2025 شباط العدد 16
No.16 Feb 2025
المجلة العراقية للبحوث الإنسانية والإجتماعية والعلمية
Iraqi Journal of Humanitarian, Social and Scientific Research
Print ISSN 2710-0952 Electronic ISSN 2790-1254

نماذج التعلم العميق والتعلم الآلي الهجينة لتصنيف المشاعر

غفران حامد زغير[1]
ضياء شهيد العزاوي[2]
نسيب عبدالله[3]
[1, 2]قسم علوم البرمجيات، كلية علوم الحاسوب وتكنولوجيا المعلومات،
جامعة واسط، الكوت، 52001، العراق
[3]LaTIM، INSERM UMR 1101، جامعة بريست، فرنسا

خلاصة:
تستكشف هذه الورقة تكامل الشبكة العصبية التلافيفية **(CNN)** مع آلة ناقل الدعم **(SVM)** والشبكة العصبية المتكررة **(RNN)** مع **SVM** للتصنيف الإدراكي. حقق النموذج الهجين**CNN- SVM** دقة مذهلة بنسبة 98٪ في مهام تصنيف الصور، والتي جمعت بين التعلم العميق الوظيفي وطرق التعلم الآلي التقليدية باعتبارها فعالة بشكل واضح، حققت خوارزمية **RNN-SVM** الهجينة دقة قدرها 87٪، مما أظهر القدرة على التقاط السلاسل الزمنية لاستشعار ما تم العثور عليه. تسلط هذه الدراسة الضوء على قدرة التعلم العميق الهجين و إطار التعلم الآلي لزيادة دقة التصنيف الحسي.
الكلمات المفتاحية: واجهات الدماغ والحاسوب، تخطيط كهربية الدماغ، آلات المتجهات الداعمة، الشبكات العصبية التلافيفية، الشبكات العصبية المتكررة

Hybrid Deep Learning and Machine Learning Models for Emotion Classification

Ghufran Hamid Zghair[1]
Dheyaa Shaheed Al-Azzawi[2]
Nassib ABDALLAH[3]
[1,2] Department of Software Science, College of Computer Science and Information Technology,
Wasit University, Al-Kut, 52001, IRAQ
[3]LaTIM, INSERM UMR 1101, University of Brest, FRANCE
Corresponding author: ghufraan.hamed@uowasit.edu.iq

ABSTRACT
This paper explores the integration of Convolutional Neural Network (CNN) with Support Vector Machine (SVM) and Recurrent Neural Network (RNN) with SVM for perceptual classification The CNN-SVM hybrid model achieved an impressive 98% accuracy in image classification tasks, which combined functional deep learning and traditional machine learning methods As evidently efficient, the RNN-SVM hybrid algorithm achieved an accuracy of 87%, which demonstrated the capability of time series capture for sensing a found in. This study highlights the ability of the hybrid deep learning and machine learning framework to increase sensory classification accuracy
Keywords: Brain-Computer Interfaces, Electroencephalography, Support Vector Machines, Convolutional *Neural Networks, Recurrent Neural Networks*

1. INTRODUCTION

The Communication Brain Computer Interface (BCI) tries to bridge the gap between the human brain and a computer or other machine. This system tracks brain activity

changes and can be shown to a computer screen or applied to external devices eg stock of emotions. BCI has attracted widespread research interest, especially for neurorehabilitation, and the patients suffering from respective diseases or after-effects [1]. ConclusionBCI technology, has been extensively investigated in recent years, may indicate hope for communicating by thoughts only [1], [2]. Neurons in Brain work through electric signals with each other. One way scientists study this electrical activity is by measuring the scalp potential changes driven by brain activity. You end up with a signal called the electroencephalogram (EEG), which indicates the variations in the potential between two points, and is used to demonstrate the potential variations between two points is useful in assessing the brain waves In 1929, Hans Berger was able to record the first EEG of a human being and he named it electroencephalogram. Extending Richard Catton's early seminal work on examining animal brain function in the nineteenth century [1], [2], [3].
Emotions are an important component of the human mind that is affected by factors in the environment and has an impact on decision making and interpersonal communication [4], [5], [6], [7] Emotions are produced in response to certain ideas, experiences or happening , it affects mental and physical health.
Positive emotions can enhance health, while negative emotions can decrease quality of life [3], [8].

