BRNO UNIVERSITY OF TECHNOLOGY
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF INFORMATION TECHNOLOGY
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

DEPARTMENT OF COMPUTER SYSTEMS
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

ELECTROENCEPHALOGRAM (EEG) AND MACHINE LEARNING-BASED CLASSIFICATION OF VARIOUS STAGES OF MENTAL STRESS
ELEKTROENCEFALOGRAM (EEG) A KLASIFIKACE RŮZNÝCH STÁDIÍ PSYCHICKÉ ZÁTĚŽE NA ZÁKLADĚ STROJOVÉHO UČENÍ

BACHELOR'S THESIS
BAKALÁŘSKÁ PRÁCE

AUTHOR   Ing. TEREZA LAPČÍKOVÁ
AUTOR PRÁCE

SUPERVISOR   MUHAMMAD ASAD ZAHEER
VEDOUCÍ PRÁCE

BRNO 2024

Bachelor's Thesis Assignment

153457

Assignment:

1. Study the neural mechanism of mental stress and its impact on individuals.
2. Learn feature extraction, feature selection, and classification methods for analyzing EEG (Electroencephalogram) data for mental stress.
3. Identify challenges and limitations in existing methods for the detection of mental stress and its various stages through a literature review.
4. Design a machine learning model for extracting relevant features and accurately classifying mental stress and its various stages.
5. Implement and evaluate the model using appropriate datasets and evaluation metrics.
6. Critically analyze results, interpret their significance, and discuss their contribution to depression classification.

Literature:
• Based on Supervisor's Recommendation.

Requirements for the semestral defence:
• Fulfillment of items 1 to 4.

Detailed formal requirements can be found at https://www.fit.vut.cz/study/theses/

Abstract

This thesis deals with the recognition of various stress stages experienced by patients from electroencephalogram (EEG). Various Support Vector Machine (SVM) and Long Short-Term Memory (LSTM) models classifying EEG into three classes – *not stressed, moderate stressed*, and *very stressed* were created. The process of implementing such a classifier consisted of data preparation, extraction, and finally, classification. This solution also implements augmentation of data. The highest accuracy achieved in this thesis was of 90 % using the SVM model. The best LSTM model was a three-layer LSTM and achieved classification accuracy of 70 %.

Abstrakt

Tato závěrečná práce se věnuje rozpoznávání a klasifikaci různých úrovní psychické zátěže z elektroencefalogramu (EEG). V rámci této práce bylo vytvořeno několik modelů SVM a LSTM, které klasifikují EEG data do tří tříd odpovídajících mentální zátěži – *nízká mentální zátěže, střední mentální zátěž* a *vysoká mentální zátěž*. Proces vedoucí k tvorbě těchto modelů se skládal z kroků jako je úprava vstupního signálu, extrakce jeho vlastností a implementace modelu pro samotnou klasifikaci. Toto řešení taktéž obsahuje augmentaci dat. Nejvyšší dosažená přesnost klasifikace byla 90 %, a to s modelem SVM. Nejlepší LSTM model obsahoval tři vrstvy LSTM a přesnost jeho výsledné klasifikace byla 70 %.

Keywords

Machine Learning, Deep Learning, EEG, Mental Stress, Stages of Stress

Klíčová slova

Strojové učení, hluboké učení, EEG, mentální stres, fáze stresu

Reference

LAPČÍKOVÁ, Tereza. *Electroencephalogram (EEG) and machine learning-based classification of various stages of mental stress*. Brno, 2024. Bachelor's thesis. Brno University of Technology, Faculty of Information Technology. Supervisor Muhammad Asad Zaheer

Rozšířený abstrakt

Tato závěrečná práce se zabývá klasifikací různých stádií psychické zátěže pomocí strojového učení. V posledních letech jsme svědky značného nárůstu psychických problémů napříč celou populací. Důvodů může být hned několik a jedním z nich je právě chronický stres, který má často kořeny v nadměrném tlaku na pracovní výkon. Chronický stres může negativně ovlivnit jak mentální, tak zdravotní stav jedince a může být jedním ze spouštěcích faktorů úzkostných a depresivních stavů. Včasná diagnostika stavu chronického stresu a jeho řešení by mohla být velkým přínosem pro zdraví člověka. Elektroencefalografie, zkráceně EEG, je diagnostická metoda zaznamenávající elektrickou aktivitu mozku pomocí elektrod přiložených na hlavu pacienta. Jedná se o neinvazivní a cenově dostupnou metodu, která umožňuje detekovat řadu procesů zpracovávaných mozkem, a to včetně psychických problémů, jako jsou deprese, úzkosti a stres. Signál změřený pomocí EEG je ovšem velmi komplikovaný a jeho vyhodnocení bez použití moderních technologií je velmi náročné. Zde přichází na řadu algoritmy strojového a hlubokého učení, které v této oblasti prokazují dobré výsledky v klasifikačních úlohách.

Cílem této práce bylo vytvořit model strojového učení pro klasifikaci různých stádií psychické zátěže. Důležitým aspektem této práce bylo také studium související tématiky nezbytné pro její tvorbu. V teoretické části byla nejprve nastíněna architektura lidského mozku a byl popsán mechanismus stresu a jednotlivá stádia stresové odpovědi organismu na přítomnost stresoru v jeho okolí. Důležité bylo také prostudovat vlastnosti a principy EEG, včetně jejích předností a nedostatků. Větší důraz byl potom kladen na studium témat souvisejících s tvourbou samotného klasifikačního modelu. Byla probrána příprava vstupního EEG signálu, metody extrakce a selekce jeho významných vlastností a především samotné algoritmy strojového a hlubokého učení používané v pro klasifikační účely. Klíčové bylo také získat přehled o stavu odborné literatury řešící tuto tématiku.

V rámci praktické části bylo implementováno několik architektur modelů klasifikujících EEG signál do tří tříd podle pocitové úrovně mentální zátěže – *nízká mentální zátěž*, *střední mentální zátěž* a *vysoká mentální zátěž*. Pro účely trénování a testování modelů byl použit veřejně dostupný dataset SAM40. Tohle řešení nabízí řadu kombinací extrahovaných vlastností signálu, vstupních parametrů modelů a architektur modelů samotných, které byly prozkoumány prostřednictvím experimentů. Klasifikační modely implementované v rámci této práce jsou SVM, z anglického Support Vector Machine, a LSTM, z anglického Long Short-Term Memory. S modelem tvořeným kombinací tří vrstev LSTM a jedné dropout vrstvy byla dosažena přesnost klasifikace 70 %. Tento výsledek nechává prostor pro další vylepšení, kterého by mohlo být dosaženo, např. hledáním lepších hodnot tzv. hyperparametrů nebo přidáním dalších vrstev. Nejvyšší přesnost byla dosažena s modelem SVM, a to 90 %. Dle srovnání s literaturou se jedná o úspěšný výsledek.

Electroencephalogram (EEG) and machine learning-based classification of various stages of mental stress

Declaration

I hereby declare that this Bachelor's thesis was prepared as an original work by the author under the supervision of Mr. Muhammad Asad Zaheer. I have listed all the literary sources, publications and other sources, which were used during the preparation of this thesis.

. . . .

Tereza Lapčíková

May 16, 2024

Acknowledgements

Contents

Chapter 1

Introduction

In recent years, we have witnessed a great increase in mental health issues among the whole population, specifically among young people. It is proven that chronic stress has a very negative influence on both the mental and physical health of the individual. It may even be one of the triggering factors of anxiety and depression and can deepen mental health issues that have already developed. Stress itself is a group of defense mechanisms of an organism that occur in response to the perception of physical or mental stressors. This state is connected with severe unpleasant physiological symptoms. Stress can be divided into two groups – acute stress and chronic stress. Acute stress is, at its core, a crucial evolutionary ability of the human body for survival and adaptation. Chronic stress corresponds to a consistent feeling of pressure over a long period of time [9] and is the main factor negatively influencing health. By early detection of chronic stress and subsequent resolving its mitigation, we could prevent a range of mental health issues from burnout syndrome to depression. This is where a method called Electroencephalography, abbreviated into EEG, comes into account. EEG is a non-invasive and financially available method suitable for exploring brain activity through the so-called brain waves. Brain waves are electrical impulses that can be detected on the scalp, and it has been proven that several processes of the brain, including stress, can be analyzed through them. EEG is used in both scientific and medical fields and is suitable for the detection of both mental and physiological processes, like blinking, heart rate, etc. The effect of stress on the human brain is not yet fully understood. EEG, in connection with a suitable machine learning or deep learning model, provides a useful method that might contribute to a deeper understanding [65].

This thesis primarily aims to create a machine learning model that classifies a subject's mental stress stages based on EEG data. It also deals with topics relevant to the creation process of machine learning or deep learning models and important facts to understanding stress response itself. This thesis starts by delivering essential information about the architecture of the human brain, which is described in section 2.1. Section 2.2 focuses on describing the stress response mechanism and its phases. A brief introduction to EEG is provided in section 2.3. The next important topic studied in the chapter 2.4 is the analysis of features, including classification and validation. Finally, section 2.5 describes the current state of research focused on this topic.

Chapter 2

Literature Review

This chapter explains the key aspects needed to comprehend the studied topic. The aim of this chapter is, firstly, to explain and discuss the anatomy of the human brain, the mechanism of stress, and how it manifests in the brain. Secondly, this chapter introduces the diagnostic method known as electroencephalography (EEG), which is nowadays widely used and researched for both scientific and medical purposes. This thesis aims primarily to create a machine learning model for stress stage recognition from EEG data, so the next and the most important topic to be discussed in this chapter is feature analysis and its individual sub-parts such as feature extraction, selection, and classification using machine learning and deep learning techniques. Finally, this thesis could not have been conducted without a proper understanding of the current state of research on this topic, which is therefore also considered in this chapter.

2.1   Human Brain Architecture

Not only is the brain the most complex part of the human body, but it is also the most complex functioning system that is known. It is a complex network of neurons (neuronal cells) and other cell types that controls the nervous system coordinates the human body, controls behavior, controls the internal state of the human body, and processes sensory information from the surrounding environment. The brain controls these activities through thinking and cognition [6, 61].

The human brain is subdivided into many regions, and each of them is highly specialized for certain functions. The main three brain regions that the brain is divided into are the cerebrum, cerebellum, and brain stem [6, 61].

2.1.1   Cerebrum

The cerebrum is the largest unit of the human brain, located at the front and the top of the brain. Two basic parts that the cerebrum is divided into are the cerebral cortex, otherwise known as gray matter, which is laminated, and the cerebral nuclei, white matter, which is mostly nonlaminated. It is a brain region responsible for a wide range of functionalities, e.g., movement coordination, conscious thoughts, judgment, speech, emotions, learning, problem-solving, sensory perception, and management, behavior, and personality, etc. [6, 61].
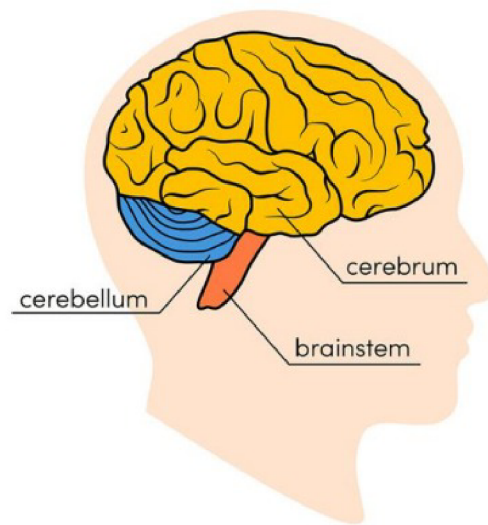
Figure 2.1: Three main regions of the human brain – cerebrum, cerebellum, and brain stem [6].

2.1.2 Cerebellum

The cerebellum is located at the back of the brain and makes up around ten percent of its total weight. Thanks to the dense arrangement of the cerebellar cortex granule cells, it also has the highest concentration of neurons out of all the brain regions.

The outer part contains neurons while the inner area communicates with the cerebral cortex. Functions such as voluntary muscle movement, coordination, maintaining posture, balance, and body equilibrium are provided by this region. According to the latest studies, the cerebellum might be involved in thinking, emotions, and social behavior as well as in addictions, autism, and schizophrenia [6, 27, 61].

The cerebrum and cerebellum are divided into halves collectively referred to as the cerebral hemispheres. The left and right hemispheres are divided by a deep median longitudinal fissure. The bundle of fibers located in the depth of this fissure called the Corpus Callosum, forms a connection between the hemispheres, which provides signal transmission between both sides. The left hemisphere controls the right part of the body and is connected with creativity, emotions, intuition, art, etc. The right hemisphere, on the contrary, controls the left part of the body and performs tasks like maths, reasoning, logic, etc. [6, 61].

Both hemispheres can be subdivided into Frontal, Parietal, Temporal, Occipital, Insular, and Limbic lobes. Most of the brain functions are performed in cooperation of several brain regions but certain functions are provided only by a specific lobe by itself. Tasks performed by the individual lobes are described below [6, 61].

- **Frontal Lobe**
  As the name suggests, this lobe is located behind the forehead, and it is connected with higher executive functions, e.g., thinking, problem-solving, speech, emotional regulation, voluntary movement, etc.

- **Parietal Lobe**
  The parietal lobe is located behind the frontal lobe on the posterior top of the head. These areas are responsible for spatial orientation and integrating sensory information such as touch, pressure, pain, temperature, and texture.

- **Temporal Lobe**
  The temporal lobes lie on both sides of the head above the ears. These brain regions are involved in processing sensory information, particularly hearing and speech perception, as well as memory, learning, and emotions. The temporal lobes also include areas responsible for visual processing.

- **Occipital Lobe**
  The occipital lobe is found at the back of the head and is often called the visual cortex. As the name suggests, the occipital lobe is responsible for vision and visual processes such as reading. We can consider it the major visual processing center in the brain.

- **Insular Lobe**
  The insular lobe lies deep within the brain as the part of the cerebrum above the brain stem. It is involved in consciousness, regulation of the body's homeostasis, as well as in some of the emotions, pain experience, social emotions, emotional intelligence, and multimodal sensory processing, e.g., during the combination of auditory and visual tasks.

