Detection of interictal epileptiform discharges (IED) in EEG using Machine Learning

Master thesis

submitted on: September 21, 2022

Westfälische Wilhelms-Universität Münster

Name:	Simone Wlotzka
Supervisor:	Prof. Dr. Wolters
Co-Supervisor:	Prof. Dr. Wirth

Contents

0 Introduction

Epilepsy affects about 6 million people worldwide [BSea09]. Beyond the diagnosis, more than five percent of people across the globe experience an epileptic seizure once in their life [BSea09, p.3]. An electroencephalography (EEG) can identify patterns in the brain of patients that accrue typical in epilepsy. These patterns are caused by activity in the affected areas of the brain and thus provide an indication of the presence of the disease.

The standard procedure in diagnostics involves an EEG, which results in a high number of such recordings. The analysis of the collected data is still a costly time investment. An interruption of the underlying background activity in the brain has to be detected, which is also of minimal duration. In addition to the time required, the expertise of the physician is essential for identifying these patterns [KABea17]. Even for experienced staff in epilepsy diagnosis, the agreement on the interpretation between different specialists is comparatively low [KABea17]. Though there exist approaches to define an epileptic pattern, individual subjective thresholds seem to be more decisive than the morphological appearance [JHea19].

Machine learning algorithms are already achieving strong results in classification studies and provide new methods to analyse and predict data. They improve image interpretation methods and have been used in various medical applications.

Therefore, the implementation of these algorithms in the setting of interictal epileptiform discharges (IED) detection seems to be a promising approach. A classification algorithm in the evaluation of EEG recordings would provide a major preliminary work for the physician. In addition to the time savings, a good basis for classification could be established in order to obtain a standardised evaluation scheme.

The aim of this thesis is to implement a machine learning algorithm that performs a binary classification of IEDs and no IEDs in an EEG recording. For this purpose, it will be tested on simulated IED data and finally operate on patient data. In contrast to the literature [LTCvPea19], it will also be investigated whether a less deep network can be useful for this problem as a decision maker. In this way, the time required to train such an algorithm is reduced.

In order to create an machine learning algorithm of an EEG, various principles are first discussed. The underlying medical condition and the difficulty of recognising typical patterns of epilepsy are explained in chapter one. Chapter two deals with the different functional principles of an EEG, dives deep into the pre-processing steps of recorded data and defines the emergence of conspicuous patterns. Subsequently, the theoretical basics in form of mathematical foundations of machine learning algorithms will be explained in chapter three. The mechanisms of a Convolutional Neural Network (CNN) for classifying images are then described in chapter four, so that an own network can be set up. The collected and simulated data is described in chapter five. Chapter six discusses the constructed network which is being applied to the aforementioned data. Finally, a presentation of results and an evaluation concern chapters seven and eight.

1 Fundamentals

The basis of human action, feeling and thinking is built on our nervous system. This can be broadly visualised as a complex network of connections that transmit information through electrical impulses. The study of the human brain is a necessary prerequisite for understanding the basic mechanisms of the EEG and the diagnosis of epilepsy. These foundations will be established by a brief insight into neuroanatomy in the following chapter. For this purpose, sources [Tre22, p.1-20] and [ED11, p.1-9] were used.

1.1 Basics of neuroanatomy

1.1.1 Structure and function of neurons

The nervous system is fundamentally divided into peripheral and central nervous system (CNS). The latter is subdivided into the brain and the spinal cord. The nervous system is made up of nerve tissue, which, in addition to glial cells, consists of nerve cells. Neurons form the fundamental element of these cells.

The main components of a neuron are the cell body (soma), the dendrites and the axon. The soma is composed of an inner cell nucleus and is linked to tree-like branched dendrites. These extensions are responsible for receiving impulses which are delivered from the soma through the axon. The junction between axon and soma is called axon hillock. It functions as a control centre for incoming impulses and can release an action potential. An action potential refers to the short-term collapse of the membrane potential. A neuron has a negative intracellular membrane potential compared to the extracellular space. Such an action potential is initiated if the sum of the incoming signals is above a certain threshold value. This results in depolarisation and subsequently a rapid repolarisation.