2. PROPOSAL MODEL SCHEMA

The proposal model outline consists of three stages. The first step is to collect and preprocess the EEG data for discrete training and testing. In the second stage, hybrid algorithms are employed, with the first group comprising CNN and SVM, and the second group comprising RNN and SVM. These algorithms are designed to leverage the strengths of both deep learning (CNN and RNN) and traditional machine learning (SVM) for emotion classification. Finally, in the third stage, emotions are evaluated and classified into three categories: sad, normal, and happy. This process aims to develop a robust system for accurately classifying emotions based on EEG data, as depicted in Fig. 1.

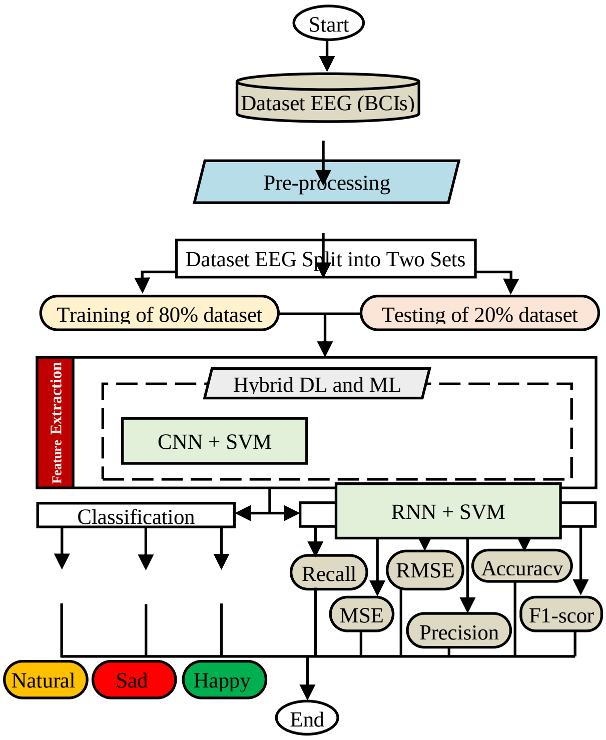


FIGURE 1. - The Proposal Model Schema

2.1. DATASET EEG (BCIS) AND PRE-PROCESSING

The electroencephalogram (EEG) dataset involves brain activity and is usually recorded using electrodes that are probably placed on the scalp of patients. The attributes for the models were extracted from an open source dataset gotten from the Kaggle website. It was reprocessed through the following stages: Data Preprocessing was followed by Data Loading, Data Cleaning, Missing Values, and Transform Data and divided the data set in two parts for training and testing. The numbers of the training set is 1705 and include the samples used for the training of CNN-SVM and RNN-SVM in order to classify the emotion according to EEG signals [9], [10], [11]. These samples are helpful for the model to find out the patterns and relationships among the data and be able to provide better predictions when the testing phase starts On the other hand, there is the testing set with the total of 427 samples that are excluded from the training process. These samples are not included in the training set and the overall performance and working ability of the model is tested on these. By determining the degree of accuracy that the model achieves for the new data that are not included in the modeling data set, the researchers are able to determine whether the emotions have been classified well in real life scenarios.

2.2. HYBRID DL WITH ML ALGORITHMS

It is with the aid of Deep Learning (DL) and Machine Learning (ML) that the authors proceed with the categorisation of emotions in EEG datasets. This hybrid model combine [6], [12], [13] deep studying constructions like Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Support Vector Machines (SVMs) with conventional tool studying approaches enhancing general perform [14], [15]. Therefore, synthesising the two procedures in the model enables the efficacious capturing of spatial and temporal patterns inherent in the EEG records for accurate emotion reputation. This gives a unified approach that appreciably enhances type accuracy and flexibility, as a result making it a helpful tool in the interpretation of feelings in complicated EEG recordings.

