Application of Support Vector Machine in Measuring Stress Levels Based on EEG Signals

Bryan Wijaya[a], Delima Sitanggang[b], Brandon Lee[c], Vicky Angie[d], Eric Simon Giovanni Siahaan[e]
[a,b,c,d,e]*Department of Science and Technology, Universitas Prima Indonesia*
Corresponding Author:
[a]*bryanwijaya153@gmail.com*

ABSTRACT

This study aims to classify stress levels based on electroencephalography (EEG) signals using the Support Vector Machine (SVM) algorithm. The data used in this study came from 21 subjects with a total of 379 datasets, which included the main variables of Subject, Electrode Channel (E), Theta, Beta 1, and Beta 2. Preprocessing was done to ensure data quality, including blank data elimination, normalization, and feature engineering. One of the main features developed was the Beta Average, which was obtained by calculating the average between Beta 1 and Beta 2, and stress level classification, which was determined based on the comparison between the Beta Average and Theta. The SVM algorithm was applied to build a stress classification model with an initial stage of manual calculation to understand the basic concepts, followed by the Python programming language implementation. The evaluation results show that the developed model has an accuracy of 92.76%, with the highest precision, recall, and f1-score values reaching 100% and the lowest value of 85%. The confusion matrix analysis showed that the model could classify low stress with 100% accuracy, while it reached 87.8% for high stress. The findings of this study prove that the SVM algorithm effectively classifies EEG signal-based stress levels. This model can be the basis for further development of stress detection methods, especially in mental health and neuroinformatics applications.

Keywords : EEG, Stress Classification, Support Vector Machine, Beta Average, Brainwaves.

INTRODUCTION

Drug abuse is a serious public health challenge that has a profound impact on the mental and physical health of individuals (Damasela et al., 2022). Drug use can significantly impair brain function, leading to extreme behavioral changes and risk to self and others. Substances such as stimulants, depressants, and hallucinogens interact with the central nervous system, affecting neurotransmitters and causing emotional and mental health disorders, such as stress, anxiety, and depression (Hidayat & Wirananda, n.d., 2021). In drug-using prisoners, the stress of the prison environment can exacerbate these conditions, intensifying stress and lowering overall mental health (Juan Felik Sitorus & Kunci, 2023).

Neuroimaging technology, such as Electroencephalography (EEG), is becoming a useful tool for understanding stressful conditions in individuals experiencing drug abuse. EEG allows non-invasive recording of the brain's electrical activity, with brain waves such as alpha, beta, and theta providing information related to mental states. Alpha waves are associated with relaxation, while beta and theta are associated with stressful states. By analyzing these patterns, changes in brain activity related to stress levels can be identified (Jobst et al., 2020). Support Vector Machine (SVM) has developed into an effective machine learning technique for classifying and analyzing EEG data. SVM works by finding the optimal hyperplane that separates data into different categories with maximum (Soufineyestani et al., 2020). In EEG signal analysis, SVM can classify brainwave patterns based on extracted features, handle high-dimensional data, and identify subtle patterns that traditional methods may not detect (Petrescu et al., 2020). This technique also allows handling non-linear data using kernel tricks, thus mapping the data to higher dimensions for better separation.

This research aims to apply SVM in EEG signal analysis to measure stress levels in drug-using prisoners. Through the use of EEG, SVM models were developed to identify and classify stress levels accurately, providing deeper insight into the impact of drugs on mental health l (Aziz et al., n.d., 2023). Hopefully, the results of this study can contribute to the development of better stress detection methods and support more effective rehabilitation programs. By exploring relevant EEG features and building a model capable of distinguishing between stress and non-stress conditions, this research seeks to improve the accuracy and sensitivity of EEG-based stress detection (Wahyuni & Kusumodestoni, 2024).

METHODS

Type of Research

This research aims to apply a Support Vector Machine (SVM) in EEG signal analysis to measure stress levels in drug-using prisoners. Utilizing EEG, SVM models were developed to identify and classify stress levels with high accuracy, providing greater insight into the impact of drug abuse on mental health (Laili, n.d., 2022). This research analyzed beta and theta brainwaves, which explored brain activity patterns relevant to stressful conditions. Beta waves are in the frequency range of 13-30 Hz and are associated with alertness, anxiety, and stress, while theta waves are 4-7 Hz and are often associated with high-stress conditions and fatigue (Saputra & Judul, 2024).