- **Limbic Lobe**
  The limbic lobe is the same as the insular lobe, located deep within the brain, beneath the cerebrum, and above the brain stem. This system is involved in motivationally driven and emotional behaviors, social skills, empathy, memory, learning, homeostatic responses, etc. [6, 38, 8, 61].

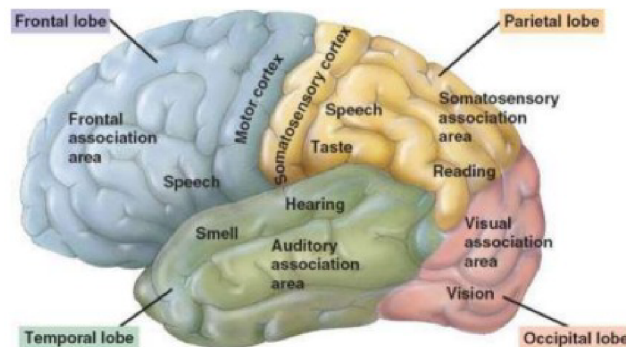Figure 2.2: Different lobes of the human brain and their functionality [6].

2.1.3 Brainstem

The brainstem is located in the middle of the brain and connects the cerebrum with the spinal cord. The spinal cord, supported by the vertebrae, is the main path carrying messages between the brain and the rest of the body. Processes like breathing, heart rate, balance, swallowing, and more are regulated by the brainstem [6].

2.2 Stress

Stress is a natural reaction of the human body to a stressor. This beneficial protective mechanism helps our organisms invoke fast reactions to changes and potential environmental and other external threatening conditions. As long as there are various types of stressors, stress can be manifested by a whole range of symptoms. Psychological stress is mainly considered in this thesis. As mentioned before, stress is beneficial and essential in a whole range of situations. However, the excessive amount of stress that a large number of the working-age population faces today can easily develop into chronic stress that can be critical for both body and mind and can be involved in several health issues. Beyond the health problems connected with chronic stress, it is also manifested by unpleasant feelings of pressure, mental overload, several aches, insomnia, weakness, distraction, and more. According to studies, stress is one of the major factors contributing to chronic disorders. Therefore it is important to take care of stress prevention, which can be achieved by a healthy lifestyle. Early diagnosis of chronic stress may, in some cases, be a key to mental and other illness prevention [9, 65].

According to Vanhollebeke et al., [65], a stress response can be divided into three stress phases – anticipatory, reactive, and recovery. The anticipatory phase begins the moment when a person is aware of the stressor but is not yet directly exposed to it. During the reactive phase, the subject is directly exposed to the stressor. The recovery phase, as the name suggests, starts directly after the end of the stressor exposure and consists of the reversal of psychological and physiological alterations caused by the stressor [65].
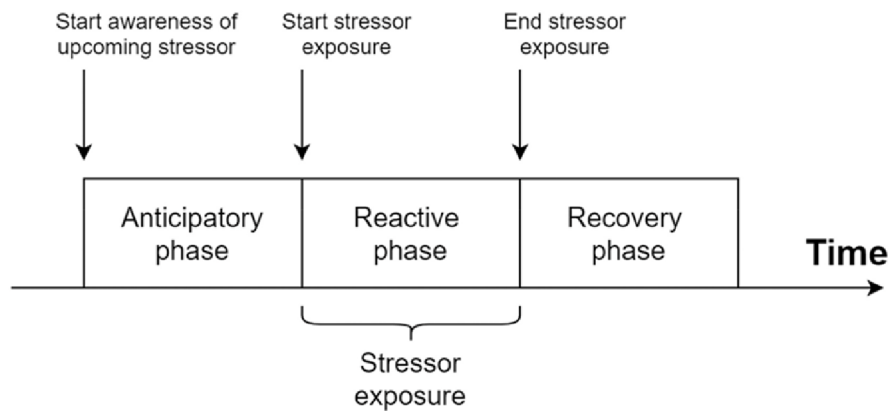
Figure 2.3: Three phases of the stress response [65].

2.2.1 Mechanism of a Stress Response

The first detection of a stressor in the brain occurs in regions called the amygdala (temporal lobe) and prefrontal cortex (frontal lobe). The amygdala sends distress signals to release the stress hormones from the adrenal glands. Hormones adrenaline and noradrenaline activate the sympathetic nervous system (SNS), which manifests itself in an increased heart rate, dilating of airways, redirecting blood flow to vital organs, increased blood flow to the muscles and other physiological mechanisms to prepare the body for immediate action known as the „flight or fight" response. The hypothalamus, a brain region located

6

in the temporal lobe, simultaneously produces corticotropin-releasing hormone (CRH), which activates the hypothalamic-pituitary-adrenal axis (HPA). The HPA axis then signals the release of adrenocorticotropic hormone (ACTH) from the pituitary gland, whose presence stimulates the release of cortisol from the adrenal glands. Cortisol is another important stress hormone, whose task is to mobilize the energy to cope with the stressor. This is achieved by an increase in the body's glucose metabolism. In addition, cortisol simultaneously affects brain functions by influencing emotional regulation, memory, and attention. High cortisol levels in the blood are a signal to the hypothalamus and pituitary glands for a negative feedback loop. This loop inhibits further release of CRH and ACTH [15, 41, 55, 63, 57].

In a summary, the stress response mechanism is a complicated process that allows the organism to react quickly to the presence of a stressor, a potential danger. Stress response typically consists of three phases: the anticipatory phase, the reactive phase, and the recovery phase.

2.3 Electroencephalography (EEG)

The history of electroencephalography dates back to the last century. Hans Berger, the German psychiatrist, did the first recording of the human brain's electric field in the year 1924. Since then, the EEG has been used as a useful method to understand and diagnose mental and neurological disorders and to assess cognitive processes, e.g., memory and perception. This neuroimaging technique consists of measurement of the brain's response to a stimulus (of a sensory, motor, or cognitive nature), so-called Event-Related Potential (ERP), and is widely used in cognitive neuroscience research. The advantage of using the EEG in comparison with other methods used in this scientific field is its good temporal resolution and low price. The disadvantage of this method is the low spatial resolution of the brain activity. EEG can also be integrated with other neuroimaging methods, such as magnetic resonance imaging (MRI), functional near-infrared spectroscopy (fNIRS), and positron emission tomography (PET) to obtain more comprehensive results [45].

2.3.1 EEG Signals

EEG is a non-invasive method studying amplified electrical signals caused by the synchronous activity of the sum of thousands of millions of brain cells, i.e., neurons with a similar spatial orientation measured with electrodes placed on a subject's scalp. Such a recording of neural oscillations is called the electroencephalogram and can be characterized by the frequency, amplitude, and phase of the oscillation. EEG signals are typical for their complexity, nonlinearity, and lack of conformity to a normal distribution. At the same time, the signal waveform can be significantly influenced by individual factors, e.g., age, health, testing environment, etc. Measured data usually contain unwanted signals caused, e.g., by eye and muscle movement called artifacts. Obtaining information from such complex signals can be very challenging [45].

The main objects of study are five major brain waves varying in their frequency ranges:

Delta Waves

Delta waves are neural oscillations with the highest amplitude and the lowest frequency within the range of 0.5 to 4.0 Hz. Delta wave activity is associated with the NREM sleep phase (deep sleep) [45].
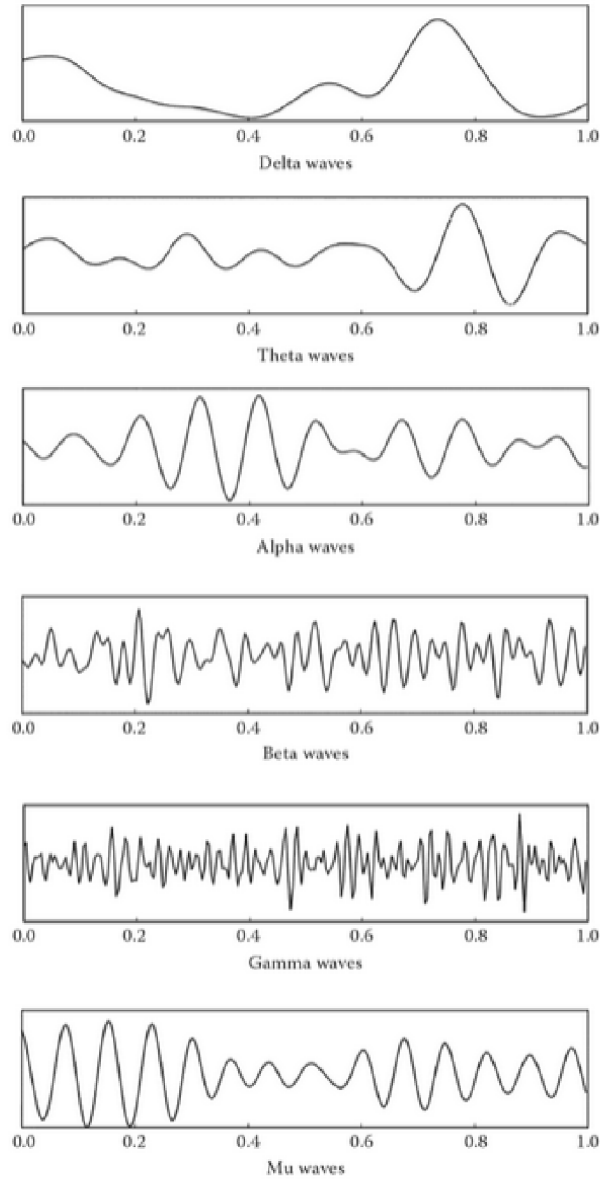

Figure 2.4: Major brain waves varying in their frequency range [45].

Theta Waves

The frequency range of theta waves is 4 to 8 Hz. Among adults, theta waves are associated with states of drowsiness, arousal, or meditation [45].

Alpha Waves

Alpha waves lie within the frequency range of 8 to 13 Hz and are correlated with wakefulness, effortlessness, alertness, and creativity. Alpha waves typically appear in the posterior half of the head [45] According to studied literature, alpha power tends to decrease due to the influence of stress [67, 53, 52].

Beta Waves

These are neural oscillations with a frequency range of 14 to 26 Hz. Beta wave activity is related to active attention and thinking, critical problem solving, and focusing on the outside world. This type of brain wave typically appears in the frontal and central region of the brain [45] According to the studied literature, beta power tends to decrease its value due to the influence of stress [67, 53, 52].

Gamma Waves

The frequency range of gamma waves is 30 to 100 Hz. The appearance of gamma waves in the human brain is rare and is related to the process of combining different senses (sound and sight) [45].

As long as EEG reflects thousands of ongoing neural processes, it is challenging to isolate, assess, and analyze them. Specific information related to sensory, cognitive, and motor events can be extracted using ERPs by repeating an event (e.g., visual stimulus) and its EEG activity analysis [45].

2.4 Feature Analysis

The raw output of an EEG measurement is a signal recording changes in electrical potential on the experiment subject's scalp. To obtain information, it is necessary to analyze measured signals through their features. The methodology of EEG signal analysis can be summarized in the so-called machine learning pipeline. The machine learning pipeline is a series of steps to automate, standardize, and streamline machine learning model building, training, evaluating, and deploying. In the case of EEG analysis, a machine learning pipeline typically consists of data acquisition, pre-processing (denoising), feature selection, and classification [68, 59].

2.4.1 Time-Domain Feature Extraction

In the case of the time-domain feature extraction, the measured signals are analyzed with respect to time. These techniques provide quantification of signal changes over time. Because of its length, the time-domain signal typically requires segmentation and windowing providing extraction of local features [59].

Time domain analysis includes statistical features such as mean, median, variance, standard deviation, skewness, and kurtosis. Another simple feature is the so-called zero crossing rate, which provides information about how often the signal crosses the horizontal axis. In some cases, it can be appropriate to count separately the number of „positive-going" crossings, which are the crossings of the horizontal axis from negative to zero to positive, and the number of opposite „negative-going" crossings [59].

Entropies

Entropies are time-domain features based on the quantification of time series regularity that express well the complexity of the signal. Entropies are widely used in this scientific field because they allow successful feature extraction even with noisy and short recordings. Extensively known are approximate entropy algorithm (AE) and sample entropy algorithm (SE). These two algorithms determine the regularity of time series for similar epochs based on the existence of patterns. The greater the data irregularity in the sequence is, the greater the non-negative number is assigned to the sequence. [67, 14, 47].

Hjorth Parameters

The Hjorth Complexity describes deviance from the course of the sinus. The higher the value of Hjorth Complexity is, the more complex the examined signal is. Hjorth Complexity is calculated using three parameters – activity, mobility, and complexity. The activity is derived as the variance of the EEG signal. The mobility is equal to the square root of the variance of the first derivative of the EEG signal. The complexity parameter is equal to the ratio of the mobility to the square root of the activity [46, 56].

Hurst Exponent

The Hurst Exponent is a time-domain feature of the trend persistence or the randomness. It is calculated using the R/S method. First, the signal is divided into windows of different sizes. Then, the average and the standard deviation of the signal are calculated for each window. The rescaled range R/S is calculated for each time window and it is defined as the ratio of the difference between maximum and minimum values of the signal and the standard deviation. Subsequently, the logarithm of the R/S is against the logarithm of window size. The resulting Hurst exponent is equal to the slope of the linear regression plot. The value of the Hurst Exponent can range between 0 to 1. The value of the Hurst Exponent under 0.5 indicates that the signal tends to reverse its direction. On the contrary, values above 0.5 indicate persistent behavior. Hurst Exponent that is equal to 0.5 suggests an unpredictable course of the signal [64].

2.4.2 Frequency-Domain Feature Extraction

Five main brain waves that occur in the human brain and their frequencies were mentioned in 2.3.1. For identification and classification of the brainwaves, it is necessary to analyze the frequency spectrum of the recording [65]. Brain waves and their frequency bands are summarized in table 2.1.

Wave	Frequency band (Hz)
Delta	$0.5 - 4.0$
Theta	$4.0 - 8.0$
Alpha	$8.0 - 13.0$
Beta	$13.0 - 26.0$
Gamma	$30.0 - 100.0$

Table 2.1: Brain waves and their frequency bands [57].