A. Hybrid CNN with SVM

The integration of a Convolutional Neural Network (CNN) and Support Vector Machine (SVM) classifier is employed for emotion classification in EEG datasets [16]. This hybrid method combines CNN's ability to extract capabilities with SVM's electricity in classification. CNN is used to extract spatial features from EEG facts, which can be then categorised into feelings via SVM. This combined version efficiently captures each spatial and temporal patterns, improving the accuracy of emotion category in EEG recordings, mainly in complicated and high-dimensional data situations. Refer to Fig. 2 for visible representation. Below is an algorithmic define of the steps accomplished in the code:

Input: Training and Testing Data (X_train, X_test, Y_train), Original Labels (Y_train), Parameters

Output: Trained Models (CNN, SVM), Predictions (CNN (Y_pred_cnn), SVM (Y_pred_svm)), Performance Metrics (Accuracy, Classification Report, Additional Metrics), Visualizations (Training History Plot, Confusion Matrix Plot)

Step 1: Start

Step 2: Import necessary libraries: a) Import 'pandas' as 'pd'. b) Import 'train_test_split' from 'sklearn. Model _selection'. c) Import 'StandardScaler' from 'sklearn. preprocessing'. d) Import necessary modules from 'keras': "Sequential, Dense, Conv1D, Flatten, and MaxPooling1D". e) Import 'SVC' from 'sklearn.svm'. f) Import 'accuracy_score' and 'classification_report' from 'sklearn. Metrics '. k) Import 'numpy' as 'np'.

Step 3: Reshape data for CNN: a) Reshape the training and testing data for 1D Convolutional Neural Network (CNN) using 'reshape'. b) Assuming it's EEG data, reshape to (number_of_samples, time_steps, 1).

Step 4: One-hot encode the labels: Use 'to_categorical' from 'tensorflow. keras. utils ' to convert the categorical labels (Y_train and Y_test) to one-hot encoded format.

Step 5: Define the 1D CNN model: a) Initialize a sequential model (`cnn_model`). b) Add a 1D convolutional layer with 64 filters, kernel size 3, and ReLU activation. c) Add a 1D MaxPooling layer with pool size 2. d) Flatten the output. e) Add a dense layer with 50 units and ReLU activation. e) Add the output layer with 3 units (for 3 classes) and softmax activation.

2025 شباط العـدد 16
No.16 Feb 2025
المجلة العراقية للبحوث الإنسانية والإجتماعية والعلمية
Iraqi Journal of Humanitarian, Social and Scientific Research
Print ISSN 2710-0952 Electronic ISSN 2790-1254

Step 6: Compile the model: Compile the model use-ing 'adam' optimizer and 'categorical_crossentropy' loss.

Step 7: Train the model with one-hot encoded labels: a) Train the model on the reshaped training data (X_train_cnn) with one-hot encoded labels (Y_train_one_hot). b) Use 10 epochs and a batch size of 32.

Step 8: Plot training history: a) Create a DataFrame (histdf) from the training history. b) Plot training accuracy and loss using 'matplotlib'.

Step 9: Extract features from the trained CNN: Use the trained CNN model to extract features from the reshaped training and test data.

Step 10: Train an SVM on the extracted features: a) Initialize an SVM model (svm_model) with a linear kernel. b) Fit the SVM model on the extracted features and the original training labels.

Step 11: Make predictions with the SVM: Use the trained SVM model to make predictions on the extracted features from the test set.

Step 12: Evaluate the model: a) Calculate and print accuracy. b) Print classification report.

Step 13: Calculate additional evaluation metrics: a) Calculate and print "precision, recall, f1-score, mean squared error, and root mean squared error". b) Generate and display a confusion matrix using 'confusion_matrix' and 'seaborn'.