Working Procedure

In order for the research to run well and be completed on time, there are research work procedures. The working procedures of this research are as follows:

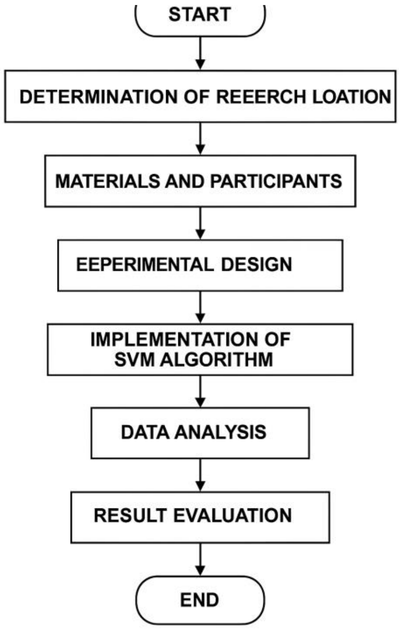


Figure 1. Research Workflow

Determination of Research Location

This research was conducted at one of the correctional institutions in North Sumatra Province through collaboration between Prima Indonesia University and the relevant correctional institutions.

Research Materials and Respondents

The equipment and materials used in this study include Win EEG software, amplifiers, gels, and electro caps, which record brain electrical activity on the scalp's surface. The amplifier amplifies the amplitude of the EEG signal obtained from the electrodes on the scalp, while the gel increases the contact between the electrodes and the scalp. The respondents in the study were 21 male subjects ranging in age from 20 to 30.

Figure 2. Electroencephalogram (EEG)-Based Data Acquisition Procedure

Experiment Design

The experiment was conducted for two minutes with eyes closed. The electrodes were precisely positioned on the subject's head using a special fastening device. Researchers applied interview techniques to obtain data and information related to the subject's life history. Meanwhile, a computer was used to control the EEG device and analyze the collected data. The experiment results were also influenced by the room conditions maintained from external disturbances, such as light and sound.

Observation of the subject's behavior.

The data were systematically documented during the experiment, and EEG software was utilized to analyze and interpret the recorded brain activity.

In this EEG experiment, data was recorded using an EEG device from Mitsar with 21 electron capture channels. Nine channels were placed in specific areas of the scalp, and the other two were auricular electrodes (A1 and A2), which were positioned in the left and right ear areas, respectively. A combined reference electrode is used for the auricular or ear area.

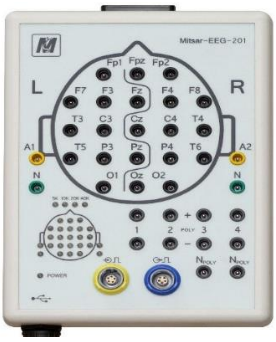


Figure 3. Mitsar EEG 201

EEG Data Preprocessing

After collecting EEG signal data, the next step is data preprocessing. Data preprocessing is a series of steps that aim to clean, process, and adjust the raw EEG signal before it is used in further analysis with the Support Vector Machine (SVM) (Abdusyukur, 2023).

This process is important to ensure that the data is free from noise and anomalies so that the analysis can be done accurately and provide optimal results.

Application of SVM Algorithm

The Support Vector Machine (SVM) algorithm is a classification technique widely used in data analysis to separate objects into different classes based on patterns or characteristics learned from training data. SVM works by finding a hyperplane that optimally separates data in different classes by maximizing the margin or distance between groups (Angraina & Putri, 2022). As one of the methods in supervised learning, SVM uses labeled data to train the model to understand the differences between existing classes.

a. Equetion 1.

In the Support Vector Machine (SVM) method, the optimal separation between two data classes is done by finding a hyperplane that maximizes the distance between different data groups. The hyperplane is represented by the following equation.

$$w.x + b = 0$$

Where w is the weight vector that determines the orientation of the hyperplane, x is the input data vector, and b is the bias that sets the position of the hyperplane in feature space.

b. Equetion 2.

