodermal Activity Artifact Correction Benchmark dataset. However, training our model requires significant computational resources due to the size of the dataset and the complexity of the procedures involved. Our current resources are insufficient, leading to excessive time and memory consumption. To address these limitations, we seek access to your HPC services, which would allow us to efficiently train our model, achieve more accurate results, and conduct a more thorough analysis in our study. Access to these resources is critical for optimizing our model's performance and ensuring the reliability of our findings.

You can contact me through phone at **5335-1777 loc 605** and email at mjmtan@pup.edu.ph for any questions on this matter.

Sincerely yours,

Supervisors:

PROF. MARY JANE TAN
Research Adviser

Date: August 20, 2024

PROF. MELVIN ROXAS
Dean, College of Computer and Information Sciences

Date:August 20, 2024

Proponents/Endorsees:

JOHANA BENOLIRAO
Student

Date: August 19, 2024

CHRISTALINE CALUBAYAN
Student

Date: August 19, 2024

LARRY MIGUEL CUEVA
Student

Date: August 19, 2024

DESEREE QUIRAY
Student

Date: August 19, 2024

Appendix 4. Ethical Clearance

Republic of the Philippines
POLYTECHNIC UNIVERSITY OF THE PHILIPPINES
OFFICE *of the* VICE PRESIDENT *for* RESEARCH, EXTENSION, *and* DEVELOPMENT
RESEARCH MANAGEMENT OFFICE
UNIVERSITY RESEARCH ETHICS CENTER

Date: November 13, 2024

To/For: **JOHANA BENOLIRAO**
CHRISTALINE CALUBAYAN
LARRY MIGUEL CUEVA
DESEREE QUIRAY

Subject: Ethical Clearance

From: **Julie Charmain O. Bonifacio**
Chief, Research Ethics Section

This is to inform you that the documentary requirements you submitted for your research project titled **"APPLICATION OF A HYBRIDIZED LSTM-SVM IN THE DETECTION OF ARTIFACTS IN ELECTRODERMAL ACTIVITY SIGNALS FOR STRESS DETECTION"** passed the evaluation of the PUP Research Ethics Committee (REC) in accordance with the requirements set by the Philippine Health Research Ethics Board (PHREB).

UREC Code	UREC-2024 -1744
Type of Review	EXPEDITED
Approval Date	November 13, 2024
Expiry Date	November 12, 2025
PUP-UREC Decision	**Approved**

The standard conditions of this approval are as follows:

1. Conduct the project following the submitted and approved research protocol and other documentary requirements.
2. If changes are to be made in the research project/study that will affect the research participants, an amendment of the research protocol must be submitted to urec@pup.edu.ph before implementing such changes.
3. When ethical clearance is about to expire, the researcher must apply to resubmit the research protocol.
4. A final report/terminal report must be submitted when the research project/study is complete.
5. Researchers must advise the PUP-UREC (email: urec@pup.edu.ph) in writing if the research project/study has been discontinued.

You may now commence your research project/study. Good luck.

S423, 4th Floor South Wing, PUP A. Mabini Campus, Anonas Street, Sta. Mesa, Manila 1016
Trunk Line: 335-1787 or 335-1777 local 235/357
Website: www.pup.edu.ph | Email: vpredl@pup.edu.ph

THE COUNTRY'S 1st POLYTECHNICU

SOCOTEC ISO 9001

ISO 9001:2015 CERTIFIED
CERTIFICATE NUMBER: SCP000413Q

Appendix 5. Certification of Originality Check

turnitin Page 2 of 165 - Integrity Overview Submission ID trn:oid:::31142:83288499

19% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Match Groups

- 535 Not Cited or Quoted 14%
  Matches with neither in-text citation nor quotation marks
- 152 Missing Quotations 4%
  Matches that are still very similar to source material
- 0 Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- 3 Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

12% Internet sources
14% Publications
9% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Appendix 6. Certification of Editing

CERTIFICATION OF EDITING

This is to certify that the thesis work entitled **Application of a Hybridized LSTM-SVM in the Detection of Artifacts in Electrodermal Activity Signals for Stress Detection** by **Benolirao, Johana G., Calubayan Christaline B., Cueva, Larry Miguel R., and Quiray, Deseree O.,** was proofread and edited by the undersigned.

This certification is being issued for whatever legal purpose it may serve.