Step 14: End

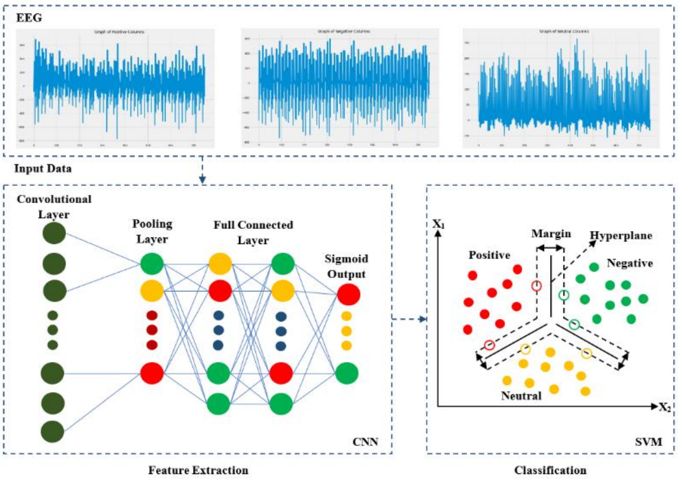


FIGURE 2. - The Hybrid CNN with SVM algorithm

B. Hybrid RNN with SVM

The proposed methodology for using an RNN (Recurrent Neural Network) and SVM (Support Vector Machine) for emotion classification is a process that takes several steps. First of all, RNN can handle temporal dependencies in the emotion data, for example, sequential sensor data. The RNN processes this sequential data and learns patterns that belong to different emotion classes [17]. The output of the RNN is then used as the features, and contains only learned temporal information. Then, these features are passed to an SVM for classification [14]. The SVM creates the best decision boundary in the feature space that discriminates the different emotion classes. This approach utilizes the temporal analysis capability of RNN and the decision boundary establishment power of SVMs to classify diverse emotions patterns. It can be noted that this hybrid model works better than both models in the aspects of accuracy and robustness of the emotion classification tasks. For example, as can be seen in Fig. 3. Below is the pseudocode of the steps taken in the code:

Input: Initial dataset (X, y), Parameters, StandardScaler, RNN model, SVM model with a linear kernel
Output: Split dataset, Standardized features, Reshaped data for RNN, One-hot encoded labels, Trained RNN model, Training history (accuracy, loss), Extracted features from RNN (X_test_features), Trained SVM model, Predictions from SVM model (y_pred), Evaluation metrics, Visualization
Step 1: Start
Step 2: Split the dataset into training and test sets using the train_test_split function with a test size of 0.2 and a random state of 42.
Step 3: Standardize the features using StandardScaler for both the training and test sets.
Step 4: Reshape the standardized data for a 1D CNN assuming EEG data structure.
Step 5: One-hot encode the labels for the training and test sets.
Step 6: Define an RNN model with two LSTM layers followed by Dense layers for classification.
Step 7: Compile the RNN model using Adam optimizer and categorical cross-entropy loss.
Step 8: Train the RNN model on the training data for 10 epochs with a batch size of 32.
Step 9: Plot the training accuracy and loss using Matplotlib.
Step 10: Extract features from the trained RNN model for the test data.
Step 11: Train an SVM model with a linear kernel on the extracted features.
Step 12: Make predictions using the SVM model on the test features.
Step 13: Evaluate the SVM model's accuracy, precision, recall, f1-score, mean squared error, and root mean squared error.
Step 14: Plot a confusion matrix and display the classification report for the SVM model.
Step 15: End