Power Spectral Density

A fundamental concept in frequency-domain analysis is the Power Spectral Density (PSD). The PSD calculates the distribution of a signal's power as a function of frequency. It can be calculated, e.g., using the Fast Fourier Transform (FFT) or Welch's method. Welch's method divides the signal into overlapping segments and, for each of them, computes the squared root of the FFT. The resulting numbers are an estimation of the spectral density, and their averaging is provided, resulting in PSD estimation. Welch's method provides a smoother frequency-domain spectrum than FFT [54, 57, 65].

Asymmetry

Another important concept in frequency-domain feature extraction is asymmetry. It is a relative measure of the difference in band powers between the right and left hemispheres in a specified brain area, such as frontal asymmetry or cerebral asymmetry [5].

2.4.3 Connectivity Analysis

The processing of information in the brain is spatially distributed and involves both local and distant brain regions. These regions are functionally connected and synchronized near optimal. Analysis of brain connectivity can be useful for a deeper understanding of studied processes and involved brain regions. Brain connectivity analysis focuses on three different but related forms of connectivity – anatomical connectivity, functional connectivity, and effective connectivity. Anatomical connectivity examines the connectomes, connecting neuron pools in spatially distant brain regions connected through synaptic contacts between neighboring neurons or fiber tracks. Functional connectivity is described as the temporal dependency of neuronal activation patterns of anatomically separated brain regions. This statistical concept relies on statistical measures like correlation, coherence, phase locking, etc. Effective connectivity describes how one neuronal system affects another [36].

2.4.4 Feature Selection

Feature selection is a process of selecting relevant features that will act as input to a machine learning model. In this process, the amount of noise in the data is reduced. In other words, it is a method of reducing input data to its relevant components. This step is essential for training a functional and accurate model. Feature selection ensures, that a machine learning model does not capture the unimportant patterns and does not learn from noise. Both supervised and unsupervised models are used for feature selection [22].

Principal Component Analysis (PCA)

PCA is a statistical procedure that reduces the dataset to its essential features by identifying its directions, in this case called principal components. The principal components are a set of new uncorrelated variables that capture the maximum possible variability in the data. It is based on a specifically chosen linear transformation that maximizes the variance when projected onto the new axes. The transformation projects the original data onto a lower-dimensional subspace by scaling and rotating the original feature space. The vectors of the features are projected on the transformed subspace in relevant directions [58, 7].

Pearson Correlation Coefficient (PCC)

Analysis of a relationship between two variables can be done by calculating the Pearson Correlation Coefficient. Key aspects for determining a casual linear relationship between variables are the correlation's strength and the correlation's direction. The value of the Pearson Correlation Coefficient can range from minus one to plus one. Pearson Correlation Coefficient value of zero is characteristic of no correlation between variables. On the contrary, a value close to plus one is characteristic of a strong positive correlation, and a value close to minus one for a strong negative correlation [48].

Variance

Variance is a statistical measure reflecting the dispersion of the data. In other words, it is an indicator of how much the data is scattered around the mean. Variance is calculated as the sum of the squared average distance from the mean divided by the count of analyzed values. The main idea behind variance is to obtain information about how big the spread of values from the average is, not the deviation of the individual. It is a useful tool for the selection of features from the data. A threshold value is selected and all features with variance lower than the defined threshold are removed [22].

Genetic Algorithms (GA)

These efficient techniques of optimization are based on natural evolutionary theory. Major steps such as selection, crossover, mutation, and replacement are used. The principle of these algorithms is based on the successive creation of generations, where each new generation generates a different solution to the optimization problem. The solution to the problem is improving with the evolving population. The first generation consists of randomly generated individuals. The transformation from the first generation to the following generations is achieved by iterative repeating of stochastic reproduction of the individuals into modified versions via crossover and mutation. Each iteration results in a greater homogenization of the population concerning the survivability of the individuals who adapted to their environment [58].

2.4.5 Feature Classification

The following section describes selected machine learning and deep learning algorithms widely used for feature classification. Classifiers typically use supervised, unsupervised, or reinforcement learning algorithms. These algorithms control the learning process by external configurations, so-called hyperparameters. The value of the hyperparameters cannot be estimated from the data and has to be specified beforehand. The process of the determination of the suitable hyperparameters can be greatly simplified by validation. The principle of selected validation methods is described in 2.4.6 [33].

K-Nearest Neighbors (K-NN)

This supervised learning algorithm is one of the most popular schemes used for classification tasks thanks to its simplicity and computational efficiency. First, the distance between a given point, and its neighboring points in the training dataset is calculated. The distance between the given point and its neighbor is usually calculated using the Euclidean

distance, but different approaches can be applied as well. The obtained values are subsequently sorted in increasing order and the *K*-nearest neighbors are selected for further use. The classification of the given point is then determined based on the vote of its neighbors. The given point is predicted to be of the class, that is the most common among the selected *K*-nearest neighbors. Choosing the correct value of the *K* hyperparameter is a key step to avoid overfitting or underfitting the model to the training dataset [3].

Support Vector Machine (SVM)

First, it is appropriate to describe the term margin. When analyzing data with two distinct classes, we can determine a threshold, that lies between the closest data points belonging to different classes. The margin is the distance between the nearest data point of a given class and the threshold. The largest possible margin is selected for use in the SVM. This type of classifier is called the maximum margin classifier. Unfortunately, a great disadvantage of this classifier is a high sensitivity to outliers in the training data. The solution to this problem is to allow misclassifications in the training data. In this context, the term soft margin refers to the distance between the nearest data point and the threshold. The number of allowed misclassifications is a hyperparameter, that can be optimized through cross-validation. The support vector classifier uses the soft margin to determine the location of a threshold. The observations on the edge and within the soft margin are called support vectors. In the context of the observations, a threshold is represented as a point in a one-dimensional space, as a line in a two-dimensional space, as a plane in a three-dimensional space, and so forth. In summary, the SVM is a set of supervised learning algorithms, that view data points as *p*-dimensional vectors. These algorithms move the data into a higher dimension and find a support vector classifier. In other words, these algorithms try to separate data points from different classes by (*p*-1)-dimensional hyperplane. The kernel function enables one to do a computation in the higher dimension without a transformation. This so-called kernel trick significantly reduces the computational demands of the SVM. The radial kernel function computes the interactions between each pair of data points within the specified dimension. The radial basis function kernel enables the discovery of support vector machines in theoretically infinite dimensions. Beyond the polynomial and radial function kernels, there are various other kernel functions, that can be used depending on the data and the task [13, 50].
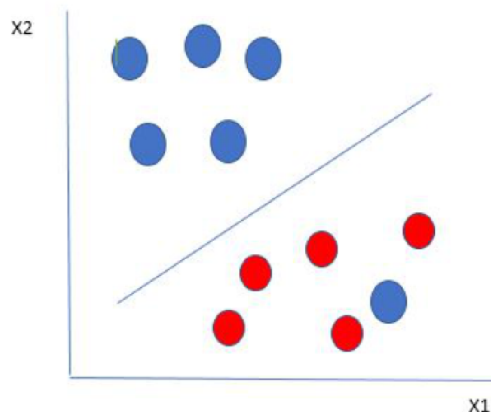

Figure 2.5: The most optimized hyperplane for the illustrated case of classification [21].

Random Forest

The Random Forest is a powerful machine learning algorithm, that creates class predictions using a collection of decision trees. The single decision tree involved in the classification task results in an overfitting model, in other words, a model, that is highly sensitive to the training data. This problem can be avoided by using the collection of multiple random decision trees, the so-called random forest. In the initial step known as bootstrapping, the new collection of datasets is created by selecting random rows from the original dataset. It is important to mention, that the algorithm does not train each decision tree with every feature. The process of training is only done with a random subset of features. This approach reduces the correlation between the trees and prevents the increase in variance. The Random Forest makes predictions by passing the data point through each decision tree and noting the predictions individually. The resulting class is decided by aggregating the predictions of the decision trees and choosing the class that receives the majority vote. This process is called aggregation [39, 24].

Convolutional Neural Networks (CNN)

Convolutional Neural Networks, a specialized variant of feed-forward artificial neural networks, are a very popular tool in image analysis. However, their effectiveness is also appreciated in other data analyses and classification problems. Convolutional Neural Networks are specialized in pattern recognition. Compared with Multilayer Perceptron, the CNN detects patterns using hidden layers otherwise called convolutional layers. Such a network usually contains other non-convolutional layers and necessarily contains a non-linear activation function. Each convolutional layer receives input, transforms it with a convolutional operation, and passes the transformed input to the next layer. A convolutional layer is composed of a specified number of filters. A filter can be conceptualized as a relatively small matrix. The size of this matrix is a hyperparameter. The values within this matrix are initialized with random numbers. The filter convolves across each input element, resulting in a matrix dot product. This new matrix of dot products is the output of the convolutional layer. As we delve deeper into the network, the filters become more complex, enabling the extraction of more sophisticated features from the input [23, 28].

Recurrent Neural Networks (RNN)

Recurrent Neural Networks stand apart from other neural networks due to their application of feedback loops, in addition to weights, biases, hidden layers, and activation functions. These feedback loops enable predictions with varying input values, even when dealing with sequential data, such as time-series data [20].

In contrary with traditional neural networks, where the inputs and outputs of layers are independent, the RNN feedback loops allow to influence the current step input by the output from the previous step. A key feature of this network is a Hidden state, also known as the Memory state. As the name suggests, this feature enables the RNN to retain information about the previous sequence while processing inputs, a desirable property for tasks such as predicting the next word in a sentence. This unique capability is provided by a processing unit called the Recurrent Unit [20].

The tasks performed to produce the outputs on hidden layers are identical across all units and the same parameters are passed for each input, reducing the complexity of the parameters. In an RNN, the weights are shared across all time steps in the network. Each
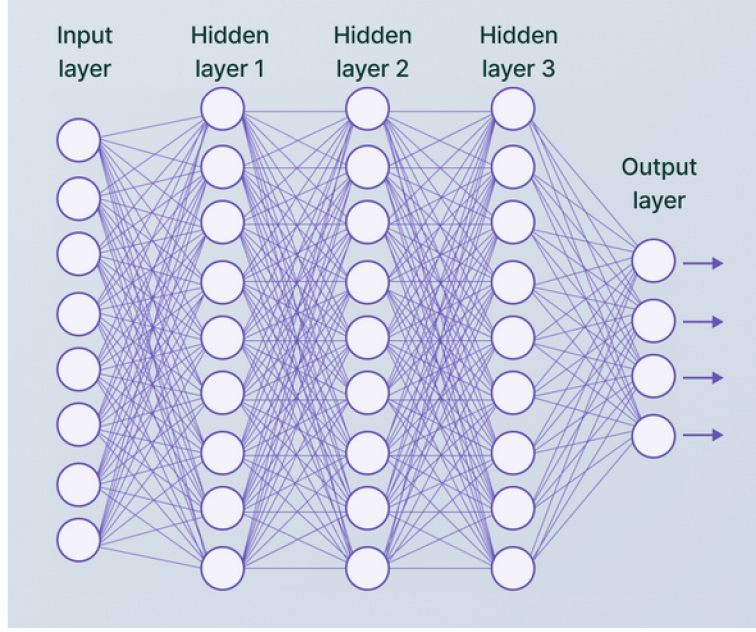
Figure 2.6: The illustration of convolutional neural network with three hidden layers [4].

recurrent unit of the RNN is associated with a specific time step and has a fixed activation function. The hidden state of the unit, representing accumulated information from the past, is updated at each time step to reflect alternations in the network's understanding of past data [20].

The current state is calculated using the following formula

$$h_t = f(h_t, x_t), \tag{2.1}$$

where h_t represents the current state, h_{t-1} the previous state and x_t the input state. The activation function is applied to the current state with the formula

$$h_t = tanh(W_{hh}h_{t-1} + W_{xh}x_t), \tag{2.2}$$

where w_{hh} is weight at the recurrent neuron and w_{xh} is weight at the input neuron. The output is subsequently computed with the formula

$$y_t = W_{hy}h_t, \tag{2.3}$$

where y_t is the output and W_hy is the weight at output layer. These parameters are updated using backpropagation through time [20].

Long Short-Term Memory (LSTM)

LSTM is a neural network built on RNN. In comparison with RNN, LSTM is extended with a memory cell, which is a container holding information about long-term dependencies. This ability makes it well-suited for tasks like speech recognition, language translation, and time series forecasting. Deep LSTM networks can learn very complex patterns and provide reliable results, but they do so at the cost of complicated and time-consuming tuning of the training process, which is in part caused by a large number of hyperparameters [17].

The operation of the memory cell is controlled by three gates – the input gate, the forget gate, and the output gate. The chain of these three gates allows LSTM to selectively retain or discard information [17].
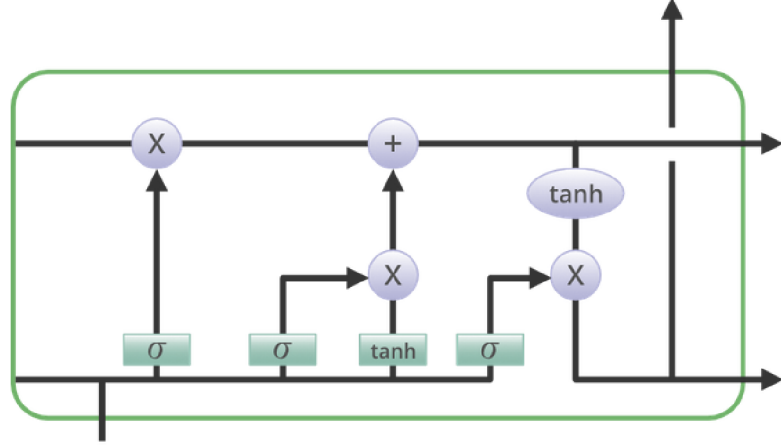

Figure 2.7: The illustration of the LSTM memory cell – from left to right is the forget gate, the input gate, and the output gate [17].

The forget gate removes information, that is no longer useful for the cell. This process is described by the following equation

$$f_t = (W_f[h_{t-1}, x_t] + b_f),\tag{2.4}$$

where W_f represents the weight matrix of the forget gate, x_t is the input at the particular time, $h_t - 1$ is the previous cell output, b_f is the bias with the forget gate, and the *sigma* parameter is the sigmoid activation function. The two inputs of the gate are multiplied with weight matrices. The result is then summed with the bias and passed through an activation function. The activation gives a binary output. If the output is zero, it means that the information about the particular cell state is no longer useful and, therefore, has been removed. For the output one, the information is retained [17].