When the data cannot be linearly separated, SVM uses a kernel function to transform the data to a high-dimensional space, thus allowing linear separation. Frequently used kernel functions, such as Radial Basis Function (RBF), use the following equation.

$$K(x_i, x_j) = \exp(-\gamma||x_i - x_j||^2$$

Where x_i dan x_j are data vectors in the original feature space, and γ is a parameter that sets the degree of separation smoothness

Result Analysis and Evaluation

After obtaining the final classification results, evaluation of model performance, analysis of EEG signal patterns, and identification of anomalies in the classification results are carried out to achieve optimal results (Talib et al., n.d, 2024.). Based on the analysis, the authors can draw conclusions relevant to the research objectives, which are then used to interpret the classification results and their implications for stress conditions in the prisoners under study.
Analisis dan Evaluasi Hasil

RESULTS

Data Identification

In this study, the data used comes from 21 research subjects and there are 379 datasets which are E channels (electrodes) which consist of five main variables that play an important role in analyzing EEG signal patterns. These variables include Subject, Channel E (electrode), THETA, BETA 1, and BETA 2. Each of these variables has a significant contribution in understanding the dynamics of brain activity associated with stress levels, as explained in the following table visualization:

Table 1. Dataset Details

No	Subject	Saluran E	Theta	Beta 1	Beta 2
1	S1	Fp1-Ref	6.78	1.85	2.02
2	S1	F7-Ref	4.69	1.61	1.46
3	S1	F3-Ref	8.58	2.53	2.75
4	S1	Fz-Ref	8.73	2.38	2.68
5	S1	F4-Ref	8.01	2.65	3.02
6	S1	F8-Ref	4.84	2.27	2.33
…	…	…	…	…	…
377	S21	T6-Ref	3.91	14.65	19.53
378	S21	O1-Ref	3.91	15.63	19.53
379	S21	O2-Ref	3.91	15.63	19.53

Data Preprocessing

Data preprocessing is a series of processes that aim to clean, adjust, and prepare raw data so that it is ready for further analysis later in the research. In this research, the data preprocessing stage is carried out to ensure that the EEG signal used is free from interference or noise and in a format that can be optimally processed by the Support Vector Machine (SVM) model.

This process includes eliminating empty data, data normalization, feature engineering, and removal of irrelevant variables to improve accuracy and analysis structure. Through these stages, the processed data will support the development of a more accurate and organized EEG-based stress level classification model.

Data normalization

Data normalization is an important step in the preprocessing process to ensure that the dataset used is clean, structured, and ready to be analyzed according to research needs (Eldo et al., 2024). In this research, data normalization is carried out through several stages as follows:

a. Checking for empty and duplicate data and customizing the dataset

In this research, after checking, there is no empty or duplicate data in the dataset, so there is no need to delete the data. However, some data cannot be directly used in the algorithm because they still have non-numeric data types. Therefore, data type conversion was carried out on certain variables previously of object type to integer type. This conversion process is done so that the data can be applied to the algorithm more optimally, as shown in the following figure:

```
[178] print("Jumlah Data Duplikat : ", df.duplicated().sum())

Jumlah Data Duplikat :  0

print(df.isnull().sum())

Subject      0
Saluran E    0
THETA        0
BETA 1       0
BETA 2       0
dtype: int64
```

```
           0
Subject    object
Saluran E  object
THETA      float64
BETA 1     float64
BETA 2     float64

dtype: object

181] label_encoder_saluran = LabelEncoder()
     label_encoder_subject = LabelEncoder()

182] label_encoder_saluran = LabelEncoder()
     df['Saluran_E_int'] = label_encoder_saluran.fit_transform(df['Saluran E'])

183] df['Subject_int'] = label_encoder_subject.fit_transform(df['Subject'])
```

Figure 4. Dataset normalization

b. Feature Engineering Implementation

Feature engineering is creating or modifying dataset features to improve machine learning models' performance. This step involves selecting, transforming, or adding new features that are relevant to the analysis needs, so that the algorithm can more easily recognize significant patterns (Rabbani et al., 2023), 2023), in this study the authors performed feature engineering by adding 2 new variables sourced from existing dataset variables, where the variable is the beta average variable which is obtained from dividing the beta1 and beta 2 variables to get an average value which will later be used for the application of the algorithm, and secondly there is a classification variable which is used as the final goal of the research classification which is obtained from the application where if the beta average value is greater than theta, This indicates a high stress phase in which the brain is in a state of over activity or beta wave activity dominates and this condition is where the subject feels anxiety, high alertness and stress, and if the classification of low stress when theta is greater than beta average brain activity is more relaxed, which indicates a calmer and less stressful state of mind, for its application in python programming can be seen in the following figure.

```
[15] df['BETA_avg'] = df[['BETA 1', 'BETA 2']].mean(axis=1)

[16] def classify_stress(row):
         if row['BETA_avg'] > row['THETA']:
             return 'Stres Tinggi'
         else:
             return 'Stres Rendah'

[17] df['Hasil klasifikasi'] = df.apply(classify_stress, axis=1)
```