Signed: MARVIN DOMINIC B. BUENA, LPT, MA
(Signature over Printed Name)

Affiliation: Southeast Asian Minister of Education Organization – Regional Centre for Innovation and Technology (SEAMEO INNOTECH)

Contact Number: 09223549524

Email Address: marvnbuena@gmail.com

Date: February 25, 2025

PUP Data Privacy Act

Thank you for providing your data. At Polytechnic University of the Philippines (PUP) we respect and value your rights as a data subject under the Data Privacy Act (DPA). PUP is committed to protecting the personal data you provide in accordance with the requirements under the DPA and IRR. In this regard, PUP implements reasonable and appropriate security measures to maintain the confidentiality, integrity and availability of your personal data. For more detailed Privacy Notice you may visit the following link: https://www.pup.edu.ph/privacy.

Appendix 7. Biographical Statements

Johana Benolirao is a fourth-year undergraduate student at the Polytechnic University of the Philippines, pursuing a Bachelor of Science in Computer Science. With a strong technical foundation, she possesses intermediate skills in web development and is proficient in programming languages such as Java, C, and Python. Additionally, she is well-versed in workspace applications and design tools, including Canva and Figma.

Christaline Calubayan is a fourth-year undergraduate student at the Polytechnic University of the Philippines, pursuing a Bachelor of Science in Computer Science. Her skill set includes programming in Python, Java, and C, along with skill in front-end web development using HTML, TailwindCSS, and Javascript. She has experience in graphic design and a background in database management.

Larry Miguel Cueva is a fourth-year undergraduate student at the Polytechnic University of the Philippines, pursuing a Bachelor of Science in Computer Science. His expertise spans multiple programming languages and frameworks, including Python, JavaScript, Flask, React, and Svelte. He is also proficient in machine learning libraries such as TensorFlow, Keras, and Scikit-Learn. With a strong passion for artificial intelligence, his interests lie in machine learning and deep learning, data loading and preprocessing, as well as data analysis and visualization.

Deseree Quiray is a fourth-year Computer Science undergraduate at the Polytechnic University of the Philippines. She is skilled in database management using SQL and web application development with HTML, CSS, and JavaScript. Additionally, she is proficient in Microsoft Office applications and design tools like Figma and Canva. She also has experience working with programming languages such as C, Java, and Python.

Appendix 8. Implementation Report

Introduction

The study, Application of Hybridized LSTM-SVM in the Detection of Artifacts in Electrodermal Activity Signals for Stress Detection, aims to identify and correct artifacts or noise present in EDA signals using a hybrid approach that combines machine learning and deep learning techniques. By leveraging the strengths of both methods, the study seeks to develop a more efficient and accurate model for artifact detection. This advancement could significantly enhance stress detection by ensuring cleaner and more reliable EDA signals. Additionally, improving artifact removal methods may contribute to the broader application of EDA in real-time stress monitoring, clinical diagnostics, and wearable health technologies.

Problem Statement

The challenge in artifact detection for EDA signals lies in the limitations of traditional machine learning and deep learning approaches, which rely on either lower-order handcrafted features or higher-order learned features. To address this, the study proposes a hybrid LSTM-SVM model that integrates both feature types, leveraging LSTM's ability to extract complex patterns and SVM's efficiency in classification. This approach aims to improve artifact detection accuracy, leading to cleaner EDA signals for more reliable stress detection. The study answers the following research questions, which are:

- What are the optimal Gamma and C hyperparameters for the SVM mechanism of the model such that it achieves higher evaluation metrics than using other Gamma and C values, to be used as the baseline of comparison to other state-of-the-art automatic artifact detection methods such as SVM, Logistic Regression, Random Forest, Gradient Boosted Tree, and LSTM-CNN?

- What would be the performance of the hybridized LSTM-SVM model compared to other existing models (SVM, Logistic Regression, Random Forest, GradBoost, LSTM-CNN) that automatically detect artifacts in EDA signals in terms of its:
    f. ROC-AUC
    g. F1-Score
    h. Precision
    i. Recall
    j. Accuracy

Description of Data

This study utilized the Electrodermal Activity Artifact Correction Benchmark (EDABE) dataset, which contains 74.46 hours of EDA recordings from 43 subjects affected by hand and body motion artifacts. The dataset is divided into a training set (33 subjects, 56.27 hours) and a test set (10 subjects, 18.19 hours), recorded using a Shimmer3 GSR+ Unit at 128 Hz.

The data were collected by Llanes-Jurado et al. (2023) during a virtual reality (VR) study designed to induce stress in participants through work-related scenarios. Participants engaged in office conversations, decision-making meetings, and three interactive minigames (firefighting, pipe reconstruction, and a maze with math tasks), all requiring hand movements, which introduced artifacts in the EDA signals. The VR environment was created in Unity3D and experienced using an HTC Vive Pro Eye headset. The experiment lasted ~104 minutes per participant, resulting in a high-quality dataset for artifact detection research.