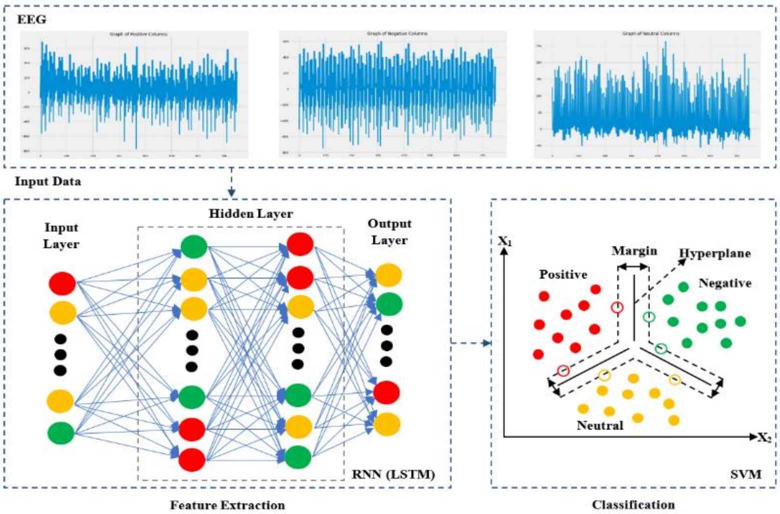


FIGURE 3. - The Hybrid RNN with SVM algorithm

2.3. CLASSIFICATION AND EVALUATION

Using EEG datasets for emotional classification, the process uses machine learning algorithms to classify emotional states and monitor their performance After training a model with labeled data, it is heard by analytical reasoning that accuracy, precision, recall, F1-score is used to measure and effectiveness [19], [20] Purpose of this iterative method The accuracy of detection of sensors improves the accuracy of the model. The combination of a rigorous analytical framework ensures the reliability of the sensory classification results, and provides a deeper understanding of sensory states in EEG recordings

3. RESULTS AND DISCUSSION

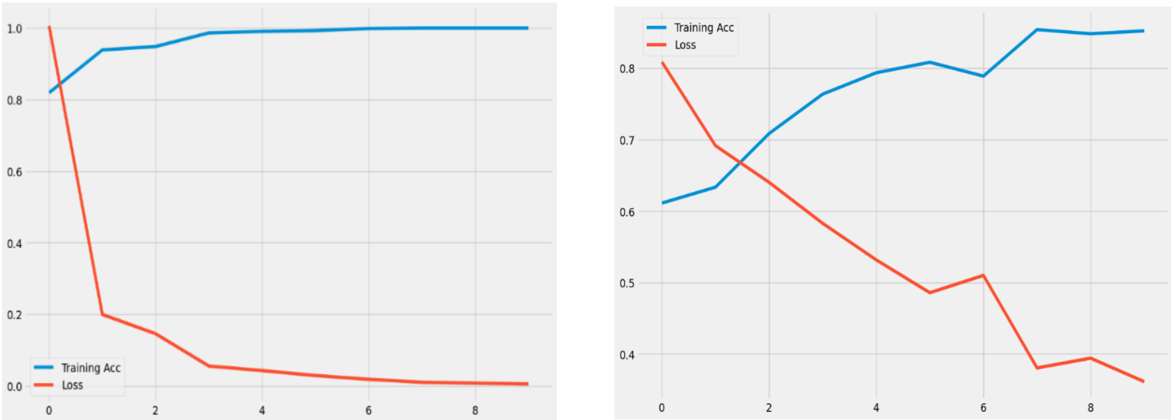
In this section, we discuss the results of the hybrid system our proposed. In which the CNN and RNN deep learning algorithms was adopted to train and test the system with the SVM machine learning algorithm to classify the results into three groups of emotions (natural, sad, and happy). Table 1 -(a), and Fig. 4 -(a), show the training phase where we achieved 100% accuracy and 0% loss. As shown in Fig. 4 -(a), the loss index, which represents the orange line, decreased and the accuracy index, which represents the blue line, increased after they reached Epoch 10/10. While, Table 1 -(b), and Fig. 4 -(b), show the training phase where we achieved 85% accuracy and 36% loss. As shown in Fig. 4 -(b), the loss index, which represents the orange line, decreased and the accuracy index, which represents the blue line, increased after they reached Epoch 10/10.

Table 1. - The epochs for training the hybrid model: a) CNN+SVM and, b) RNN+SVM

Epochs	CNN+SVM		RNN+SVM	
	Loss	Accuracy	Loss	Accuracy
Epoch 1/10	1.0059	0.8194	0.8086	0.6111
Epoch 2/10	0.1997	0.9390	0.6915	0.6334
Epoch 3/10	0.1460	0.9484	0.6402	0.7085

المجلة العراقية للبحوث الإنسانية والإجتماعية والعلمية
Iraqi Journal of Humanitarian, Social and Scientific Research
Print ISSN 2710-0952 Electronic ISSN 2790-1254