Figure 5. Application of Feature Engineering

	Subject	Saluran E	THETA	BETA 1	BETA 2	Saluran_E_int	Subject_int	BETA_avg	Hasil klasifikasi
0	s1	Fp1-Ref	6.78	1.85	2.02	17	11	1.935	Stres Rendah
1	s1	F7-Ref	4.69	1.61	1.46	5	11	1.535	Stres Rendah
2	s1	F3-Ref	8.58	2.53	2.75	3	11	2.640	Stres Rendah
3	s1	Fz-Ref	8.73	2.38	2.68	7	11	2.530	Stres Rendah
4	s1	F4-Ref	8.01	2.65	3.02	4	11	2.835	Stres Rendah
5	s1	F8-Ref	4.84	2.27	2.33	6	11	2.300	Stres Rendah
6	s1	T3-Ref	4.81	2.18	2.17	13	11	2.175	Stres Rendah
7	s1	C3-Ref	8.52	2.67	2.72	0	11	2.695	Stres Rendah
8	s1	Cz-Ref	11.20	3.09	3.79	2	11	3.440	Stres Rendah
9	s1	C4-Ref	7.55	2.90	2.84	1	11	2.870	Stres Rendah
10	s1	T4-Ref	3.60	25.67	28.89	14	11	27.280	Stres Tinggi
11	s1	T5-Ref	5.20	2.01	1.95	15	11	1.980	Stres Rendah
12	s1	P3-Ref	9.40	2.94	2.97	10	11	2.955	Stres Rendah
13	s1	Pz-Ref	9.48	2.93	3.00	12	11	2.965	Stres Rendah
14	s1	P4-Ref	6.76	2.68	2.31	11	11	2.495	Stres Rendah

Figure 6. Dataset After Data Preprocessing

Application of Support Vector Machine (SVM) Algorithm

Before applying the SVM algorithm using the Python programming language, a manual calculation will be carried out using the dataset. This step aims to understand the calculation process of the Support Vector Machine algorithm manually, as described below.

a. SVM Manual Calculation
b. Sample data table

Table 2. Sample Data

Subject Int	Saluran E_int	Beta 1	Beta 2	Beta Avg	Theta	Label
17	11	1.85	2.02	1.935	6.78	-1 (Stres rendah)
5	11	1.61	1.46	1.535	4.69	-1 (Stres rendah)
5	12	14.65	19.53	17.09	4.88	1 (Stres tinggi)
1	12	16.60	22.46	19.53	3.91	1 (Stres tinggi)

The data above are some samples taken from the primary dataset used. Since the subject and channel data will be involved in the calculation, each value in the data has been converted into a unique number to represent the characteristics of each data numerically.

This study determines the stress level classification by comparing the average beta and theta brainwave activity. Beta waves (14-26 Hz) are associated with increased cognitive activity and high alertness, which in some studies have also shown a correlation with increased stress and anxiety. In contrast, theta waves (4-8 Hz) are associated with relaxation and emotional processing, which may indicate lower stress levels (Nurrachmat Hidayat, n.d., 2023). Based on these findings, if the beta average value is greater than theta, the condition is categorized as high stress. In comparison, if the beta average value is smaller or equal to theta, the condition is categorized as low stress.

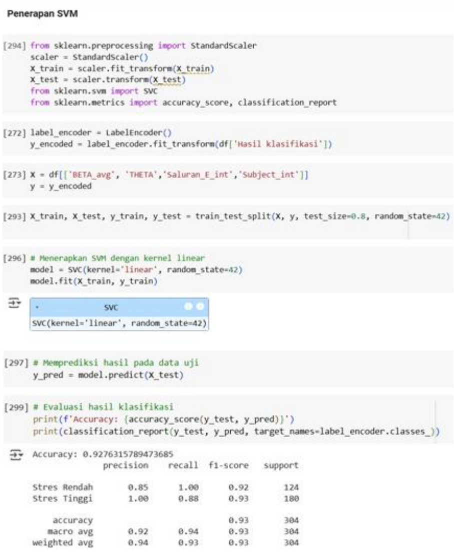
For application in programming algorithms, stress categories need to be expressed in numerical format to be used in programming algorithms. Therefore, in this study:

1. High Stress is labeled 1
2. Low Stress is labeled -1Stres Tinggi diberikan label 1

This approach was chosen so the data can be more easily processed in mathematical calculations and SVM algorithms. This format also allows the model to recognize differences in stress levels more effectively through regression-based analysis, classification, or other algorithmic methods in brain signal processing (J., Syukron, et al., 2022).