Nodular branches are found at the other end of the axon and are known as terminal buttons. Together with the cell membrane of the subsequent cell and the gap in between, they form a synapse. The transmission of impulses from one neuron to another takes place at the synapses.

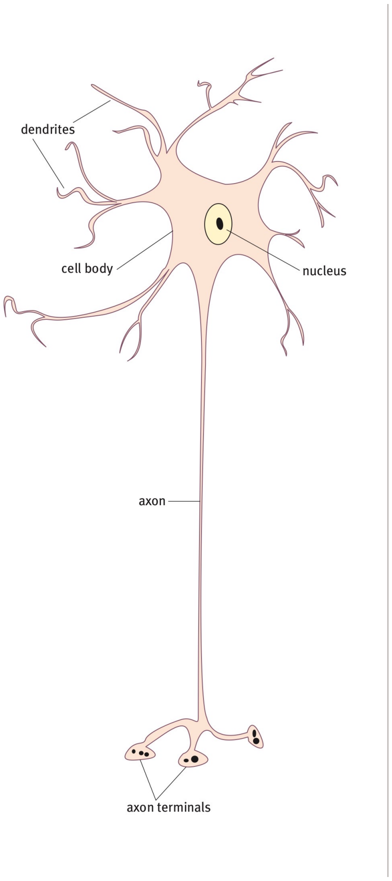


Figure 1 Components of a neuron [WKea10, p.3]

1.1.2 Transmission of excitation

A transmitter is released at the synapses during excitation. This transmitter causes the subsequent nerve cell to open or close its ion channels. This synaptically induced membrane potential change can have an excitatory or inhibitory effect. More precisely, a transmitter can lead to a negativisation of the extracellular space in which a sodium or calcium ion influx is generated or the potassium outflow is reduced. This leads to depolarisation of the dendrites and thus to a decrease in the positive charge on the surface of the cell. In addition, an influx of negatively charged chloride ions or an outflow of positively charged potassium ions can also occur. It is known as the positivation of the extracellular space.

1.2 Epilepsy Diagnosis

Having established some basics of neuroanatomy, a brief introduction to the clinical picture of epilepsy will follow. Epilepsy is a disease of the brain. It affects approximately 60 million people worldwide, making it the second most common neurological disease in the world. In order to be diagnosed with epilepsy, the affected person must have experienced at least two seizures within interval of at least 24 hours. More information on this topic can be found in the sources [MG14, p.3-5] and [ED11, Chapter 12] used for the next sub-chapters. The last two sub-sections are based on [BSea09, p. 29-36] and [BSea09, p. 38-41; 100-109].

1.2.1 Seizure

Epileptic activity, pathophysiologically, occurs due to neuronal hyperexcitability or hypersynchrony. During hyperexcitability, the neuronal response to a stimulus is heightened. Subsequently, several action potentials are fired by the cell in response to a synaptic input. Hypersynchrony works differently, with increased local neuronal firing in the cortex. In this case, the cells fire in close temporal and spatial proximity. These processes can be summarised under the name epileptogenesis, on the basis of which an epileptic seizure can occur. This leads to a temporary interruption of brain functions. The diagnostic picture of such a seizure depends on the exact location of the firing cortical neurons and their spreading pattern. For example, motor functions or perception may be disturbed, while a combination of several disturbances is also possible. More than five precent of people worldwide experience a seizure once in their life.

1.2.2 Epilepsy

The recurrent and unprovoked occurrence of such seizures is described as epilepsy. The origin of this diagnosis may be a chronic pathological condition or a genetic susceptibility. Seizures are grouped according to the site of onset and pattern of spread, and are therefore referred to as focal or generalised epilepsy. A focal (or partial) seizure has its origin in a localised region of the brain.In this case, a spreading pattern is found locally or extended to the whole brain, which is defined as secondary generalization. During the seizure, impairments related to functions in this area can be observed.

These disturbances have the potential to be severe enough to cause unconsciousness or unresponsiveness, which is known as a complex seizure type. If these functions remain undisturbed, the seizure is classified as simple. In a generalised seizure, the discharges start in both hemispheres simultaneously. The affected region spreads to the reciporal thalamocortical connections. As many parts are involved, the clinical picture is broad and can range from brief loss of responsiveness to severe convulsions and loss of consciousness. The seizure types and a number of other characteristics such as age of onset, precipitating factors, etc. are summarised under the term epilepsy syndrome. Seizures are the main feature of this syndrome.