Time Frame

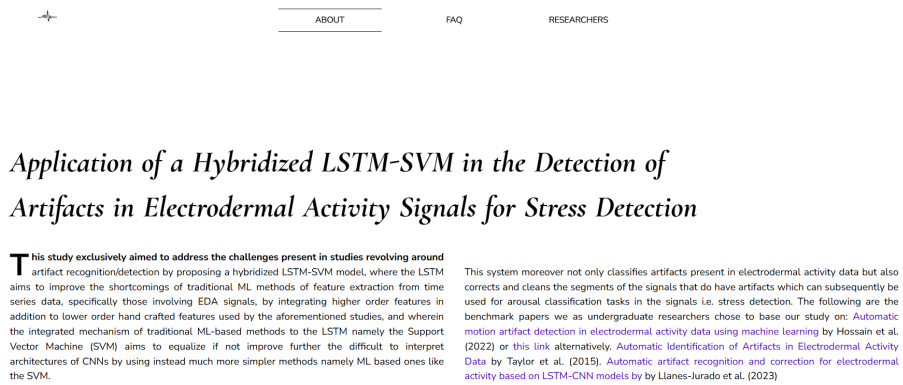
Activities	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan 2025	Feb 2025
Proposal Defense									
Proposal Revisions									
Tool Development									
Tool Defense									
Tool Revisions									
Implementation of the Tool									
Final Defense									
Redefense									
Final Revision									
Final Manuscript									

Implementation Procedure

The research examines an LSTM-SVM model's performance against established baseline methods presented in previous studies. The research utilizes three baseline models from the three benchmark authors using the EDABE dataset that was segmented into 0.5-second intervals for processing signals. Recursion-based Random Forest classifier performs feature elimination to eliminate redundancy from statistical, time-frequency and wavelet features. Each baseline model makes use of different classifiers including Logistic Regression and SVM together with Random Forest and Gradient Boosted Trees. The model parameters were optimized by running grid search combined with Leave-One-Subject-Out (LOSO) cross-validation approaches. The study[5] introduces their deep learning-based LSTM-CNN model as part of multiple baseline models. The proposed LSTM-SVM model combines LSTM-produced high-level features

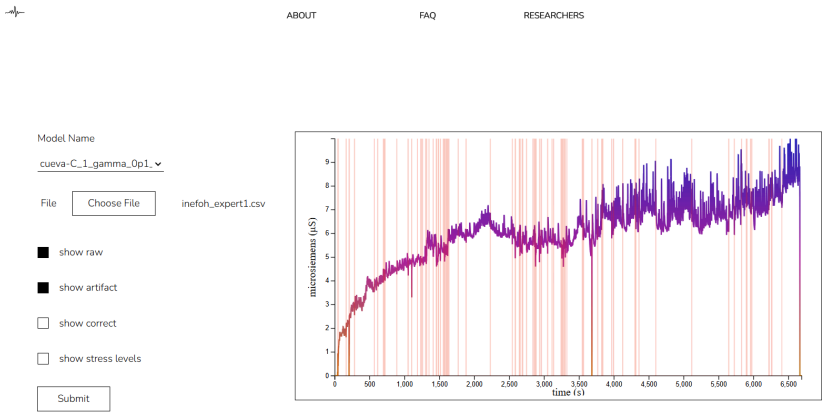
with traditionally derived low-level features which were originally presented in earlier work. The proposed model implements an SVM classifier as it uses hyperparameter optimization to achieve optimal performance. This research uses LOSO cross-validation with 43 subjects to evaluate the data without any shared information between training and testing phases for enhanced generalization capabilities. The model selected from cross-validation testing possesses superior results that are checked against baseline approaches for the final assessment.

Appendix 9. Screen Layout and Description



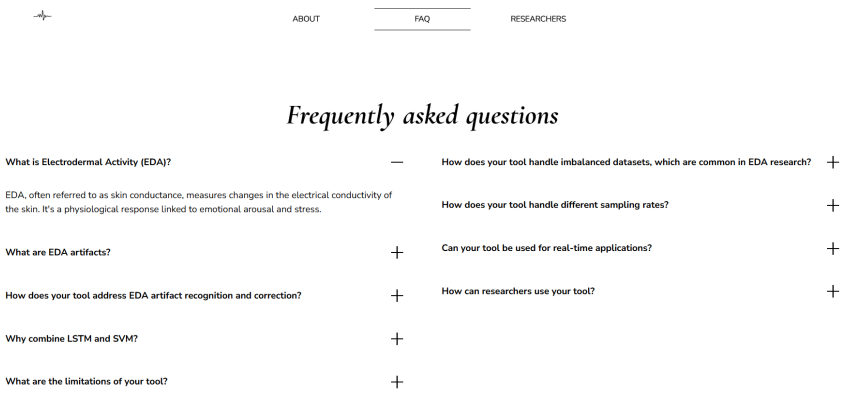
About Page

This page provides a concise overview of both the research study and the system developed for it. It includes the thesis title, a general summary of the study, and an explanation of the system's functionality.