The addition of new information to be retained by the cell is provided by the input gate. The operation performed in the input gate is illustrated with the following equations [17].

$$i_t = (W_i[h_{t-1}, x_t] + b_i)\tag{2.5}$$

$$\hat{C}_t = tanh(W_c[h_{t-1}, x_t] + b_c)\tag{2.6}$$

Parameters x_t and h_{t-1} represent the input at the particular time and the previous cell output, similarly to the forget gate. The b_c parameter represents the bias with the input gate. First, the information is regulated using the sigmoid function – the values to be remembered are filtered in a similar way to the forget gate. Second, the *tanh* activation function is applied in order to create a vector. The output value is in the range from minus one to plus one and contains all positive values from the x_t and h_{t-1}. Then, the useful information is obtained by multiplication of the vector values and the regulated values. The prior state is multiplied by f_t, while the information we previously decided to overlook

16

is not considered. Following that, the expression $i_t * C_t$ is incorporated. This expression signifies the revised candidate values, modified according to the degree to which we decided to update each state value [17].

$$C_t = f_t C_{t-1} + i_t \hat{C}_t \tag{2.7}$$

The last gate is the output gate, which determines what information from the current cell state is going to be presented as the output. First, the *tanh* activation function is applied to the cell input with the purpose of creating the vector. The input information is then regulated to values between zero and one using the sigmoid activation function. The output from the sigmoid function is then used to filter the values that need to be remembered. Finally, the regulated values and the vector values are multiplied and sent as an output of the output gate. This process can be described by the following equation

$$o_t = (W_o[h_{t-1}, x_t] + b_o), \tag{2.8}$$

o_t is the output, W_t is the weight matrix associated with the output gate, h_{t-1} is the previous cell output, x_t is the input at the particular time and b_o is the bias with the output gate [17].

There is a variation of LSTM, so-called Bidirectional LSTM (BLSTM), that processes input sequential data in both forward and backward directions – one LSTM layer does so in the forward direction, the other in the backward direction. This feature enables the attainment of even better results compared to basic forward LSTM [17].

2.4.6 Validation

Machine learning and deep learning algorithms learn to make predictions during a process called training. The model learns to recognize important patterns in the training dataset. The learning process, however, is not a straightforward task. There are several steps, that have to be considered. The training process requires high-quality training data. The noise, missing values, and incorrect labels can lead to poor performance because the model can learn incorrect patterns. Choosing the right machine learning model itself can also influence the performance. Finding a suitable model for the task and given data is a complex process. Next, choosing an irrelevant or redundant feature as the model input can negatively impact its performance. Another problem occurs when the model learns the training dataset too well, so-called overfitting. Underfitting is the contrary situation when the complexity of the model is too low to capture the structure of the data. Finally, very time consuming can be fine-tuning of the model's hyperparameters. Hyperparameters define higher-level model structures and have to be set before the training process starts. Validation is useful here to detect potential problems, especially overfitting. The following lines discuss selected validation strategies [19].

Holdout Validation

The dataset is divided into two parts—the training set and the testing set. The training set is used to build the model. The test set checks the model's ability to generalize the new, unseen data by calculating selected validation metrics, in other words, how the model will be applicable in the real world. Unfortunately, this approach can lead to unstable estimates. If the split does not accurately reflect the overall data distribution, it may result in inconsistent estimations [19].

K-Fold Cross-Validation

Another validation method, that may be used, is the K-Fold Cross-Validation. This technique is very popular because it ensures thorough learning across the entire dataset. Initially, the dataset is partitioned into K parts. The K-1 parts of the dataset are used for the training, while the remaining part utilizes the testing process. The procedure of training and testing is iterated until each part of the dataset has been used for the testing [19].
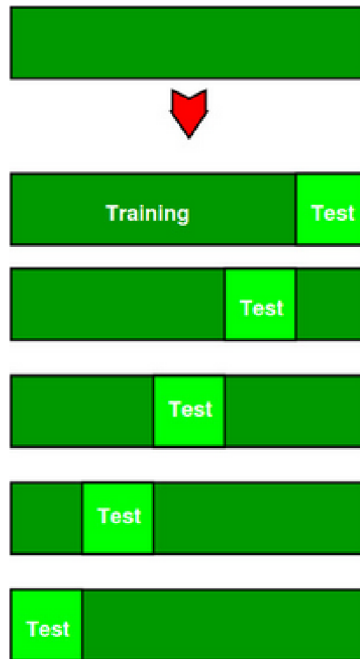

Figure 2.8: Division of the dataset into the training and testing part within the 5 folds [19].

Stratified K-Fold Cross-Validation

The main idea of this validation method is to divide the dataset into K folds, where each fold contains a representative proportion of each class. This ensures accurate learning results across all classes even though the layout of classes in the dataset is imbalanced. The input data is first shuffled and then partitioned into sub-parts. Each sub-part is then used for training. The step ensures the shuffle is executed only once [19].

Leave-One-Out Cross-Validation (LOOCV)

As well as the previous methods, the LOOCV divides the entire dataset into folds. In this case, every individual data point is treated as a separate test dataset, while the rest of the dataset is used to train the model. Although LOOCV offers the most precise performance evaluation from the described methods, in the case of large datasets, it can be computationally expensive [19].

Time Series Cross-Validation

As the name suggests, this cross-validation method is specialized for validation in time series. The dataset is split into overlapping windows. The model undergoes training on the training part of the dataset and then proceeds to evaluation on the subsequent testing segment. It is important to emphasize that the dataset's training set has to be in position before the testing set because future data should not be used for training. After these two steps, the window of the training set is expanded forward by a time step. The earliest step of the format training data is dropped, and the validation set is expanded forward by a one-time step. These steps are then repeated until the entire dataset is incorporated. This approach's great advantage is that it considers the inherent time dependencies in the data series, which can increase the model's ability to predict future data [19].

Validation Metrics

The choice of validation method is closely linked to the selection of the right metrics to evaluate model performance. The following lines describe only selected metrics used for evaluation of the model implemented in this thesis, so-called classification-specific metrics.

- **Accuracy**
  This metric provides information about the proportion between the number of correct predicted classes and the total number of predictions [1].

- **Precision**
  Precision is a metric, that quantifies how accurate the model is when predicting the target class. More precisely it is equal to a ratio of true positive predictions to the total number of positive predictions (both true and false positive predictions). In this context, it is important to mention that a true positive prediction considers a correct prediction of a given class, and a false positive prediction is an instance that was incorrectly predicted as positive of a given class [1].

- **Recall**
  Another useful metric used for learning model validation is recall. This metric quantifies correct identifications of the true positives in ratio to all true positives and false negatives in the dataset. In this case, false negative means the missed case of positive. In other words, recall provides information on how often the model correctly identifies positive instances from all the positive samples, that are present in the dataset [1].

- **F1-Score**
  F1-score is an evaluation metric that combines the model's precision and recall scores. However, this metric can be unviable in datasets with imbalanced classes. The F1-score value equals the harmonic mean of the precision and recall scores. The higher the value of the F1-score is, the classifier is of the better quality [18].

2.5 Current State of Research

In recent years we have seen a rapid increase in mental health issues. Although the modern way of life brings many advantages and living is easier than it used to be, there are some factors that can poorly influence an individual's well-being, e.g., excessive consumption of social media, bad work environment, loneliness, economic uncertainty, and above

all, an increase of individual's psychosocial stress. Stress is a factor that can significantly catalyze the development of mental health issues. However, the influence of psychosocial stress on brain activity is not yet completely understood. This is the point, where the electroencephalography (EEG) comes into play. It is a non-invasive method that can be helpful in stress diagnosis. Data obtained from stress-related EEG experiments can greatly contribute to a deeper understanding of stress. This chapter summarizes the findings of selected EEG stress-related studies [65].

Certain articles focus on stress response in its entirety, but some articles investigate stress as a process that can be divided into three discrete phases – the anticipatory, the reactive, and the recovery phase. Physiological processes, that are involved in stress response and three discrete stress phases were described in section 2.2. The strength of the EEG is its high temporal resolution, and therefore, most stress-related studies focus on the investigation of event-related potentials (ERPs). A whole range of EEG stress-related articles also focus on the analysis of changes in signal oscillations through spectral analysis. ERP analysis is not very suitable for the investigation of the whole neural response to stress because ERP recordings are usually dependent on clearly defined stimuli, and the recovery phase is usually not recorded at all. In this case, the spectral analysis yields better results, because both stimulus-defined time windows and long continuous EEG recordings can be performed [65].

According to studied articles, some EEG measures report stress phase-dependent behavior. On the other hand there are measures that are entirely stress phase independent. According to the literature, stress response typically manifests in a significantly decreasing trend of the alpha band power in the frontal lobe. As long as the alpha waves are connected with a state of calmness and relaxation, the decreasing trend confirms that regardless of the stress response phase, the subject is not experiencing comfort. Most of the studied literature recorded a significant increase in the beta band power due to experienced psychosocial stress. These changes occurred mainly in the frontal lobe. On the other hand, two articles dealing with this measure report an insignificant decrease. Measures such as delta and theta band power, relative gamma, and theta-alpha band power ratio report stress phase dependency. However, deeper investigation of the stress phase dependence of measures has many pitfalls as long as some pairs formed by measure and stress phase are overrepresented in studies and others are completely absent [65].

According to the literature, the reactive stress phase is associated with an increase in the delta band power [40]. On the other hand, during the recovery phase, when the body starts to reverse the alterations caused by the stressor, the delta band power decreases [49]. Additionally, two articles describe a significant increase in theta band power during the reactive phase [35, 66]. However, another article describes a decrease in theta band power in the recovery phase [40]. It is also noted, that theta-alpha power is significantly higher during the reactive phase than during the recovery phase [29]. Interestingly, relative gamma has a completely opposite flow. The reactive phase is associated with a significant decrease in relative gamma, which significantly decreases during the recovery phase [43, 44].

Within this work, numerous articles discussing EEG stress stage classification were studied. I would like to mention the results of a few in the following lines. Varying stress stages were classified using a model based on the Support Vector Machine algorithm. The best results were obtained by combining fractal dimensions and statistical features. The authors obtained an average accuracy of 67.07 % with the stress divided into four stages. An accuracy of 75.22 % was obtained with three stress stages and an accuracy of 85.71 % with the simple distribution of the dataset to stressed and non-stressed stages [30].

Different researchers published stress stage classifiers using more sophisticated algorithms like Recurrent Neural Networks and Random Forests with only a single feature extracted – the Power Spectral Density. The RNN provided better classification accuracy of value 87 % for arousal and 83 % for valence. For comparison, the Random Forest model classified arousal with an accuracy of 83 % and valence with an accuracy of 75 % [34].

A wide range of research is focused on stress stage recognition using a multilayer Long Short-Term Memory (LSTM) classifier. A two-layer LSTM with extracted Power Spectral Density achieved a classification with 95 % accuracy [12].

Chapter 3

Proposed Methodology

This chapter deals with the main components of the proposed learning pipeline, which was designed to recognize stress stages from EEG data. The first step is to obtain suitable data for the training and testing of the deep learning model. For this thesis, a publicly available dataset was chosen. The subsequent step is pre-processing the data. In the context of my implementation, this step was omitted since the used dataset contains both raw and already pre-processed data. Thus, the dataset is ready for extraction of the relevant features. The acquired features are finally loaded into the model and the training process can start. The proposed steps are described in this chapter in more detail.

3.1 SAM40 Dataset

The SAM40 dataset provides EEG recordings of subjects experiencing varying stages of short-term induced stress. The recordings were made with forty subjects, with a mean age of 21.5 years, specifically fourteen females and twenty-six males. None of the subjects had health issues. Each subject had to participate in a series of recordings, which consisted of recordings of four different tasks repeated in three trials. In summary, 480 trials were performed. Subjects have performed three different cognitive tasks – the Stroop color-word test, the mirror image-recognition task, and the arithmetic problem-solving task. Each of them has also participated in relaxation trials. The state of relaxation was achieved by listening to relaxing music. The time duration of each trial was 25 s. Each trial was followed by the subjects' evaluation of the experienced stress stage on a scale of 1–10, where one is equivalent to the minimal amount of experienced stress and then is equivalent to the maximal amount of experienced stress [26].

The utilized EEG set is the Emotiv Epoc Flex gel kit, which contains 32 channels named and located according to the international 1020 system. CMS and DRL electrodes were used as a reference. The internal sampling frequency of the system is 1024 Hz. The sampling frequency of the obtained data is 128 Hz [26].

The dataset contains raw data and also a preprocessed version. According to the authors, first was applied a band-pass filter in the range of 0.5–45 Hz. Other artifacts, such as eye movements, muscular activity, etc., were removed using the combination of the Savitzky-Golay filter and wavelet thresholding [26].
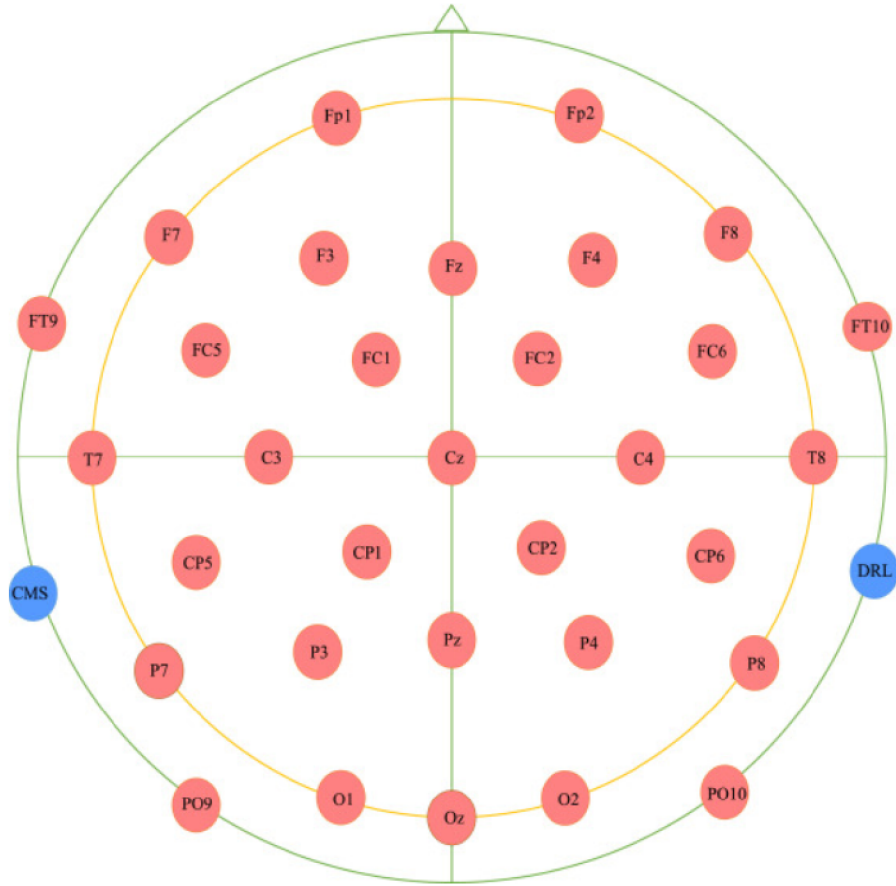
Figure 3.1: Electrode layout employed in recordings [26].