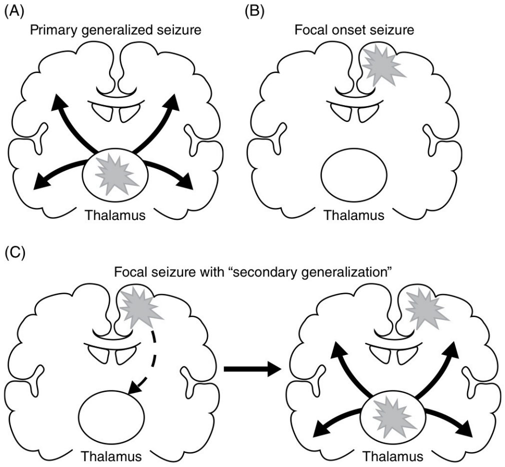

(A)
Primary generalized seizure
Thalamus
(B)
Focal onset seizure
Thalamus
(C)
Focal seizure with "secondary generalization"
Thalamus
Thalamus

Figure 2 Illustration of the spreading pattern in the brain [MG14, p.4].

1.2.3 Diagnosis

A diagnostic identification of epilepsy requires a number of steps. There are numerous other diseases that are similar to an epileptic seizure. In order to avoid unnecessary and ineffective therapy, a clear classification of the seizure should be performed and other clinical pictures should be excluded where possible. Difficulties can also arise if a patient suffers from both non-epileptic seizures and epileptic seizures. Differentiation of the seizures is particularly important for targeted therapy. In addition, a distinction should be made between provoked and unprovoked seizures. Approximately 50 percent of all seizures are caused by acute sympthomatics of the CNS, such as those resulting from an acute infection or trauma. In order to be able to diagnose an epileptic seizure clearly, a number of parameters should be observed. An important factor is the detailed description of the event by the patient and also by witnesses. This description can already provide initial information about the affected brain regions. In this context, a detailed examination of the patient's medical, social and drug history is necessary. Further questions about the potential cause of the seizure may also be addressed. Blood tests and an electrocardiogram (ECG) can provide deeper information about triggers, such as cardiac arrhythmias. To distinguish between epileptic and psychogenic seizures, an electroencephalogram (EEG) is used. Using electrodes, it records patients' brain waves. The recorded EEG of epilepsy patients shows a typical pattern, which we will be discussed in more detail in the next chapter. In addition, a magnetoencephalogram (MEG) can be used to record abnormalities that are not detectable by the EEG. In the event of a seizure, a video EEG provides information about the neurological aspects of the disorder and is usually used prior to epilepsy surgery. Finally, neuroimaging techniques, such as MRI, are an important tool to characterise structural changes in the brain. After a range of examinations, it should be assessed whether the physiological abnormalities and the patient history in summary meet the criteria of epilepsy, in order to start with the most precise treatment available.

1.2.4 Treatment

After an extensive diagnosis, the decision regarding the treatment is taken. The aim is to allow the patient to live a normal life in society as best as possible while minimising the side effects of the treatment. A preliminary choice is the selection of an antiepileptic drug (AED). Monotherapy with an AED has a about 50% chance of success after the first drug. If this is not successful, a second drug is taken, in which case the chances of success increase by 9 percent. After this, the usage of an additional AED will not significantly increase the probability of being seizure free. In addition, fewer side effects can be expected through monotherapy than with a combination of several drugs. For this reason, a significant focus should be placed on the choice of the first AED and this should be determined in dependence on seizure type and epileptic symptoms. Other factors in the choice include patient-specific characteristics such as age and gender. A successful choice of medication is intended to be lifelong. For this reason, the impact of side effects should also be given considerable consideration. Patients with drug-resistant seizures have an increased mortality rate compared to those who successfully take a medication, and impairments may increase over time [BSea09, p.100-109]. After unsuccessful medication, a possible resection of the epilepsy provoking area in the brain is considered. Prior to surgery, the patient undergoes a detailed examination, including neuroimaging. The effects of surgery should not cause further neurological or cognitive deficits. If these side effects are expected to occur with a high probability, a number of other methods will be considered. One option is vagus nerve stimulation (VNS), in which a generator is implanted in the patient's left chest. The generator then sends impulses to the vagus nerve. A reduction in the severity and frequency of seizures can be expected. Another possible treatment is the deep brain stimulation. Electrical signals are emitted into the brain using this method. So far, only short-term relief has been observed.