Epoch 4/10	0.0556	0.9865	0.5828	0.7636
Epoch 5/10	0.0430	0.9906	0.5316	0.7935
Epoch 6/10	0.0291	0.9930	0.4857	0.8082
Epoch 7/10	0.0186	0.9988	0.5100	0.7889
Epoch 8/10	0.0104	1.0000	0.3805	0.8540
Epoch 9/10	0.0078	1.0000	0.3941	0.8481
Epoch 10/10	0.0055	1.0000	0.3613	0.8522



(a) **(b)**

FIGURE 4. - The epochs for training the hybrid model: a) CNN+SVM and, b) RNN+SVM

Table 2, shows an explanation of the results we reached in the training phases. In which we used the following (Precision, Recall, F1-Score, MSE, RMSE, and Accuracy) to evaluate the models. The work of the proposed system was divided into two phases: the first was training, and the second was testing with a total of 2132 data sets. In the training phase, 80% of the total data was approved, as the algorithms achieved an accuracy of 89% and 87% for each of the CNN+SVM and RNN+SVM algorithms for a data set of 1705.

Table 2. - The accuracy results of machine learning algorithms (Training and Testing)

DL+ML	**Training**	
	CNN+SVM	**RNN+SVM**
Precision	0.98	0.87
Recall	0.98	0.87
F1-Score	0.98	0.87
MSE	0.05	0.39
RMSE	0.23	0.62
Accuracy	0.98	0.87
Support	1705	

Print ISSN 2710-0952 Electronic ISSN 2790-1254

A confusion matrix serves as a tabular representation for evaluating the performance of a classification model on a specific set of test data with known true labels. It comprises rows representing actual labels and columns indicating expected labels. Each cell indicates the count of cases falling into its corresponding category. Correct classifications are depicted along the diagonal, with misclassifications shown in cells outside the diagonal. The provided summary encapsulates the key values outlined in Fig. 5.

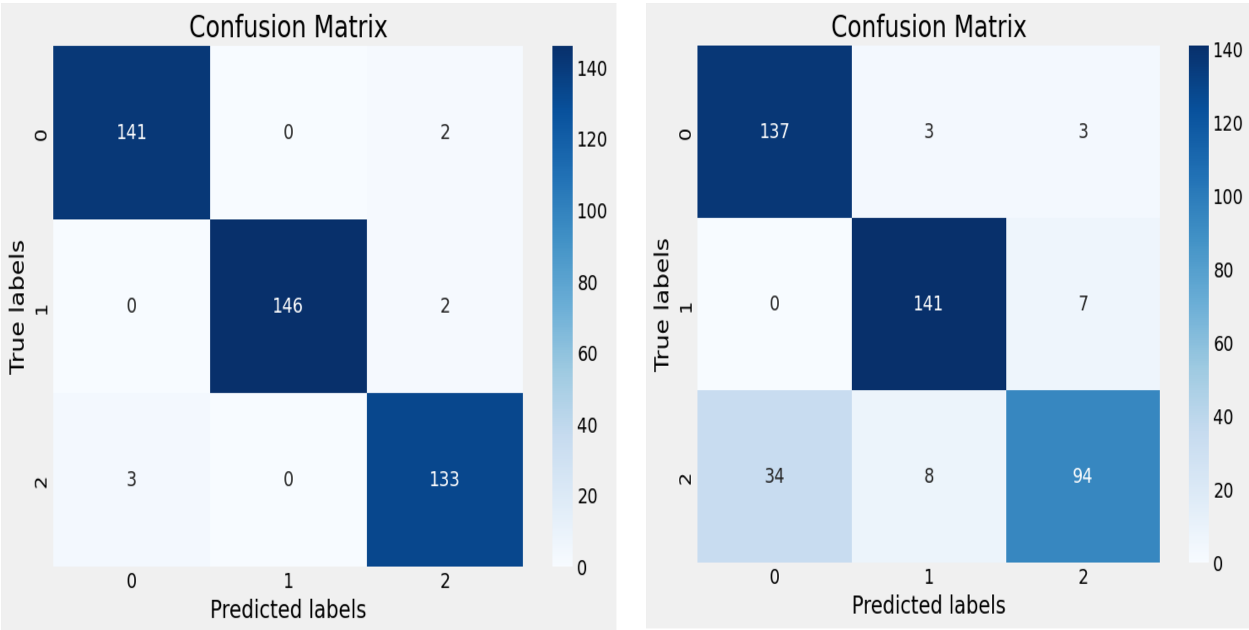
Fig. 5: C-Matrix for Test of Hybrid CNN with SVM Algorithms:

- The model naturally classified 141 cases as happy, 146 cases as sad, and 133 cases as happy. These are the real positives of each category.
- The model incorrectly classified 3 cases as happy, when they were actually natural. This is a false negative for the natural class, and a false positive for the happy class.
- The model incorrectly classified 2 states as natural, when they were actually happy. This is a false negative for the happy category, and a false positive for the natural category.
- The model incorrectly classified 2 cases as sad, when they were actually happy. This is a false negative for the happy category, and a false positive for the sad category.

Fig. 5: C-Matrix for Test of Hybrid RNN with SVM Algorithms:

- The model naturally classified 137 cases as happy, 141 cases as sad, and 94 cases as happy. These are the real positives of each category.
- The model incorrectly classified 34 cases as happy, when they were actually natural. This is a false negative for the natural class, and a false positive for the happy class.
- The model incorrectly classified 3 cases as natural, when they were actually sad. This is a false negative for the sad class, and a false positive for the natural class.
- The model incorrectly classified 8 cases as happy, when they were actually sad. This is a false negative for the sad class, and a false positive for the happy class.
- The model incorrectly classified 3 states as natural, when they were actually happy. This is a false negative for the happy category, and a false positive for the natural category.
- The model incorrectly classified 7 cases as sad, when they were actually happy. This is a false negative for the happy category, and a false positive for the sad category.

Table 3, shows the classification results our obtained in the testing phase. The data was classified into several groups. The first group represented if the feelings were natural (0) with a total of 148, the second group represented if the feelings were sad (1) with a total of 143, and the last group represented if the feelings were happy (2) with a total of 136 for the proposed hybrid system in which our adopted the DL with ML algorithms (CNN+SVM and, RNN+SVM). The rating for each classification is calculated based on (Precision, Recall, F1-Score, and Accuracy). In which we achieved 98% accuracy for CNN+SVM while, achieved 87% accuracy for RNN+SVM.



(a) (b)

FIGURE 5. - C-Matrix for Test of Hybrid DL with ML Algorithm: 1) CNN+SVM and, b) RNN+SVM

Table 3. - The results for classifying hybrid DL with ML algorithms (Testing)

DL with ML – Models	Class	Precision	Recall	F1-Score	Support
Hybrid CNN with SVM Algorithm	Natural (0)	0.98	0.99	0.98	148
	Negative (1)	1.00	0.99	0.99	143
	Positive (2)	0.97	0.98	0.97	136
	Accuracy		0.98		427
Hybrid RNN with SVM Algorithm	Natural (0)	0.80	0.96	0.87	148
	Negative (1)	0.93	0.95	0.94	143
	Positive (2)	0.90	0.69	0.78	136
	Accuracy		0.87		427

4. CONCLUSION

Our research has demonstrated the effectiveness of adopting hybrid algorithms for emotion classification using EEG data. The first hybrid algorithm, combining CNN with SVM, achieved an impressive accuracy of 98%, showcasing the power of integrating deep learning with traditional machine learning methods. Similarly, the second hybrid algorithm, combining RNN with SVM, achieved a commendable accuracy of 87%. Our findings underscore the importance of leveraging diverse algorithms to capture both spatial and temporal patterns in EEG signals, leading to more accurate emotion classification. With a dataset comprising 2132 samples, our study reinforces the value of robust data collection and preprocessing techniques in developing reliable emotion classification models.

FUNDING
None

ACKNOWLEDGEMENT
None

CONFLICTS OF INTEREST
The author declares no conflict of interest.