3.2 Data Preparation

Before feature extraction, the data preparation steps should be applied depending on the data and the model. These steps will include data segmentation, interpolation, and derivation.

3.2.1 Data Segmentation

 fussy The first step is data segmentation, also known as windowing. This step comprises splitting the time series data into overlapping or non-overlapping segments or windows. Data segmentation can increase the model's ability to learn from the data. It maintains the temporal structure of the data and, at the same time, provides contextual information about the sequence of data points. Windowing can also simplify data manipulation. In the case of this thesis, the original data will be segmented into non-overlapping time windows.

3.2.2 Interpolation

The dataset will be interpolated to achieve a more accurate reconstruction of the original signal, which can increase the model's ability to understand the underlying patterns.

23

3.2.3 Derivation

The derivation, which will be applied to the interpolated signal achieves quantification of the changes in the signal. Multiple-order derivations will be calculated. The best fitting order, based on the experiments will be used in the final model.

The dataset prepared in this way is finally ready for the extraction of relevant features, which is described in the following section.

3.3 Feature Extraction

Extracting the relevant features is one of the crucial steps in achieving a well-trained model with good classification results. Extracted features are the input that the learning model learns from during the training process. I propose utilizing the following features:

3.3.1 Time-Domain Features

- statistical features – mean, variance, standard deviation, peak-to-peak amplitude, skewness, kurtosis

- zero crossing rate

- entropies – approximate entropy, sampling entropy, spectral entropy, and singular value decomposition entropy (SVD entropy)

- Hurst exponent

- Hjorth parameters

3.3.2 Frequency-Domain Features

- power spectral density (PSD)

- band energy

3.3.3 Connectivity Analysis

- phase locking value (PLV)

The above-mentioned features were described in the literature, where they achieved good results in similar classification tasks and, therefore, will be used in models described in this thesis [32].

3.4 Classification

This section describes the division of the dataset into classes evaluating the stress stages experienced by the subjects and proposed classification models.

3.4.1 Stress Stages Classes

First, the way of stress stage classification has to be decided. The dataset provides trials with labels in the range from one to ten, where one is the minimum value of stress and ten is the maximum value of stress. This thesis deals with the classification of stress into three classes. The original labels will be distributed into the following classes describing the subject's experiences – *not stressed*, *moderate stressed*, and *very stressed*. The *not stressed* stage represents the original labels in the range from one to two. The *moderate stressed* stage includes the original labels in the range from three to five and finally, the *very stressed* stage symbolizes the original labels in the range from six to ten.

3.4.2 Classification Models

Classification task experiments will be performed with two different learning algorithms to create the optimal classifier of the stress stage. First, the Support Vector Machine (SVM), a machine learning algorithm, will be employed. The second algorithm utilized will be the Long Short-Term Memory (LSTM), a deep learning algorithm. Different projects that used these algorithms for stress stage classification tasks achieved accurate results. More detailed information about the results of these projects is provided in section 2.5. In both cases, experiments with varying models will be performed and compared.

SVM Models

- SVM with sigmoid kernel

- SVM with RBF kernel

- SVM with polynomial kernel

- SVM with linear kernel

Two-Layer and Three-Layer LSTM Model

Simple two-layer LSTM and three-layer LSTM models are proposed. Both LSTM layers contain 40 neurons. The last LSTM layer of both of these models is followed by a dropout layer with a dropout rate of 0.3. In both cases, the dropout is followed by a linear layer. The last layer is the softmax layer, which outputs the probability of each class of the classifier. Both models are inspired by studied literature [51].
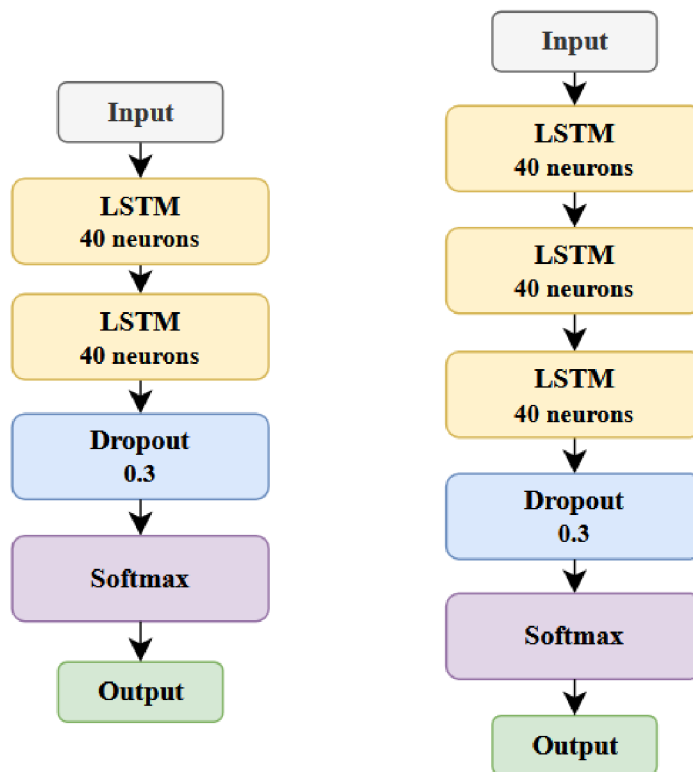
Figure 3.2: The illustration of the model with a two-layer LSTM on the left and then with a three-layer LSTM on the right.

Advanced LSTM Model

The third proposed LSTM model is more complex than the previous two. Because of these reasons, it is called the advanced LSTM model. The base of this model is almost the same as in the previous two-layer LSTM – two layers of LSTM with 40 neurons followed by the dropout layer with a dropout rate of 0.5. The next layer is the sigmoid activation function layer with 20 neurons, again followed by a dropout layer with a dropout rate of 0.5. The following layer is the rectifier activation function (ReLU). The ReLU can be very helpful when dealing with the vanishing gradient problem [62]. The last layer is the same as in the previous two models – a softmax layer. This model is inspired by a studied literature [60].
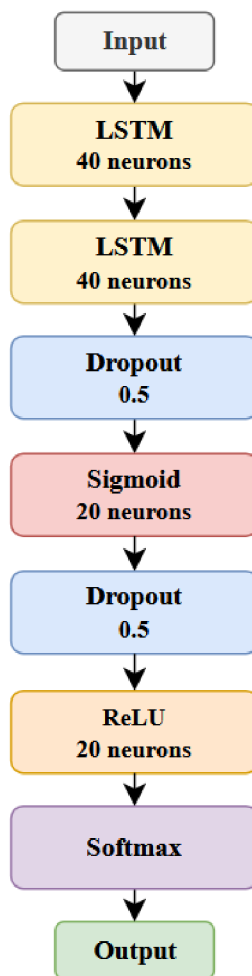
Figure 3.3: The illustration of the more advanced model of the LSTM.

3.5 Validation

This section describes the employed methods of validation. First, the dataset will be split into a training set and a testing set. The training set will be created as a randomly chosen series of data formed by 80 % of the original dataset. The other 20 % will be later used as the testing dataset. Another validation approach that will be used is the Stratified K-Fold Cross-Validation. The purpose is to avoid potential problems with the underrepresentation of any of the classes. The training dataset will be split into a number of folds. A new model will be trained on each fold. The best of these models will be selected and used for the final testing on the separated 20 % of the dataset, the testing set. A concise illustration of the validation process is provided in figure 3.4.
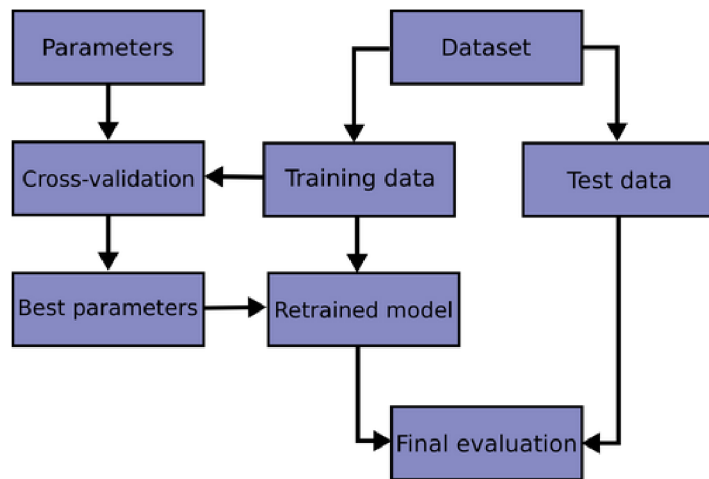
Figure 3.4: The scheme illustrating the validation process proposed in this thesis [11].

Chapter 4

Implementation

This chapter focuses on further describing the implementation details. The solution was created using Python programming language, which provides powerful libraries specialized in implementing machine learning and deep learning algorithms, processing EEG data, and other useful functionalities important for this thesis. It is necessary to mention the utilized libraries and further describe the philosophy of this solution. The object-oriented programming paradigm was employed. The implementation is based on the methodology, that was proposed in chapter 3.

4.1 Data Augmentation

During the model training experiments, the SAM40 dataset turned out to be too small to efficiently train on such complex data as EEG recordings. Further information about the dataset is in section 3.1. The training process of deep learning models requires large amounts of data. Therefore, it was decided to utilize data augmentation to achieve more accurate results. This approach has proven to be successful. Three different methods of data augmentation and a combination of them were applied to the dataset SAM40.

The first set of augmented data was achieved by shifting the original signals. Data from all trials were separately shifted, either to the right or left, depending on the randomly generated integer in the range of -50 to +50. The range of values used in randomization was derived by experiments. The second set of augmented data was performed by adding a Gaussian noise to the original signal. This method creates random samples from a Gaussian distribution according to set parameters. The generated Gaussian noise had a mean distribution of 0 and a standard deviation of 0.05. The third data augmentation was accomplished by a combination of the two previous techniques. The fourth and final set of augmented data was created by scaling the original signal. The random scaling transformation was applied with a random scale factor between 0.8 and 1.2 [37].

In this manner, the original dataset with 480 trials was expanded to 2400 trials.

4.2 Object-Oriented Design

This section describes the object-oriented design utilized in the implementation. The object-oriented paradigm was used to reach an easier orientation in the code and troubleshooting. Another advantage is the possibility of comfortable editing and flexible replacement, removal, or addition of modules if needed.

This approach allows easy use of varying datasets and comfortable experimentation with data preparation. Many different feature extractions can be used, and additional ones can be implemented easily. The underlying learning algorithm can be easily swapped for a new one.
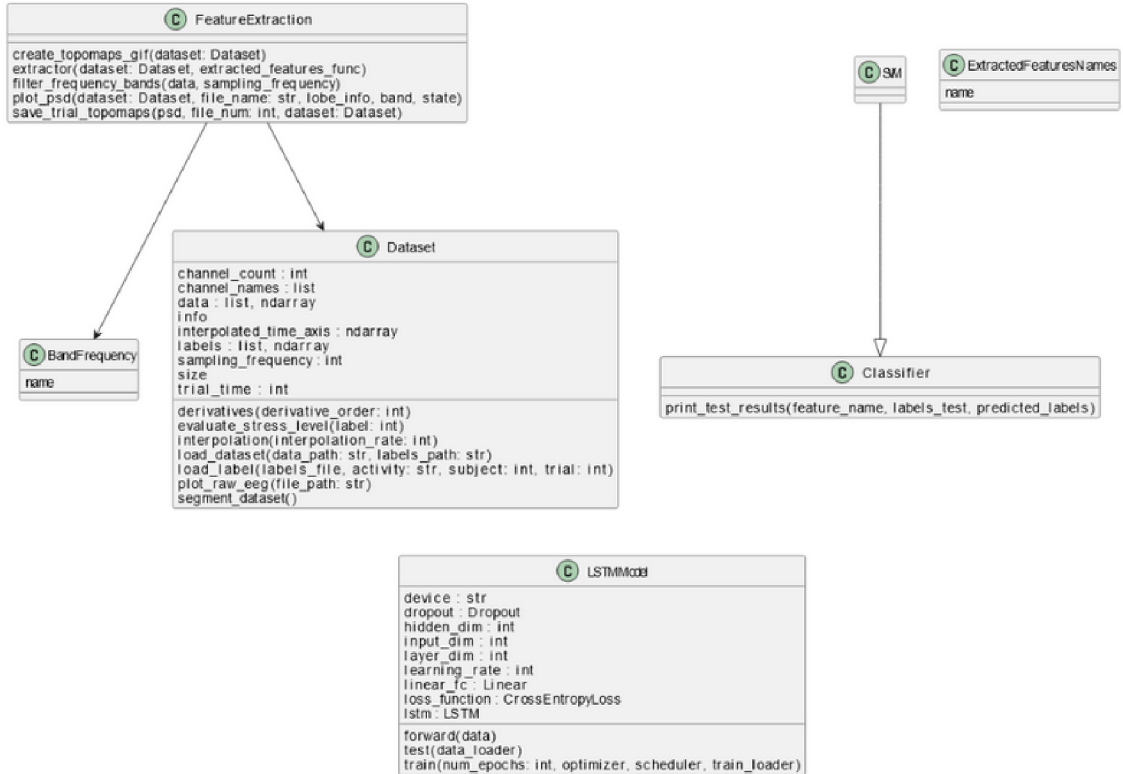

Figure 4.1: Class diagram.