Application of SVM with Python

After previously performing manual calculations to understand the work process of the SVM algorithm calculation, the author will implement the SVM algorithm using the Python programming language. The implementation can be seen in the following figure.



Penerapan SVM

```
[294] from sklearn.preprocessing import StandardScaler
      scaler = StandardScaler()
      X_train = scaler.fit_transform(X_train)
      X_test = scaler.transform(X_test)
      from sklearn.svm import SVC
      from sklearn.metrics import accuracy_score, classification_report

[272] label_encoder = LabelEncoder()
      y_encoded = label_encoder.fit_transform(df['Hasil klasifikasi'])

[273] X = df[['BETA_avg', 'THETA','Saluran_E_int','Subject_int']]
      y = y_encoded

[293] X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.8, random_state=42)

[296] # Menerapkan SVM dengan kernel linear
      model = SVC(kernel='linear', random_state=42)
      model.fit(X_train, y_train)

      SVC
      SVC(kernel='linear', random_state=42)

[297] # Memprediksi hasil pada data uji
      y_pred = model.predict(X_test)

[299] # Evaluasi hasil klasifikasi
      print(f'Accuracy: {accuracy_score(y_test, y_pred)}')
      print(classification_report(y_test, y_pred, target_names=label_encoder.classes_))

      Accuracy: 0.9276315789473685
                    precision    recall  f1-score   support

      Stres Rendah       0.85      1.00      0.92       124
      Stres Tinggi       1.00      0.88      0.93       180

          accuracy                           0.93       304
         macro avg       0.92      0.94      0.93       304
      weighted avg       0.94      0.93      0.93       304
```

Figure 7. Application of SVM Algorithm

Based on the results of the following application, it can be concluded that the results are 92.49% accurate, the lowest precision, recall, and f1-score metrics are 0.85, and the highest is 1.00, indicating that the score provides excellent performance in classifying data into high-stress and low-stress classes, and support refers to the number of actual data samples in each class contained in the test dataset.

Analysis and Evaluation of Results

Data results are analyzed through various visualization methods, including data comparison, data distribution, and accuracy results obtained after the application of the support vector machine (SVM) algorithm. This visualization aims to provide a clearer picture of the model's performance in classifying data and patterns in the research dataset.

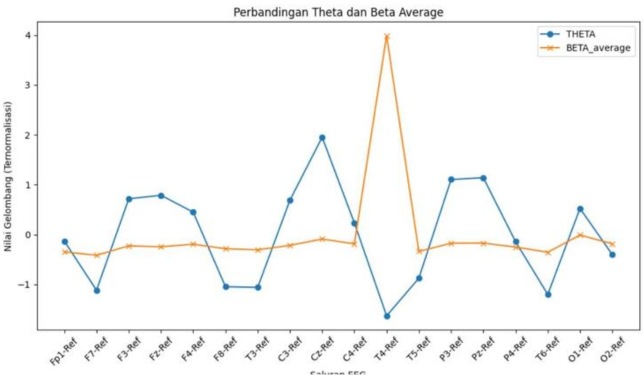


Figure 8. Comparison of Theta and Beta Average

Comparison of Theta and Beta Average

This graph shows a comparison of the normalized Theta and Beta Average wave values on each EEG channel. From these results, there is a difference in pattern between the two waves, where Beta Average values tend to be more stable than Theta. The highest peak in Beta Average indicates that a particular channel has significant activity, which is an indicator of relevance to stress classification.