REFERENCES
[1] E. P. Torres, E. A. Torres, M. Hernández-Álvarez, and S. G. Yoo, "EEG-based BCI emotion recognition: A survey," *Sensors*, vol. 20, no. 18, p. 5083, 2020.
[2] M. Koctúrová and J. Juhár, "An overview of BCI-based speech recognition methods," in *2018 World Symposium on Digital Intelligence for Systems and Machines (DISA)*, IEEE, 2018, pp. 327–330.
[3] B. Kaur, D. Singh, and P. P. Roy, "EEG based emotion classification mechanism in BCI," *Procedia Comput Sci*, vol. 132, pp. 752–758, 2018.
[4] M. Quiles Pérez, E. T. Martínez Beltrán, S. López Bernal, G. Martínez Pérez, and A. Huertas Celdrán, "Analyzing the impact of Driving tasks when detecting emotions through brain–computer interfaces," *Neural Comput Appl*, vol. 35, no. 12, pp. 8883–8901, 2023.
[5] R. Sánchez-Reolid *et al.*, "Artificial neural networks to assess emotional states from brain-computer interface," *Electronics (Basel)*, vol. 7, no. 12, p. 384, 2018.
[6] X. Wu and J. Dai, "A Deep-Learning-Based Neural Decoding Framework for Emotional Brain-Computer Interfaces," *arXiv preprint arXiv:2303.04391*, 2023.
[7] R. Chalupnik, K. Bialas, Z. Majewska, and M. Kedziora, "Using simplified EEG-based brain computer interface and decision tree classifier for emotions detection," in *International Conference on Advanced Information Networking and Applications*, Springer, 2022, pp. 306–316.
[8] M. Naji, M. Firoozabadi, and P. Azadfallah, "Emotion classification during music listening from forehead biosignals," *Signal Image Video Process*, vol. 9, pp. 1365–1375, 2015.

[9] C. Ardito *et al.*, "Brain computer interface: Deep learning approach to predict human emotion recognition," in *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, IEEE, 2022, pp. 2689–2694.

[10] A. J. A. AlBdairi, Z. Xiao, and M. Alghaili, "Identifying ethnics of people through face recognition: A deep CNN approach," *Sci Program*, vol. 2020, pp. 1–7, 2020.

[11] B. Liu, L. Yu, C. Che, Q. Lin, H. Hu, and X. Zhao, "Integration and Performance Analysis of Artificial Intelligence and Computer Vision Based on Deep Learning Algorithms," *arXiv preprint arXiv:2312.12872*, 2023.

[12] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.

[13] S. Salman and X. Liu, "Overfitting mechanism and avoidance in deep neural networks," *arXiv preprint arXiv:1901.06566*, 2019.

[14] E. E. Bron, M. Smits, W. J. Niessen, and S. Klein, "Feature selection based on the SVM weight vector for classification of dementia," *IEEE J Biomed Health Inform*, vol. 19, no. 5, pp. 1617–1626, 2015.

[15] W. Gong *et al.*, "A novel deep learning method for intelligent fault diagnosis of rotating machinery based on improved CNN-SVM and multichannel data fusion," *Sensors*, vol. 19, no. 7, p. 1693, 2019.

[16] A. Saidi, S. Ben Othman, and S. Ben Saoud, "A novel epileptic seizure detection system using scalp EEG signals based on hybrid CNN-SVM classifier," in *2021 IEEE Symposium on Industrial Electronics & Applications (ISIEA)*, IEEE, 2021, pp. 1–6.

[17] M. K. Chowdary, J. Anitha, and D. J. Hemanth, "Emotion recognition from EEG signals using recurrent neural networks," *Electronics (Basel)*, vol. 11, no. 15, p. 2387, 2022.

[18] N. Pandey and O. Sharma, "Emotion Recognition Classification Using an EEG-based Brain Computer Interface System based on Different Machine Learning Models," in *2022 2nd International Conference on Innovative Sustainable Computational Technologies (CISCT)*, IEEE, 2022, pp. 1–5.

[19] Ž. Vujović, "Classification model evaluation metrics," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 6, pp. 599–606, 2021.

[20] D. S. Al-Azzawi, "Evaluation of Genetic Algorithm Optimization in Machine Learning.," *Journal of Information Science & Engineering*, vol. 36, no. 2, 2020.