4.3 Data Loading

The files are loaded one by one from the original `.mat` format into the array using the `glob()` function from `glob` library and `loadmat()` function from `scipy` library. The array preserves the dimension of the input data to maintain the information about the time window and the position of the EEG channel. According to the name of the `.mat` file, the corresponding label is found in the `.xls` file with labels and assigned to one of the three classes representing the stress stage. The new label is stored in the label array. The `.xls` file is processed using the `read_excel()` function from `pandas` library.

4.4 Data Preparation

Data segmentation is the first step in preparing the dataset. The time duration of the single time window is 1 s. According to the literature, using a window size shorter than 0.5 s can corrupt the data patterns and degrade the performance of the model [2]. At the same time, we want to keep the number relatively low to get as much training data as possible.

Then, the data is interpolated. Since the EEG data is non-periodic, the *B-spline*, which is the non-periodic interpolation function, is used to achieve the desired results. Utilizing the interpolation, the sampling rate is increased to 300. Interpolation is implemented using the `make_interp_spline()` function from `interpolate` module of `scipy` library.

The last step of data preparation is calculating the derivatives. Derivative of any order can be computed. The best results were obtained with the first and second-order derivatives. The second-order derivative is, therefore, incorporated in the implementation of the final model. The derivatives are implemented using the `gradient()` function from `numpy` library.

4.5 Feature Extraction

Extraction of all proposed features in chapter 3 is implemented. For these purposes is utilized the `FeatureExtractor` class from the `feature_extraction` module of the `mne_features` library. This class provides a very elegant solution for extracting features using the build-in `mne_features` module's functions without a large number of repetitive code. The features to be extracted are specified as the input parameter for the initialization of this class. This class expects the input parameter to be a list with names of `mne_features` build-in functions designed for feature extraction. For simplicity, an enumerator with the names of these functions was created. Since the library is focused on the feature extraction of the EEG data, the dimension of the data array, which maintains information about the EEG channel position and time window of the data, does not have to be changed.

4.6 Classification

This section describes the necessary steps to load the dataset into the model, implement the classifiers, and validate the model.

In cases of both SVM and LSTM classifiers, the same validation techniques are applied. The first step is to split the dataset into the training and the testing set using the `train_test_split()` function from `sklearn.model_selection` module. The training set matches 80 % of the dataset. The remaining 20 % corresponds to the testing set. The Stratified K-fold Cross-Validation technique (3.5) is applied to the training dataset. The training dataset is split into 5 different folds. Varying numbers of folds were tried in experiments, including the very common 10 folds in the literature, but the best results were achieved with 5 folds. The effective number of folds corresponds with the size of the dataset. The stratified k-fold cross-validation was implemented using the `StratifiedKFold()` function from the `sklearn.model_selection` module.

4.6.1 Support Vector Machine (SVM)

The SVM is the first model implemented in this solution (2.4.5). All models mentioned in the proposed methodology chapter 3 were implemented and tested in numerous experiments. Varying settings of the hyperparameters were applied and tested. The implementation of the model is realized using the `SVM()` function from `sklearn.SVM` module. The tuning of the model's hyperparameters is supported utilizing the `GridSearchCV()` function from `sklearn.model_selection` module. This function provides a search for the optimal hyperparameter setting of the specified model (SVM). The specific parameter grid containing the dictionary of hyperparameters to be applied has to be the input of this function

and delivers an option to easily apply a whole range of hyperparameters. The optimization is achieved by cross-validation grid-search over the parameter grid, specifically the stratified k-fold cross-validation in the case of this solution. The hyperparameters, that are employed in the SVM are the kernel function and the `c` parameter. The `c` parameter is balancing the trade-off between the low training error and the potential of enabled misclassification 2.4.5. The best combination is applied in the final model.

4.6.2 Long Short-Term Memory (LSTM)

The layers applied in the LSTM model are further described in the proposed methodology chapter 3.4.2. The LSTM model layers are implemented using the `nn` module from the `torch` library, so the first step to achieving a classification model is to transform the input features into data type, which is required by this library – tensors. `Torch` library provides a whole range of functions that simplify the work with the data, such as `TensorDataset`, which transforms the data into the tensor dataset format, and the `DataLoader` function, which provides the easy loading of the dataset into the model.

The optimization algorithm used for the optimization of the weights of the LSTM is the Adam optimizer from the `torch.optim` module. Other optimization algorithms were tested in experiments, but the models using the Adam optimizer achieved better results. Adam optimizer is computationally efficient and, at the same time, has low memory requirements. It minimizes the oscillations during gradient descent when approaching the global minimum, and the step size is large enough to pass the local minimum [25].

The `learning rate` hyperparameter is tuned using the `ExponentialLR()` function from the `torch.optim.lr_scheduler` module. This function provides a learning rate scheduler that supplies the exponential decrease of the learning rate value with every epoch [16].

The model's performance is quantified using the `torch.nn` module `CrossEntropyLoss()` function. This function measures the model's performance based on the probabilities output by the model and is suitable for multi-class classification [10].

4.7 Displaying the Results

The progress of the training process is monitored by printing the loss value. The classification ability of the model is evaluated using the `classification_report` function, which calculates the validation metrics, like accuracy, precision, f-1 score, etc., and with the *confusion_matrix* function, which prints the confusion matrix. Both functions are from the `sklearn.metrics` module. The `matplotlib.pyplot` module is employed to plot the graphs. This solution also provides functionality that generates pictures of the topomaps illustrating the changes of the specified extracted feature, e.g., PSD, of a specified brainwave frequency band and a time. This functionality is implemented with the `plot_topomap` function from `mne.viz` module. Finally, this implementation provides a functionality, that creates a GIF from saved topomaps of a specified trial. For these purposes, the `imageio` library is used.

Chapter 5

Results

This chapter describes the results achieved with the proposed and subsequently implemented classifier models. The achievements are compared with the results presented in the scientific literature.

Regardless of the stress response phase, stress typically manifests in a decrease of alpha power, and an increase of beta power, both especially in the frontal lobe. Stress also manifests itself in other measures further described in 2.5. Figure 5.2 illustrates the alpha band power spectrum from the frontal lobe EEG channels of subject n. 20 during the 1. trial of the arithmetic task. This trial is classified as *very stressed* class. For comparison, the figure 5.1 depicts the alpha band power spectrum from the frontal lobe EEG channels of subject n. 1 during the 1. trial of the relaxation, which is classified as *not stressed* class. A decrease in the alpha power when experiencing stress in comparison with relaxation is evident just at a glance on plots of some EEG channels, e.g., Fp_1 and Fp_2 (both have a green color of the line varying in shade).
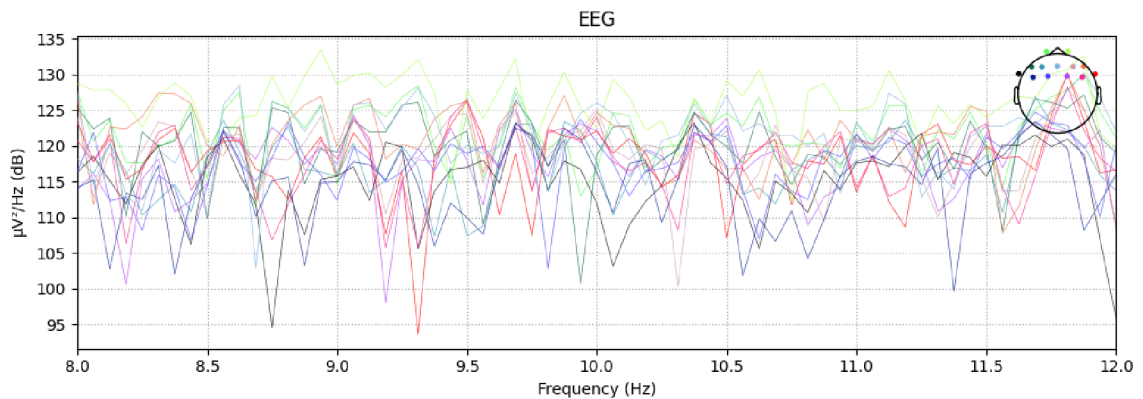

Figure 5.1: The alpha band power spectrum from the frontal lobe EEG channels of the subject in a *not stressed* state – specifically subject n. 1, the 1. of the relaxation.

For a better illustration, the topomaps with the PSD of 3 time windows depicting the corresponding trial are supplied in figures 5.3, 5.4, and 5.5. A decrease in alpha power in time can be observed in several channels of the frontal and parietal lobes. It is important to mention that this progression does not have to be observed in all time windows because an electroencephalogram consists of highly dynamic data due to the other ongoing brain processes and has to be judged more likely in its entirety.
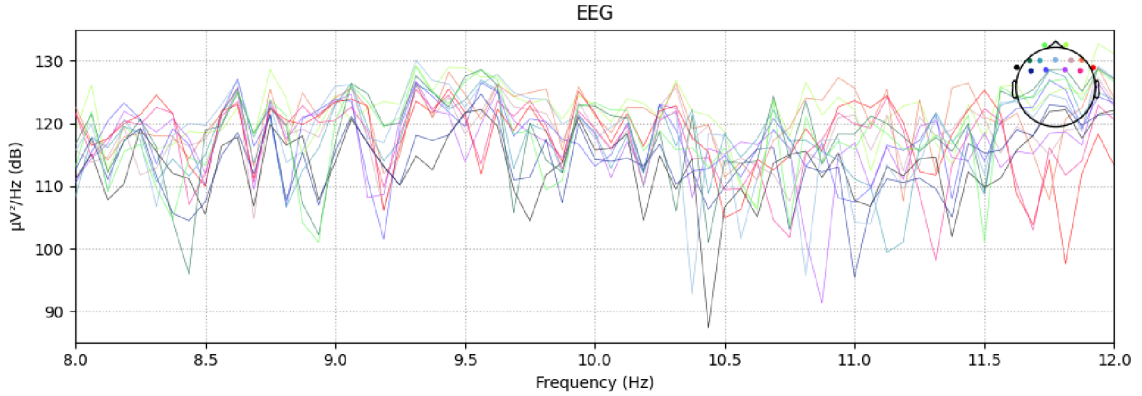
Figure 5.2: The alpha band power spectrum from the frontal lobe EEG channels of the subject in a *very stressed* state – specifically subject n. 20, the 1. trial of the arithmetic task.
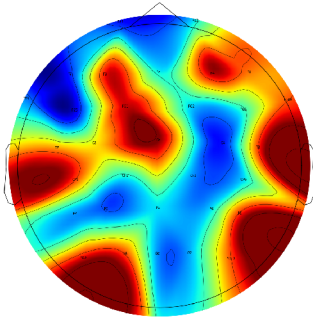

Figure 5.3: Topomap with the PSD of the 1. time window (0 – 1 s) of the arithmetic trial, which is classified as a *very stressed*.
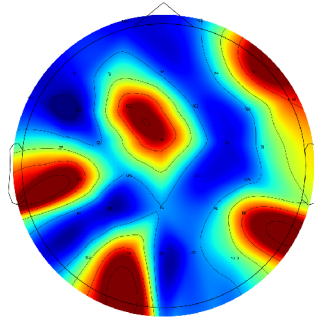
Figure 5.4: Topomap with the PSD of the 2. time window (1 – 2 s) of the arithmetic trial, which is classified as a *very stressed*.
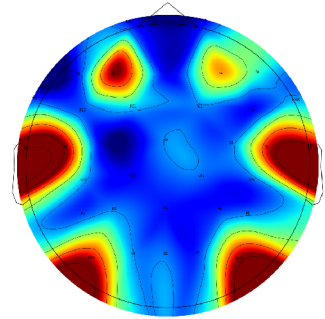
Figure 5.5: Topomap with the PSD of the 6. time window (5 – 6 s) of the arithmetic trial, which is classified as a *very stressed*.

A detailed study of stress phases in the EEG data is, however, not the purpose of this thesis, and therefore is not further discussed.

The following lines focus on the evaluation and the summary of the results achieved with implemented classification models. All models proposed in chapter 3 were created. A description of the implementation details is provided in chapter 4.

Finding a suitable setting for data preparation, training models with a number of extracted features one by one and in groups, and training varying classification models with a whole range of hyperparameters that have to be tuned is quite a time-consuming process. For these reasons, only certain combinations of input parameters were tested as a part of the solution.

The first achieved results were not satisfactory. Even when using the parameter grid for the tuning of the SVM hyperparameters, the best accuracy achieved was 41 %. Validation metrics are described in table 5.1 in more detail.

classes	precision (%)	recall (%)	f1-score (%)	support (-)
not stressed	0.42	0.56	0.48	41
moderate stressed	0.36	0.36	0.36	36
very stressed	0.60	0.16	0.25	19
accuracy			0.41	96
macro avg	0.46	0.36	0.36	96
weighted avg	0.43	0.41	0.39	96

Table 5.1: Validation metrics achieved with the SVM model with the original dataset. Extracted feature loaded into the model was PSD.

The results acquired with the LSTM models were just as problematic. As described in the table 5.2, the best achieved accuracy was 44 %.

classes	precision (%)	recall (%)	f1-score (%)	support (-)
not stressed	0.39	0.41	0.40	32
moderate stressed	0.45	0.60	0.51	35
very stressed	0.50	0.28	0.36	29
accuracy			0.44	96
macro avg	0.45	0.43	0.42	96
weighted avg	0.45	0.44	0.43	96

Table 5.2: Validation metrics achieved with the two-layer LSTM model with 40 hidden neurons using Hjorth parameters extracted from the original dataset as an input.

One of many problems these models' had was the uneven distribution of classes in training and testing datasets and probably the insufficient quantity of training data. Even after testing many combinations of hyperparameters, extracted features, ways of data preparation, and implementation of the stratified k-fold cross-validation, which solves the problem of uneven class distribution, the results did not improve. That is when data augmentation, which was mentioned in the literature as a helpful solution when dealing with small datasets, came into consideration. Data augmentation finally turned out to be the right solution.