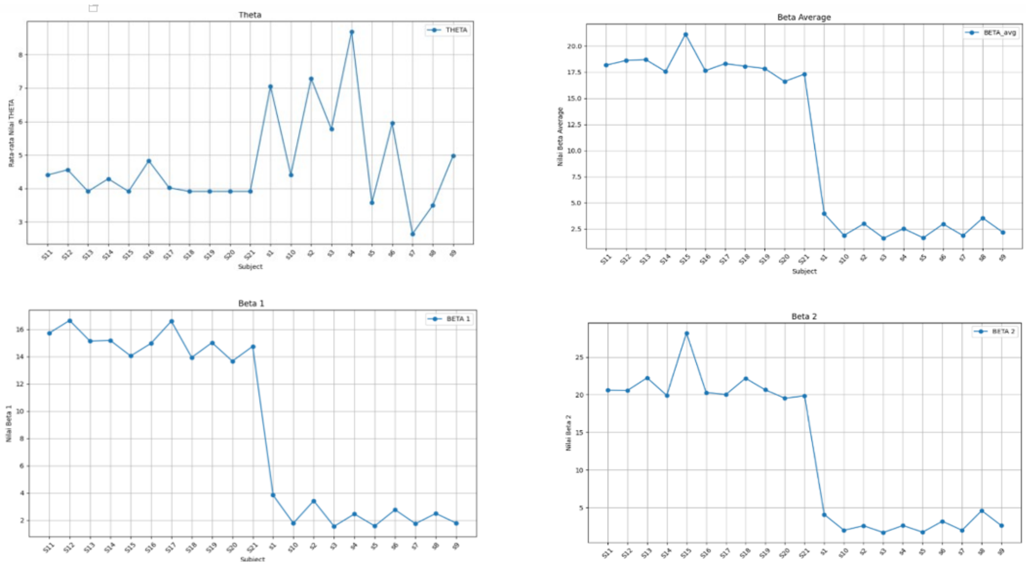


Figure 9. Data Distribution of Theta, Beta Average, Beta1, Beta2

Waveform Value Distribution by Subject

Activity Variation between Subjects: The graph shows that the waveform values (Theta, Beta Average, Beta 1, Beta 2) vary between subjects. Specific subjects, such as S10, S15, and S20, had higher average values, indicating a tendency towards higher stress levels.

Consistency Between Beta Waves: Beta 1 and Beta 2 waves showed highly consistent patterns, supporting their reliability as key features for stress classification.

Theta Wave Characteristics: Theta waves showed significant fluctuations, indicating different stress activity patterns between subjects.

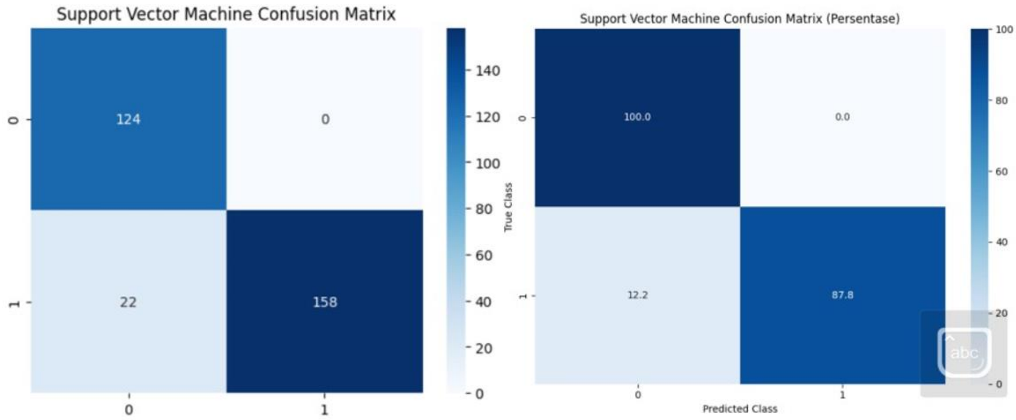


Figure 10. Confusion Matrix

Confusion Matrix

124 Low-Stress class data was predicted correctly. 158 High Stress class data were correctly predicted.

The accuracy for the Low-Stress class reached 100%, while the accuracy for the High-Stress class was 87.8%. The data was obtained from 80% of the testing set using the Support Vector Machine (SVM) algorithm. Overall, the original dataset consists of 379 samples, while the number of test data used is 304. From the test results, 124 true negatives (TN), 158 true positives (TP), 22 false negatives (FN), and zero false positives (FP) were obtained. These results show that the model performs excellently in predicting both classes, especially in identifying the Low-Stress class, which achieved perfect accuracy.

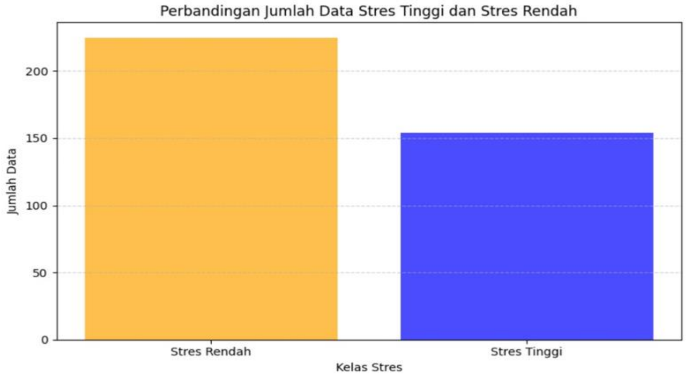


Figure 11. Comparison of the number of stress data

Comparison of high and low-stress data.