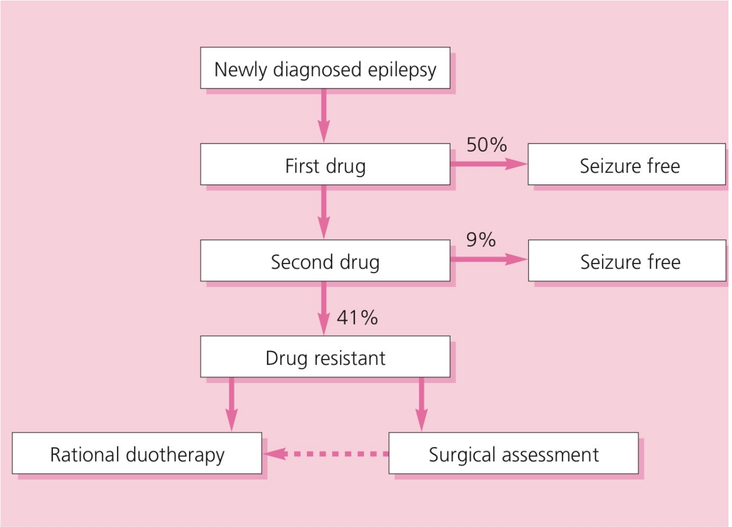


Figure 3 Illustration of the steps following after an epilepsy diagnosis [BSea09, p.40].

2 EEG

The following chapter is based on book [ED11]. The introduction and the first section rely on the first two chapters of the book. For the 'Pre-Processing' section, chapters two and three of the book are used. Finally, the description of epilepsy follows chapter 12 and the paper [KABea17].

An electroencephalography (EEG) can be used to record, amplify and document electrical potentials of the brain. In medicine and research, EEG is used in many ways due to its cost-effectiveness, non-invasiveness and portability. For example, clinical epilepsy diagnostics use it to detect an epileptogenic lesion via subsequent source analysis or to register seizures that are not recognised by patient self-awareness. For this reason, it provides indispensable information about the disease pattern of the patient. In order to understand the exact application of an EEG in epilepsy research, it is necessary to understand the basic mechanisms of the device.

2.1 Functionality and structure

The basic principle of EEG is to measure the electrical potential fields generated in the brain on the surface of the head using electrodes and to display them graphically along a time axis. The underlying neuronal structures of this principle have already been explained in more detail in chapter 1 and will now be applied to the EEG.

The mechanisms described in the mentioned chapter lead to a synaptically induced membrane potential change, which generates a current flow that is measured by the EEG. The recorded electrical activity depends on the type of synaptic potentials and the localization as well as position of the synapses in the cortex. One can distinguish between current sink and source. A sink describes the place where a relative loss of positive charges is found in the extracellular space. The compensation of these charge changes in neighbouring areas occurs through volume flows in the cells. A source is then defined as the location of the positive compensation of the charge change. Furthermore, the relative position of the cells to each other plays an essential role in the recording of the measurement. If there is a parallel arrangement of the neurons, the EEG sums up the recorded fields, known as open field. The opposite is true for the closed field. There, the neurons are arranged randomly, which includes a symmetrical non-parallel arrangement. A symmetrical alignment can lead to cancellation of the potentials in the EEG, so that it is not registered (as illustrated in figure 4). A further distinction is the relative position in the cortex. When the neurons are in a perpendicular orientation with respect to the brain surface, a negative field on the head surface is registered by the EEG, known as radial dipole. If the neurons are aligned parallel to the brain surface, the source is located between the potential maxima. In this case, a so-called tangential dipole is given.