Experiments with varying features extracted from the original dataset revealed the best results by using Hjorth parameters, power spectral density (PSD), and phase-locking value (PLV). Therefore, further experiments with the augmented dataset were performed only with these features.

Despite high expectations, the advanced LSTM model turned out to be the worst of the proposed models. This model was inspired by the one introduced in the literature, where authors classify stress into four classes and achieved an accuracy of 93.27 %. The best result achieved with this model within this solution is described in table 5.3. There can be several reasons why this model was unsuccessful, starting with the excessive complexity of the model in view of the dataset and the inappropriate setting of hyperparameters, ending with issues caused by unsuitable combinations of layers according to different studied literature. Another reason why the results of this model might be worse is because of the less precise process of the hyperparameter tuning, which was not performed due to time constraints.

classes	precision (%)	recall (%)	f1-score (%)	support (-)
not stressed	0.52	0.59	0.55	152
moderate stressed	0.56	0.53	0.55	187
very stressed	0.52	0.49	0.50	141
accuracy			0.54	480
macro avg	0.53	0.53	0.53	480
weighted avg	0.54	0.54	0.53	480

Table 5.3: Validation metrics achieved with the advanced LSTM model using the Hjorth parameters extracted from the augmented dataset.

Among the two-layer LSTM models, the best results were achieved with the two-layer LSTM model with 40 hidden neurons. The accuracy earned by testing this model is 67 %. Other validation metrics are depicted in table 5.5. The phase locking value (PLV) was extracted from the second-order derivative values of the augmented dataset. The setting of the LSTM hyperparameters is described in table 5.4. Information about incorporated layers is in more detail described in section 3.4.2.

hyperparameter	value
batch size	50
number of epochs	200
learning rate	0.0007
learning rate decay	0.999

Table 5.4: Setting of two-layer LSTM hyperparameters with the best results.

classes	precision (%)	recall (%)	f1-score (%)	support (-)
not stressed	0.67	0.69	0.68	159
moderate stressed	0.69	0.72	0.70	203
very stressed	0.62	0.55	0.58	118
accuracy			0.67	480
macro avg	0.66	0.65	0.66	480
weighted avg	0.67	0.67	0.67	480

Table 5.5: Validation metrics achieved with the best version of the two-layer LSTM model with 40 hidden neurons in each layer. Phase locking value was extracted feature in this model.

Extracting the phase-locking value (PLV) from the second-order derivative of the augmented dataset achieved the best results among the three-layer LSTM models. The validation metrics achieved with the testing dataset are summarized in the table 5.7. Further description of the model is provided in section 3.4.2. The setting of hyperparameters is depicted in table 5.6.

It was already mentioned before that this implementation uses a parameter grid as an input with the hyperparameters for the SVM model. A more detailed description of the parameter grid and the `GridSearchCV()` function is in section 4.6.1. The best results of SVM models were achieved with the parameter grid containing the list of c parameters and the list of kernel functions set as in the following code snippet. The parameter grid is the illustrated Python dictionary called `svm_parameters`.

hyperparameter	value
batch size	50
number of epochs	300
learning rate	0.0007
learning rate decay	0.999

Table 5.6: Setting of three-layer LSTM hyperparameters that achieved the best results.

classes	precision (%)	recall (%)	f1-score (%)	support (-)
not stressed	0.72	0.69	0.70	163
moderate stressed	0.72	0.75	0.74	179
very stressed	0.65	0.64	0.65	138
accuracy			0.70	480
macro avg	0.70	0.70	0.70	480
weighted avg	0.70	0.70	0.70	480

Table 5.7: Validation metrics achieved with the best version of the three-layer LSTM model with 40 hidden neurons in each layer. The extracted feature in this model was the phase-locking value (PSD).

```
svm_parameters = {
    'C' : [0.1,1,10,100,1000],
    'kernel' : ['sigmoid', 'linear', 'poly', 'rbf']
}
```

Next, Hjorth parameters were extracted from the first-order derivative values of the augmented dataset. SVM model with this setting achieved an accuracy of 90 %, which is the best classification result gained in this thesis. Other validation metrics are depicted in table 5.8.

classes	precision (%)	recall (%)	f1-score (%)	support (-)
not stressed	0.95	0.87	0.91	187
moderate stressed	0.82	0.96	0.89	171
very stressed	0.96	0.85	0.90	122
accuracy			0.90	480
macro avg	0.91	0.90	0.90	480
weighted avg	0.91	0.90	0.90	480

Table 5.8: Validation metrics achieved with the best version of the SVM model.

5.1   Summary

This section summarizes the results of the created classifiers and discusses them in comparison with those found in the literature. The table 5.9 recapitulates the acquired validation metrics of the proposed and implemented models in this thesis.

Research done as a part of this thesis found two articles dealing with the same topic, using similar learning algorithms and the SAM40 dataset. The first of these articles describes a classifier of two stress stages, a *stressed* and *not stressed* class. This solution is based on learning multi-variate weighted visibility graphs with Graph Signal Processing

model	accuracy (%)	precision (%)	recall (%)	f1-score (%)
SVM	0.90	0.91	0.90	0.90
two-layer LSTM	0.67	0.66	0.65	0.66
three-layer LSTM	0.70	0.70	0.70	0.70
advanced LSTM	0.54	0.53	0.53	0.53

Table 5.9: Summary of the validation metrics achieved with implemented models – accuracy, macro average precision, macro average recall, and macro average f1-score.

(GSP) techniques as feature extraction and subsequent classification using SVM. The results achieved with this approach are comparable with those achieved in this thesis, but it is important to highlight that it was a classifier of two classes, not three. The accuracy that was achieved was 93.38 % [42].

The second article represents a complex solution using the LSTM. First, the Discrete Wavelet Transform and Particle Swarm Optimization (DWT-PSO) based pre-processing was applied. The classification is based on the hybrid deep learning model using Grid Search Hyperparameter Optimization (GSHPO) based stacked Bidirectional LSTM and LSTM. According to the studied article, the accuracy of this binary classifier is remarkably 98.07 %. The model described in this solution provides better results and, therefore, might be a better solution for this classification task, but it is still only a binary classifier [31].

In comparison with the literature, the SVM model created in this thesis provides good ternary classification results with an accuracy of 90 %. The three-layer LSTM model provides relatively good results with an accuracy of 70 %, which leaves space for future work on tuning the hyperparameters.

Chapter 6

Conclusion

This thesis aimed to create a machine learning classifier that detects a patient's stage of stress based on EEG recordings. Given the increase of mental health issues in the population observed in recent years, the topic of this thesis is very relevant. It is proven that chronic stress can significantly worsen the state of patients with mental health issues and can also be a triggering factor for depression, anxiety, or burnout syndrome. Therefore, early detection of chronic stress among patients might be very beneficial. Electroencephalography (EEG), in connection with machine learning and deep learning algorithms, provides a cheap, non-invasive, and effective solution to this problem.

This thesis explored the basics of EEG and outlined the architecture of the human brain on a level relevant to this study. Furthermore, it studied the mechanism of stress and, above all, the analysis of EEG features, specifically feature extraction, selection, and machine learning and deep learning algorithms used for classification. An overview of the current state of research was also provided.

This thesis successfully implemented and tested various ternary classification models using Support Vector Machine (SVM) – a machine learning algorithm, and Long Short-Term Memory (LSTM) – a deep learning algorithm. The best results were achieved through the application of data augmentation techniques and extraction of Hjorth parameters and of a phase locking value (PLV). An accuracy of 90 % was achieved using the SVM model. The advantage of this model is a ternary classification. In comparison, a previously published article achieved comparable results but only with a binary classifier [42]. The best LSTM model implemented within this thesis was a three-layer LSTM, which achieved a ternary classification accuracy of 70 %. This result leaves potential for further improvements by tuning the hyperparameters, or modifying the architecture of the model's layers themselves.

Bibliography

[1] *Accuracy vs. precision vs. recall in machine learning: what's the difference?* [online]. Evidently AI Team [cit. 2024-4-3]. Available at: https://www.evidentlyai.com/classification-metrics/accuracy-precision-recall.

[2] ATYABI, A. and POWERS, D. M. W. The impact of segmentation and replication on non-overlapping windows: An EEG study. In: *2012 IEEE International Conference on Information Science and Technology*. 2012, p. 668–674. DOI: 10.1109/ICIST.2012.6221730.

[3] BABLANI, A., EDLA, D. R. and DODIA, S. Classification of EEG Data using k-Nearest Neighbor approach for Concealed Information Test. *Procedia Computer Science*. 2018, vol. 143, p. 242–249. DOI: 10.1016/j.procs.2018.10.392. ISSN 1877-0509. 8th International Conference on Advances in Computing & Communications (ICACC-2018). Available at: https://www.sciencedirect.com/science/article/pii/S1877050918320891.

[4] BAHETI, P. *A Comprehensive Guide to Convolutional Neural Networks* [online]. June 2021 [cit. 2024-4-10]. Available at: https://www.v7labs.com/blog/convolutional-neural-networks-guide.

[5] BERRETZ, G., PACKHEISER, J., WOLF, O. T. and OCKLENBURG, S. Acute stress increases left hemispheric activity measured via changes in frontal alpha asymmetries. *IScience*. 2022, vol. 25, no. 2, p. 103841. DOI: https://doi.org/10.1016/j.isci.2022.103841. ISSN 2589-0042. Available at: https://www.sciencedirect.com/science/article/pii/S2589004222001110.

[6] *Brain Anatomy and How the Brain Works* [online]. Johns Hopkins Medicine, Jul 2021 [cit. 2023-11-15]. Available at: https://www.hopkinsmedicine.org/health/conditions-and-diseases/anatomy-of-the-brain.

[7] BUZZELL, G. A., NIU, Y., AVIYENTE, S. and BERNAT, E. A practical introduction to EEG Time-Frequency Principal Components Analysis (TF-PCA). *Developmental Cognitive Neuroscience*. 2022, vol. 55, p. 101114. DOI: 10.1016/j.dcn.2022.101114. ISSN 1878-9293. Available at: https://www.sciencedirect.com/science/article/pii/S1878929322000573.

[8] CECHETTO, D. F. and TOPOLOVEC, J. C. Cerebral Cortex. In: RAMACHANDRAN, V., ed. *Encyclopedia of the Human Brain*. New York: Academic Press, 2002, p. 663–679. DOI: https://doi.org/10.1016/B0-12-227210-2/00087-X. ISBN 978-0-12-227210-3. Available at: https://www.sciencedirect.com/science/article/pii/B012227210200087X.

[9] *Chronic Stress* [online]. Yale Medicine, august 2022 [cit. 2023-11-15]. Available at: https://www.yalemedicine.org/conditions/stress-disorder.

[10] CONTRIBUTORS, P. *CrossEntropyLoss* [online]. 2023 [cit. 2024-5-1]. Available at: https://pytorch.org/docs/stable/generated/torch.nn.CrossEntropyLoss.html.

[11] DEVELOPERS, S. learn. *Cross-validation: evaluating estimator performance* [online]. [cit. 2024-4-26]. Available at: https://scikit-learn.org/stable/modules/cross_validation.html.

[12] DHAKE, D., GAIKWAD, K., GUNJAL, S. and WALUNJ, S. LSTM Algorithm for the Detection of Mental Stress in EEG. 2023, p. 1–6. DOI: 10.1109/CONIT59222.2023.10205636.

[13] GAIKWAD, P. and PAITHANE, A. N. Novel approach for stress recognition using EEG signal by SVM classifier. 2017, p. 967–971. DOI: 10.1109/ICCMC.2017.8282611.

[14] GARCIA, B., MARTINEZ RODRIGO, A., ZANGRONIZ, R., PASTOR GARCÍA, J. M. and ALCARAZ, R. Application of Entropy-Based Metrics to Identify Emotional Distress from Electroencephalographic Recordings. *Entropy.* june 2016, vol. 18, p. 221. DOI: 10.3390/e18060221.

[15] GARCÍA BUENO, B., CASO, J. R. and LEZA, J. C. Stress as a neuroinflammatory condition in brain: Damaging and protective mechanisms. *Neuroscience & Biobehavioral Reviews.* 2008, vol. 32, no. 6, p. 1136–1151. DOI: 10.1016/j.neubiorev.2008.04.001.

[16] GEEKSFORGEEKS. *Cross Validation in Machine Learning* [online]. December 2023 [cit. 2024-4-3]. Available at: https://www.geeksforgeeks.org/cross-validation-machine-learning/.

[17] GEEKSFORGEEKS. *Deep Learning | Introduction to Long Short Term Memory* [online]. December 2023 [cit. 2024-4-2]. Available at: https://www.geeksforgeeks.org/deep-learning-introduction-to-long-short-term-memory/.

[18] GEEKSFORGEEKS. *F1 Score in Machine Learning* [online]. December 2023 [cit. 2024-4-3]. Available at: https://www.geeksforgeeks.org/f1-score-in-machine-learning/.

[19] GEEKSFORGEEKS. *Impact of learning rate on a model* [online]. July 2023 [cit. 2024-5-1]. Available at: https://www.geeksforgeeks.org/impact-of-learning-rate-on-a-model/.

[20] GEEKSFORGEEKS. *Introduction to Recurrent Neural Network* [online]. March 2023 [cit. 2024-4-2]. Available at: https://www.geeksforgeeks.org/introduction-to-recurrent-neural-network/.

[21] GEEKSFORGEEKS. *Support Vector Machine (SVM) Algorithm* [online]. June 2023 [cit. 2024-4-10]. Available at: https://www.geeksforgeeks.org/support-vector-machine-algorithm/.

[22] GEEKSFORGEEKS. *Feature Selection Techniques in Machine Learning* [online]. March 2024 [cit. 2023-12-10]. Available at: https://www.geeksforgeeks.org/feature-selection-techniques-in-machine-learning/.

[23] GEEKSFORGEEKS. *Introduction to Convolution Neural Network* [online]. March 2024 [cit. 2024-4-10]. Available at: https://www.geeksforgeeks.org/introduction-convolution-neural-network/.