Artifact Detector Page

This section outlines the overall functionality of the tool. Upon selecting a file, it displays the electrodermal activity signals along with detected artifacts. The system can show the raw signal, the artifacts, the corrected signals, and the corresponding stress levels.



Frequently Asked Questions (FAQs) Page

This page presents frequently asked questions related to electrodermal activity signals, EDA artifacts, LSTM-SVM, and other topics concerning our system and study. It offers answers that may provide a different perspective on understanding the research.

Researchers Page

This section introduces the researchers behind the study, featuring their names and images. It highlights the individuals responsible for making the research possible.

Appendix 10. Thesis Revision Matrices

Polytechnic University of the Philippines
COLLEGE OF COMPUTER AND INFORMATION SCIENCES
Sta. Mesa, Manila

THESIS PROPOSAL DEFENSE REVISION MATRIX

PROPOSAL TITLE	SECTION	GROUP NO.
APPLICATION OF A HYBRIDIZED LSTM-SVM IN THE DETECTION OF ARTIFACTS IN ELECTRODERMAL ACTIVITY SIGNALS FOR STRESS DETECTION	3-1	7

Document Part	COMMENTS/SUGGESTIONS/RECOMMENDATIONS by the Panel	ACTION TAKEN by the Proponent/s	Page No.	REMARKS (Panelist Approval by indicating the date) Panel 1	Panel 2	Panel 3	Panel 4
Title	• There is problems, Since you have a supervised algorithm there is problem. You're going to detect stress, what would be the output? Positive stress or negative stress? How are you going to detect stress? positive or negative? may levelling or wala? How are going to determine stress? what is purpose of the signal?	• We included review related studies for stress classification and how we will classify stress.	• Page 29				
Preliminaries							
Chapter 1	• In SOP specify what other models need to compare • In Hypothesis - There is no Significant • Conceptual Framework - define what is your output • Theoretical framework - should be related to your title, you just define your theoretical framework, there is problem. Change your theoretical framework based on the study you are referring.	• We specified the other models we will compare (SVM, GradBoost, LSTM-CNN). • The hypothesis is changed into one the null hypothesis only; we also included phrase "no significant difference". • Conceptual framework is simplified and the output was stated. • The theoretical framework is updated into the existing framework from the foundational studies referenced from.	• Page 9 • Page 14 • Page 13 • Page 11				
Chapter 2	• There are no related studies about stress detection, there is no related studies related to your title. What is your purpose? Why are you relating signals to stress detection? Why do you have noise, what does it represent? Relate your artifact to stress detection if you know the output, then there is no problem, and everything could follow.	• We included related studies about stress detection in the review of related literature and studies. We also added RRL and figures about the artifacts to better	• Page 19 • Page 29				
apter 3							
erences							
endices							
eral Observations	• What are the labels? What features to say that the signals are artifacts or non-artifacts	• We defined the labels, which are artifacts and non-artifacts included review of related studies	• Page 7 • Page 29				

(Please use additional sheet if necessary)

Polytechnic University of the Philippines
COLLEGE OF COMPUTER AND INFORMATION SCIENCES
Sta. Mesa, Manila

Office N206 2nd Floor North Wing, PUP A. Mabini Campus, Anonas Street, Sta. Mesa, Manila 1016
Line: 335-1787 or 335-1777 local 272
Website: www.pup.edu.ph | Email: ccis@pup.edu.ph
COUNTRY'S 1st POLYTECHNICU

Polytechnic University of the Philippines
COLLEGE OF COMPUTER AND INFORMATION SCIENCES
Sta. Mesa, Manila

BSCS THESIS TOOL REVISION MATRIX

PROPOSAL TITLE	SECTION	GROUP NO.
APPLICATION OF A HYBRIDIZED LSTM-SVM IN THE DETECTION OF ARTIFACTS IN ELECTRODERMAL ACTIVITY SIGNALS FOR STRESS DETECTION	BSCS 4-1	7