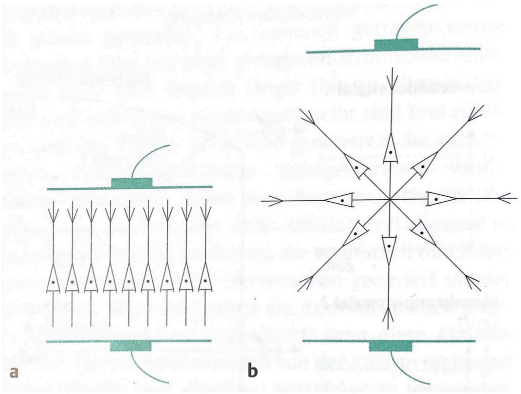


Figure 4 Example of an open (a.) and closed (b.) field [ED11, p.4].

EEG measures the fields described above. To better understand the exact signal processing, a brief look at the individual components of an EEG needs to be given. The basic components of an EGG are the electrodes and an amplifier.

Typically there are 21 electrodes, whereas in research up to 256 are used. They are evenly distributed on the surface of the head, usually according to the 10-20 system, and named depending on their position on the surface of the head. To better register the electrical activity, an electrolyte-containing paste or liquid is applied between the metal electrodes and the scalp. Through these electrodes EEG typical amplitudes of only 20-50 μV can be recorded. However, the combination of metal and electrolyte causes the effect of a battery, so that a direct current (DC) voltage is generated. This overlaps with the actual measured signal. Furthermore, distortions of the measurement by external sources, such as a power supply, cannot be sufficiently shielded, so that, for example, a 50 Hz alternating voltage influences the signal. In order to counteract this, the electrodes are connected to a differential amplifier. This amplifier is responsible for increasing the small amplitudes while simultaneously suppressing almost identical interfering signals. The measured signals are usually displayed on a computer monitor and processed afterwards.

2.2 Pre-processing

During the measurement, multiple artefacts may occur. Generally, a distinction is made between technical and biological artefacts. Both categories overlay the actual measured signal, making the interpretation of the EEG more difficult. However, a number of features allow artefacts to be recognized. For example, if an electrode does not lie completely on the scalp, the technical interference can be distinguished from the normal EEG signal because it is only visible on one channel. Eye movements (the eye forms an electrical dipole) are continuously visible on the EEG, but can be classified as such by topographical evaluation. If artefacts have become recognisable as such, the next step is usually to remove them through various pre-processing steps. Classic EEG devices still have built-in filters and montage methods, unlike newer products. New devices thus offer the advantage of not having the measured signal pre-determined and being able to use different filters only in the digital transformation. To transform the signal to a digital medium, an analogue-to-digital-converter is needed. The limitation to accurately represent the signal arises from the predetermined number of grid points for recording the amplitudes. Measured signal waveforms are transferred to a grid at discrete points in time so that recorded amplitudes are transferred to the nearest numerical values available on the grid. The amount of amplitude sampling per second is called the sampling rate and given in Hz. Depending on this size, the storage space required for the measured file increases. Therefore, in the course of processing, it may sometimes make sense to reduce the size of the file by downsampling it. Using a digitally applied filter, a weighted averaging of successive samples takes place, which are then multiplied by a factor, depending on the choice of filter. High-pass and low-pass filters allow a reduction of artefacts if they are used purposefully. This involves attenuating the measured signal above (low-pass filter) or below a certain frequency, relative to the signal component of a fixed frequency. The aforementioned interference in the signal caused by a 50 Hz AC voltage can be suppressed by a notch filter. However, when used incorrectly, filters cancel clinically significant signal. For example, a high-frequency filter between 10-15 Hz can cause epilsietypic patterns to be mistaken for eye movements [ED11, p.73].

Another option for pre-processing an EEG signal is to choose a so called montage for the electrodes.The purpose of pre-processing is to separate clinical signals from technical or biological artefacts and to classify the field potential generated in the patient's brain. Montages describe the representation of voltage differences between electrodes and allow certain clinical patterns to be highlighted accurately. Generally, a distinction is made between referential and bipolar montages. If a fixed electrode is used as a common reference for all electrodes, the potential field itself is mapped. That is described as reference montage. For the actual representation of the potential field, the selected reference electrode itself should not lie in the active brain area. The bipolar montage describes the visualisation of the potential differences of neighbouring electrodes (see figure 6). This leads to the representation of the gradient of the potential field. The resulting subtraction of the signal of two electrodes is defined as channel. Classical montages can be modified greatly and offer advantages and disadvantages in reading the EEG signal depending on the diagnostic background of the patient.