[24] GEEKSFORGEEKS. *Random Forest Algorithm in Machine Learning* [online]. February 2024 [cit. 2024-3-25]. Available at: https://www.geeksforgeeks.org/random-forest-algorithm-in-machine-learning/.

[25] GEEKSFORGEEKS. *What is Adam Optimizer?* [online]. March 2024 [cit. 2024-4-29]. Available at: https://www.geeksforgeeks.org/adam-optimizer/.

[26] GHOSH, R., DEB, N., SENGUPTA, K., PHUKAN, A., CHOUDHURY, N. et al. SAM 40: Dataset of 40 subject EEG recordings to monitor the induced-stress while performing Stroop color-word test, arithmetic task, and mirror image recognition task. *Data in Brief*. 2022, vol. 40, p. 107772. DOI: 10.1016/j.dib.2021.107772. ISSN 2352-3409. Available at: https://www.sciencedirect.com/science/article/pii/S2352340921010465.

[27] GLICKSTEIN, M. What does the cerebellum really do? *Current Biology*. 2007, vol. 17, no. 19, p. R824–R827. DOI: https://doi.org/10.1016/j.cub.2007.08.009. ISSN 0960-9822. Available at: https://www.sciencedirect.com/science/article/pii/S096098220701785X.

[28] GU, J., WANG, Z., KUEN, J., MA, L., SHAHROUDY, A. et al. Recent advances in convolutional neural networks. *Pattern Recognition*. 2018, vol. 77, p. 354–377. DOI: 10.1016/j.patcog.2017.10.013. ISSN 0031-3203. Available at: https://www.sciencedirect.com/science/article/pii/S0031320317304120.

[29] HOLM, A., LUKANDER, K., KORPELA, J., SALLINEN, M. and MÜLLER, K. Estimating Brain Load from the EEG. *TheScientificWorldJournal*. february 2009, vol. 9, p. 639–51. DOI: 10.1100/tsw.2009.83.

[30] HOU, X., LIU, Y., SOURINA, O., TAN, Y. R. E., WANG, L. et al. EEG Based Stress Monitoring. 2015, p. 3110–3115. DOI: 10.1109/SMC.2015.540.

[31] JADHAV, A., MALVIYA, L., SHANDILYA, S. K. and MAL, S. Human Stress Detection from SWCT EEG Data Using Optimised Stacked Deep Learning Model. In:. June 2023, p. 183–196. DOI: 10.1007/978-981-99-3478-2_17. ISBN 978-981-99-3477-5.

[32] KATMAH, R., AL SHARGIE, F., TARIQ, U., BABILONI, F., AL MUGHAIRBI, F. et al. A Review on Mental Stress Assessment Methods Using EEG Signals. *Sensors*. july 2021, vol. 21. DOI: 10.20944/preprints202107.0255.v1.

[33] KETKAR, N. and MOOLAYIL, J. Introduction to Machine Learning and Deep Learning. In: *Deep Learning with Python*. Berkeley, CA: Apress, 2021, chap. 1, p. 1–25. DOI: 10.1007/978-1-4842-5364-9_1. Available at: https://doi.org/10.1007/978-1-4842-5364-9_1.

[34] KHAN, M. R. and AHMAD, M. Mental Stress Detection from EEG Signals Using Comparative Analysis of Random Forest and Recurrent Neural Network. 2024, p. 1–6. DOI: 10.1109/iCACCESS61735.2024.10499496.

[35] KORTINK, E., WEEDA, W., CROWLEY, M., MOOR, B. and MOLEN, M. van der. Community structure analysis of rejection sensitive personality profiles: A common neural response to social evaluative threat? *Cognitive Affective & Behavioral Neuroscience.* april 2018, vol. 18. DOI: 10.3758/s13415-018-0589-1.

[36] LANG, E., TOMÉ, A., KECK, I., GORRIZ, J. and PUNTONET, C. Brain Connectivity Analysis: A Short Survey. *Computational intelligence and neuroscience.* october 2012, vol. 2012, p. 412512. DOI: 10.1155/2012/412512.

[37] LASHGARI, E., LIANG, D. and MAOZ, U. Data Augmentation for Deep-Learning-Based Electroencephalography. *Journal of Neuroscience Methods.* july 2020, vol. 346, p. 108885. DOI: 10.1016/j.jneumeth.2020.108885.

[38] *Lobes of the brain* [online]. The University of Queensland Australia, Jul 2021 [cit. 2023-11-17]. Available at: https://qbi.uq.edu.au/brain/brain-anatomy/lobes-brain.

[39] MALVIYA, L., MAL, S. and LALWANI, P. EEG Data Analysis for Stress Detection. 2021, p. 148–152. DOI: 10.1109/CSNT51715.2021.9509713.

[40] MANLIN, Y., LEI, Y., LI, P., YE, Q., LIU, Y. et al. Shared Sensitivity to Physical Pain and Social Evaluation. *The Journal of Pain.* november 2019, vol. 21. DOI: 10.1016/j.jpain.2019.10.007.

[41] MARIOTTI, A. The effects of chronic stress on Health: New Insights Into the molecular mechanisms of brain–body communication. *Future Science OA.* 2015, vol. 1, no. 3. DOI: 10.4155/fso.15.21.

[42] MATHUR, P., KAISTHA, S. and CHAKKA, V. K. Mental Task Induced Stress Detection using Multi-Variate Weighted Visibility Graph (MV-WVG) from EEG Signals. In: *2023 IEEE 20th India Council International Conference (INDICON).* 2023, p. 1265–1270. DOI: 10.1109/INDICON59947.2023.10440952.

[43] MINGUILLON, J., LOPEZ, M. and PELAYO, F. Stress Assessment by Prefrontal Relative Gamma. *Frontiers in Computational Neuroscience.* september 2016, vol. 10. DOI: 10.3389/fncom.2016.00101.

[44] MINGUILLON, J., LOPEZ, M., RENEDO CRIADO, D., SANCHEZ CARRION, M. and PELAYO, F. Blue lighting accelerates post-stress relaxation: Results of a preliminary study. *PLOS ONE.* october 2017, vol. 12, p. e0186399. DOI: 10.1371/journal.pone.0186399.

[45] NIDAL, K. and MALIK, A. S. *EEG/ERP Analysis: Methods and Applications.* 1st ed. CRC Press, 2014. ISBN 9780429170836.

[46] OH, S.-H., LEE, Y.-R. and KIM, H.-N. A novel EEG feature extraction method using Hjorth parameter. *International Journal of Electronics and Electrical Engineering.* 2014, p. 106–110. DOI: 10.12720/ijeee.2.2.106-110.

[47] OLBRYS, J. and MAJEWSKA, E. Approximate entropy and sample entropy algorithms in financial time series analyses. In:. 2022, vol. 207, p. 255–264. DOI: https://doi.org/10.1016/j.procs.2022.09.058. ISSN 1877-0509. Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 26th International Conference KES2022. Available at: https://www.sciencedirect.com/science/article/pii/S1877050922009310.

[48] PAWAN and DHIMAN, R. Electroencephalogram channel selection based on pearson correlation coefficient for motor imagery-brain-computer interface. *Measurement: Sensors.* 2023, vol. 25, p. 100616. DOI: 10.1016/j.measen.2022.100616. ISSN 2665-9174. Available at: https://www.sciencedirect.com/science/article/pii/S2665917422002501.

[49] PERRIN, S., JAY, S., VINCENT, G., SPRAJCER, M., LACK, L. et al. Waking qEEG to assess psychophysiological stress and alertness during simulated on-call conditions. *International Journal of Psychophysiology.* 2019, vol. 141, p. 93–100. DOI: 10.1016/j.ijpsycho.2019.04.001. ISSN 0167-8760. Available at: https://www.sciencedirect.com/science/article/pii/S0167876018310924.

[50] PISNER, D. A. and SCHNYER, D. M. Chapter 6 - Support vector machine. In: MECHELLI, A. and VIEIRA, S., ed. *Machine Learning.* Academic Press, 2020, p. 101–121. DOI: 10.1016/B978-0-12-815739-8.00006-7. ISBN 978-0-12-815739-8. Available at: https://www.sciencedirect.com/science/article/pii/B9780128157398000067.

[51] RASTGOO, M. N., NAKISA, B., MAIRE, F., RAKOTONIRAINY, A. and CHANDRAN, V. Automatic driver stress level classification using multimodal deep learning. *Expert Systems with Applications.* 2019, vol. 138, p. 112793. DOI: https://doi.org/10.1016/j.eswa.2019.07.010. ISSN 0957-4174. Available at: https://www.sciencedirect.com/science/article/pii/S0957417419304890.

[52] ROY, S., ISLAM, M., YUSUF, M. S. U. and JAHAN, N. EEG based stress analysis using rhythm specific spectral feature for video game play. *Computers in Biology and Medicine.* 2022, vol. 148, p. 105849. DOI: https://doi.org/10.1016/j.compbiomed.2022.105849. ISSN 0010-4825. Available at: https://www.sciencedirect.com/science/article/pii/S0010482522006023.

[53] SAFFARI, F., NOROUZI, K., BRUNI, L. E., ZAREI, S. and RAMSØY, T. Z. Impact of varying levels of mental stress on phase information of EEG Signals: A study on the frontal, Central, and parietal regions. *Biomedical Signal Processing and Control.* 2023, vol. 86, p. 105236. DOI: 10.1016/j.bspc.2023.105236.

[54] SAME, M. H., GANDUBERT, G., GLEETON, G., IVANOV, P. and LANDRY, R. Simplified welch algorithm for spectrum monitoring. *Applied Sciences.* 2020, vol. 11, no. 1, p. 86. DOI: 10.3390/app11010086.

[55] SANFORD, L. D., WELLMAN, L. L., ADKINS, A. M., GUO, M.-L., ZHANG, Y. et al. Modeling integrated stress, sleep, fear and neuroimmune responses: Relevance for understanding trauma and stress-related disorders. *Neurobiology of Stress.* 2023, vol. 23, p. 100517. DOI: 10.1016/j.ynstr.2023.100517.

[56] Sanggarini, H., Wijayanto, I. and Hadiyoso, S. Hjorth Descriptor as Feature Extraction for Classification of Familiarity in EEG Signal. In: *2019 International Conference on Information and Communications Technology (ICOIACT)*. 2019, p. 306–309. DOI: 10.1109/ICOIACT46704.2019.8938532.

[57] Seo, S. H. and Lee, J.-T. Stress and EEG. In:. March 2010. DOI: 10.5772/9651. ISBN 978-953-307-068-1.

[58] Shon, D., Im, K., Park, J.-H., Lim, D.-S., Jang, B. et al. Emotional stress state detection using genetic algorithm-based feature selection on EEG Signals. *International Journal of Environmental Research and Public Health*. 2018, vol. 15, no. 11, p. 2461. DOI: 10.3390/ijerph15112461.

[59] Singh, A. K. and Krishnan, S. Trends in EEG signal feature extraction applications. *Frontiers in Artificial Intelligence*. 2023, vol. 5. DOI: 10.3389/frai.2022.1072801. ISSN 2624-8212. Available at: https://www.frontiersin.org/articles/10.3389/frai.2022.1072801.

[60] Sundaresan, A., Penchina, B., Cheong, S., Grace, V., Valero Cabré, A. et al. Evaluating deep learning EEG-based mental stress classification in adolescents with autism for breathing entrainment BCI. *Brain Informatics*. december 2021, vol. 8. DOI: 10.1186/s40708-021-00133-5.

[61] Swanson, L. W. *Brain Architecture: Understanding the Basic Plan*. 1st ed. Oxford University Press, 2002. ISBN 978-0195105056.

[62] Tan, H. H. and Lim, K. H. Vanishing Gradient Mitigation with Deep Learning Neural Network Optimization. In: *2019 7th International Conference on Smart Computing & Communications (ICSCC)*. 2019, p. 1–4. DOI: 10.1109/ICSCC.2019.8843652.

[63] Tong, R. L., Kahn, U. N., Grafe, L. A., Hitti, F. L., Fried, N. T. et al. Stress circuitry: Mechanisms behind nervous and immune system communication that influence behavior. *Frontiers in Psychiatry*. 2023, vol. 14. DOI: 10.3389/fpsyt.2023.1240783.

[64] Torres García, A. A., Mendoza Montoya, O., Molinas, M., Antelis, J. M., Moctezuma, L. A. et al. Pre-processing and feature extraction. *Biosignal Processing and Classification Using Computational Learning and Intelligence*. 2022, p. 59–91. DOI: 10.1016/b978-0-12-820125-1.00014-2.

[65] Vanhollebeke, G., De Smet, S., De Raedt, R., Baeken, C., Mierlo, P. van et al. The neural correlates of Psychosocial Stress: A systematic review and meta-analysis of spectral analysis EEG studies. *Neurobiology of Stress*. 2022, vol. 18, p. 100452. DOI: 10.1016/j.ynstr.2022.100452.

[66] Veen, F. van der, Molen, M. van der, Molen, M. van der and Franken, I. Thumbs up or thumbs down? Effects of neuroticism and depressive symptoms on psychophysiological responses to social evaluation in healthy students. *Cognitive, Affective, & Behavioral Neuroscience*. may 2016, vol. 16. DOI: 10.3758/s13415-016-0435-2.

[67] W Azlan, W. A. and Low, Y. F. Feature extraction of electroencephalogram (EEG) signal - A review. In: *2014 IEEE Conference on Biomedical Engineering and Sciences (IECBES)*. 2014, p. 801–806. DOI: 10.1109/IECBES.2014.7047620.

[68] *What Is a Machine Learning Pipeline?* [online]. 2023 [cit. 2023-11-20]. Available at: https://www.ibm.com/topics/machine-learning-pipeline.

Appendix A

Contents of the Included Storage Media

Directory	Description
objects/	Directory with `.py` scripts implementing classes.
SAM40/	Directory with `data_augmentation.py` script where should be placed the SAM40 dataset, specifically `scales.xls` and `filtered_data` folder.
docs/	Directory with `.tex` files and other content of the thesis.

Table A.1: Directories of the included storage media.

The root directory also contains `__main__.py` script, `Makefile`, `README.md`, `requirements.txt` and `thesis.pdf` with the bachelor's thesis text.