Proposal/Tool Part	COMMENTS/SUGGESTIONS/RECOMMENDATIONS by the Panel	ACTION TAKEN by the Proponent/s	Page No.	REMARKS (Panelist Approval by indicating the date)			
				Panel 1	Panel 2	Panel 3	Panel 4
Title*							
Preliminaries							
Chapter 1*							
Chapter 2*							
Chapter 3*							
Tool (Software/Hardware)	**Panelist No.1: Aleta Fabregas** • Show the sample of artifacts • What are "artifacts"? • Ano yung purpose • Paano kayo naglinis? • There is already a model? What is the use of this? • Dapat nakikita niyo ano ang problema sa model niyo • How can you show that you solve the gaps	• Sample of artifacts and definition, criteria and guidelines for identification were included. • The purpose of this model is to improve the performance of artifact detection in the context of high-level and lower-order features, wherein the LSTM-SVM model is expected to					

Polytechnic University of the Philippines
COLLEGE OF COMPUTER AND INFORMATION SCIENCES
Sta. Mesa, Manila

	• Even though it is intensive, you still need to show • Wala nang papalitan na dataset • How to solve SOP2? • Why is there RF in SOP2 but none in SOP1? **Panelist No.2: Montaigne Molejon** • Data is already controlled, what is the purpose of your model? • What are the features? • What are the issues of the previous models? • What does your model do, and what problem is it pointing to? **Panelist No.3: Ma. Esperanza Reyes**	perform better than existing models. • The absence of Random Forest in SOP1 is due to a typographical error. It is included to the approaches compared to the proposed model.					
References							
General Observations							

(Please use additional sheet if necessary)

Name of Panelist 1: Aleta Fabregas 11/4/24 Fabregas

Name of Panelist 2: Montaigne Molejon 11/4/2024

Name of Panelist 3: Maria Esperanza Reyes 11.04.24

Name of Panelist 4:

CCIS Office N206 2nd Floor North Wing, PUP A. Mabini Campus, Anonas Street, Sta. Mesa, Manila 1016
Trunk Line: 335-1787 or 335-1777 local 272
Website: www.pup.edu.ph | Email: ccis@pup.edu.ph
THE COUNTRY'S 1st POLYTECHNICU

Polytechnic University of the Philippines
COLLEGE OF COMPUTER AND INFORMATION SCIENCES
Sta. Mesa, Manila

BSCS THESIS FINAL DEFENSE REVISION MATRIX

PROPOSAL TITLE	SECTION	GROUP NO.
APPLICATION OF A HYBRIDIZED LSTM-SVM IN THE DETECTION OF ARTIFACTS IN ELECTRODERMAL ACTIVITY SIGNALS FOR STRESS DETECTION	4-1	7

Proposal/Tool Part	COMMENTS/SUGGESTIONS/RECOMMENDATIONS by the Panel	ACTION TAKEN by the Proponent/s	Page No.	REMARKS (Panelist Approval by indicating the date) Panel 1	Panel 2	Panel 3	Panel 4
Chapter 4	**Panel 1: Aleta C. Fabregas** - Why is the value in hyperparameter high, when ROC-AUC is low? - Why did the TPR and TNR produced low results - TPR and TNR's result are misleading **Panel 2: Montaigne G. Molejon** - Show the gaps by the show raw plus show stress levels, and the show correct plus show stress levels **Panel 3: Maria Esperanza H. Reyes** - If the result has four decimal places, it should be uniform to all results	- The ROC-AUC is a metric that helps to determine the performance of the models, it is also used to determine the performance after hyperparameter tuning - The TPR and TNR results are removed as its results are misleading and it is not included in the Statement of the Problem - Uniformity was ensured in the results in Chapter 4	P. 99-112 P.99 - 214 P.99 - 214				
Chapter 5							
Supporting Documents							
References							

Polytechnic University of the Philippines
COLLEGE OF COMPUTER AND INFORMATION SCIENCES
Sta. Mesa, Manila

General Observations	**Panel 1: Aleta C. Fabregas** - 'Submit' button is misleading						

(Please use additional sheet if necessary)

Name of Panelist 1: **Aleta C. Fabregas** 01/29/2025 **Name of Panelist 2:** **Montaigne G. Molejon**

Name of Panelist 3: **Maria Esperanza H. Reyes** 01-24-2025 **Name of Panelist 4:**

*** Changes to the approved and revised proposal (Chap. 1-3) should NOT be accommodated in order to satisfy the inadequacies of Chapters 4 and 5.**

CCIS Office N206 2nd Floor North Wing, PUP A. Mabini Campus, Anonas Street, Sta. Mesa, Manila 1016
Trunk Line: 335-1787 or 335-1777 local 272
Website: www.pup.edu.ph | Email: ccis@pup.edu.ph
THE COUNTRY'S 1st POLYTECHNICU