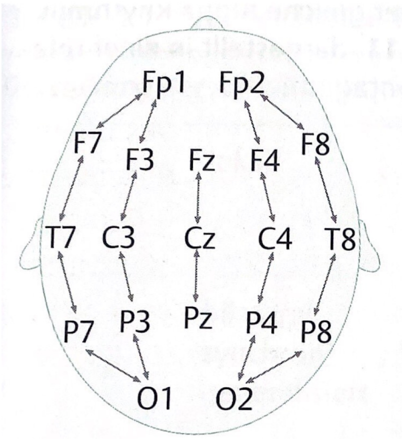


Figure 5 Representation of the bipolar longitudinal montage. When processing an EEG in digitally, it is often abbreviated as a 'longitudinal' montage. There are different variations of this reference method [ED11, p.23].

As mentioned above, the EEG is used in various situations in everyday clinical practice. In this thesis, the application of EEG in epilepsy diagnostics will be considered.

2.3 Epilepsy

EEG can be used to evaluate the diagnosis of epilepsy and is one of the most important methods. A number of epilepsy-typical waveforms may appear in patients with epilepsy, but they are not a necessary condition for diagnosis. Nevertheless, these temporal appearance potentials correlate approximately 90% with the presence of focal temporal epilepsy. Conversely, the patterns are only typical for epilepsy but not specific. Meaning they can also occur in healthy patients. Overinterpretation of EEG is among the most common reasons of wrong diagnosis. Thus, between 25 and 30% of drug resistant patients presented in epilepsy centers actually do not suffer from epilepsy. This leads to serious consequences such as the occurrence of medication side effects.

For successfully diagnosed patients, however, the EEG can be extremely helpful, such as for preoperative source analysis. In addition, the number of interictal potentials is a measure for assessing the effectiveness of an antiepileptic drug. The fewer of these potentials are found in the EEG, the less likely a seizure is to occur. Patients who are dependent on drugs that cause severe side effects have thereby the possibility to at least lower the dosage.

2.3.1 Epilepsy typical potentials

The EEG can record interictal and ictal typical patterns. Ictal epilepsy typical activities may be difficult to read due to abrupt onset of tonic postural seizures, as they are thus overlapped. Also, they may be located in specific frontal brain regions, so that electrodes can not record them. Interictal epilepsy potentials are measurable much more frequently and can be used significantly for the above-mentioned diagnostic procedures.

Visual analysis of the EEG is still considered the standard in order to recognise patterns . Yet the classification of patterns is not unambiguous. There are no quantifiable characteristics, so that experience of the rater plays an essential role in the assessment of an EEG. Nevertheless, there are several attempts to define qualitative criteria for recognition. In [ED11][p.125], interictal epilepsy potentials are described as combinations of spikes and waves. A

spike is considered to have the following properties: It has a between 20 and 70 milliseconds long sharp asymmetric morphology and is bi- or triphasic. Further, the amplitude is variable, but the main phase is usually negative. Another important factor is the interruption of the basic activity caused by the pattern. A sharp wave also satisfies the criteria of a spike, but lasts for approx. between 70 and 200 milliseconds. Combinations of these two patterns called spike-wave-complexes occur with different frequencies, such as the typical 3-Hz-spike-wave-complex, where the wave decays slowly. Polyspikes have more than three phases and generate a polyspike-wave-complex when combined with a slow wave.
The occurrence of these properties in combination with a recognisable electric field distribution starting from a cerebral generator defines an IED according to [ED11, p.125]. This definition is consistent with the International Federation operational definition [KABea17].