The data visualization shows that the Low-Stress class has more than the High-Stress class. Despite this, the model still provides excellent classification results in both classes, as seen in the confusion matrix. This shows that the algorithm used is adaptable to the data.

This study successfully demonstrated that the Support Vector Machine (SVM) algorithm can classify stress levels based on brain waves with excellent results. Visualization and evaluation showed that Theta and Beta wave features contributed significantly to the stress level analysis, with high accuracy results for both classes. In addition, the different brain activity patterns between subjects indicate that the data used is of strong quality to support the classification. Overall, the results of this study prove that the approach used is practical and can serve as a foundation for further development in brainwave-based stress analysis.

CONCLUSION

Based on the results of this study, the Support Vector Machine (SVM) algorithm shows excellent performance in classifying stress levels based on EEG data. The developed model achieved an accuracy rate of 92.76%, which shows high reliability in distinguishing stress categories. Evaluation of the model performance showed that the precision reached 85% for the Low-Stress class and 100% for the High-Stress class, while the recall values were 100% and 88%, respectively. In addition, the F1-score values obtained showed an optimal balance between precision and recall, with an average of 93%. These results confirm that the SVM algorithm can provide accurate and consistent predictions in identifying stress levels based on brainwave patterns. Therefore, this method has the potential to be further developed in EEG-based mental health monitoring systems, especially in the context of specific populations such as prisoners, to support early detection and more appropriate interventions.

SUGGESTION

The suggestions that can be given in this study are as follows:

1. Explore additional features of EEG data, such as Theta-Beta wave ratio or Alpha wave analysis, to enrich information in stress classification.
2. Develop ensemble-based classification models, such as Random Forest or Gradient Boosting, to compare their performance with SVM and identify the best algorithm.
3. Applying data balancing techniques, such as oversampling (SMOTE) or undersampling, to overcome class imbalance and improve classification accuracy.

REFERENCES

Abdusyukur, F. (2023). PENERAPAN ALGORITMA SUPPORT VECTOR MACHINE (SVM) UNTUK KLASIFIKASI PENCEMARAN NAMA BAIK DI MEDIA SOSIAL TWITTER. *KOMPUTA : Jurnal Ilmiah Komputer Dan Informatika*, *12*(1).

Angraina, D., & Putri, A. (2022). Analisis Sentimen Pengguna Aplikasi Google Meet Menggunakan Algoritma Support Vector Machine. *Jurnal CoSciTech (Computer Science and Information Technology)*, *3*(3), 472–478. https://doi.org/10.37859/coscitech.v3i3.4260

Aziz, F., Pancasakti Makassar, U., Wahab, A., Keperawatan, I., Karya, U., & Muna, P. (n.d.). *JOURNAL PHARMACY AND APPLICATION OF COMPUTER SCIENCES IMPLEMENTASI METODE SUPPORT VECTOR MACHINE UNTUK DETEKSI DINI PERSISTENSI OBAT DALAM FARMASI IMPLEMENTATION OF THE SUPPORT VECTOR MACHINE METHOD FOR EARLY DETECTION OF DRUG PERSISTENCE IN PHARMACY*. https://www.kaggle.com/datasets/harbhajansing

Damasela, R., Tomasouw, B. P., & Leleury, Z. A. (2022). PENERAPAN METODE SUPPORT VECTOR MACHINE (SVM) UNTUK MENDETEKSI PENYALAHGUNAAN NARKOBA. *PARAMETER: Jurnal Matematika, Statistika Dan Terapannya*, *1*(2), 111–122. https://doi.org/10.30598/parameterv1i2pp111-122

Eldo, H., Ayuliana, A., Suryadi, D., Chrisnawati, G., & Judijanto, L. (2024). Penggunaan Algoritma Support Vector Machine (SVM) Untuk Deteksi Penipuan pada Transaksi Online. *Jurnal Minfo Polgan*, *13*(2), 1627–1632. https://doi.org/10.33395/jmp.v13i2.14186

Hidayat, T., & Wirananda, H. A. (n.d.). *PENGARUH TERAPI PSIKOLOGI DAN TERAPI FISIK DALAM UPAYA REHABILITASI TERHADAP PARA PECANDU NARKOBA DI PANTI REHABILITASI AL KAMAL SIBOLANGIT DESA SUKA MAKMUR SUMATERA UTARA*.

Homepage, J., Syukron, H., Fauzi Fayyad, M., Junita Fauzan, F., Ikhsani, Y., & Rizkya Gurning, U. (2022). *MALCOM: Indonesian Journal of Machine Learning and Computer Science Comparison K-Means K-Medoids and Fuzzy C-Means for Clustering Customer Data with LRFM Model "Perbandingan K-Means K-Medoids dan Fuzzy C-Means untuk Pengelompokan Data Pelanggan dengan Model LRFM*. *2*, 76–83.

Jobst, B. C., Bartolomei, F., Diehl, B., Frauscher, B., Kahane, P., Minotti, L., Sharan, A., Tardy, N., Worrell, G., & Gotman, J. (2020). Intracranial EEG in the 21st Century. *Epilepsy Currents*, *20*(4), 180–188. https://doi.org/10.1177/1535759720934852

Juan Felik Sitorus, V., & Kunci, K. (2023). Jurnal JPILKOM (Jurnal Penelitian Ilmu Komputer) ANALISIS ARSITEKTUR JST BERDASARKAN ANGKA KEMATIAN AKIBAT OVERDOSIS NARKOTIKA MENGGUNAKAN ALGORITMA BACKPROPAGATION. In *Jurnal JPILKOM* (Vol. 1, Issue 2). Online.

Laili, U. H. (n.d.). IMPLEMENTASI DATA MINING NAÏVE BAYES DAN SVM (SUPPORT VECTOR MACHINE) DALAM PENGUMPULAN BERITA KRIMINAL DI INDONESIA PADA MASA PANDEMI COVID-19 Studi Kasus : Situs Berita Detik.com. In *Journal of Engineering and Sustainable Technology*.

Nurrachmat Hidayat, F. (n.d.). Analisis Sentimen Masyarakat Terhadap Perekrutan PPPK Pada Twitter Dengan Metode Naive Bayes … (Fajar Nurrachmat Hidayat, Sugiyono) Analisis Sentimen Masyarakat Terhadap Perekrutan PPPK Pada Twitter Dengan Metode Naive Bayes Dan Support Vector Machine. *Jurnal Sains Dan Teknologi*, *5*(2), 665–672. https://doi.org/10.55338/saintek.v5i1.1359

Petrescu, A. M., Taussig, D., & Bouilleret, V. (2020). Electroencephalogram (EEG) in COVID-19: A systematic retrospective study. *Neurophysiologie Clinique*, *50*(3), 155–165. https://doi.org/10.1016/j.neucli.2020.06.001

Rabbani, S., Safitri, D., Rahmadhani, N., Sani, A. A. F., & Anam, M. K. (2023). Perbandingan Evaluasi Kernel SVM untuk Klasifikasi Sentimen dalam Analisis Kenaikan Harga BBM. *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, *3*(2), 153–160. https://doi.org/10.57152/malcom.v3i2.897

Saputra, A. B., & Judul, H. (2024). *FINAL PROJECT REPORT TEXT-BASED SMART SURVEY SYSTEM USING SUPPORT VECTOR MACHINE METHOD FOR FACILITIES SURVEYING AT THE FACULTY OF INDUSTRIAL TECHNOLOGY*.

Soufineyestani, M., Dowling, D., & Khan, A. (2020). Electroencephalography (EEG) technology applications and available devices. In *Applied Sciences (Switzerland)* (Vol. 10, Issue 21, pp. 1–23). MDPI AG. https://doi.org/10.3390/app10217453

Talib, S., Sudin, S., & Suratin, M. D. (n.d.). *PENERAPAN METODE SUPPORT VECTOR MACHINE (SVM) PADA KLASIFIKASI JENIS CENGKEH BERDASARKAN FITUR TEKSTUR DAUN*.

Wahyuni, S. D., & Kusumodestoni, R. H. (2024). Optimalisasi Algoritma Support Vector Machine (SVM) Dalam Klasifikasi Kejadian Data Stunting. *Bulletin of Information Technology (BIT)*, *5*(2), 56–64. https://doi.org/10.47065/bit.v5i2.1247