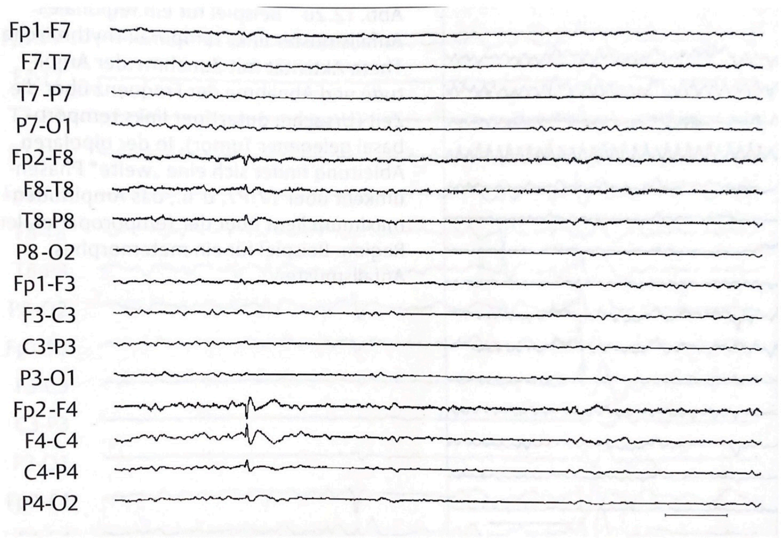
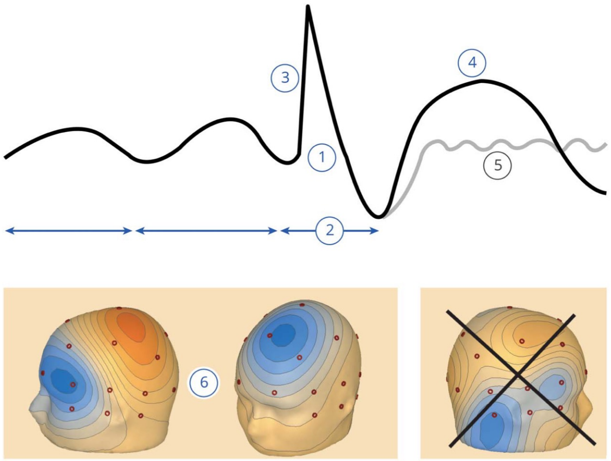


Figure 6 Illustration of a sharp spike-wave complex. Recorded from a patient with frontal epilepsy [ED11, p.133].

2.3.2 Quality of visual IED detection

Based on the aforementioned problem of interpreting an EEG, this can lead to differently classified patterns depending on the physician. In a study from 2020, seven raters were asked to mark IEDs in EEGs of 100 patients with and without epilepsy. Their markings were finally compared with those of two experts. The IFCN criteria were applied in sensor space according to figure 7. Considering all five criteria visible in the EEG, the raters achieved a specificity of 95.65% and sensitivity of 81.48%. The accuracy was 88%. The interrater agreement was moderate when classifying IED using all five criteria. The difficulty of pattern recognition lies not only in recognising the regular appearance of a spike in the irregular basic activity, but also in not confusing it with other patterns. Some activities have an epileptic appearance but are not associated with epilepsy, such as wicket spikes [ED11, p.100].



(1) Di- or tri-phasic wave with pointed peak; (2) different wave duration than the ongoing background activity; (3) asymmetry of the waveform; (4) followed by a slow after-wave; (5) the background activity is disrupted by the presence of the IEDs; and (6) voltage map with distribution of the negative and positive potentials suggesting a source in the brain corresponding to a radial, oblique, or tangential orientation of the source.^{23} For further details, see the Methods section. The voltage maps in window (6) show a tangential orientation (source in the left middle frontal gyrus) and a radial orientation (source in the left superior frontal gyrus); the irregular distribution of the potentials in the last voltage map does not imply a source in the brain (it was an artifact with sharp morphology) and does not fulfill this criterion; negative potentials are in blue, and positive potentials are in red. IED = interictal epileptiform discharge; IFCN = International Federation of Clinical Neurophysiology.

Figure 7 [KABea17]

3 Machine learning

The previous chapters have given a medical introduction to epilepsy diagnosis. In the following, the mathematical foundations for the development of a machine learning algorithm will presented.
If not stated otherwise, the lecture [JKea21] was used as source. The two introductory chapters refer to chapter 2.1 of the lecture. The definition of the loss function and the gradient descent follows 1.1. and 4.1 course chapter.

3.1 Vectorized description

In this section we want to give a general introduction to the mathematical point of view of deep learning. Given $m, n \in \mathbb{N}$ we define a function $\mathcal{A} : \mathbb{R}^n \to \mathbb{R}^m$ by $\mathcal{A}(x) = Wx + b$ with $x \in \mathbb{R}^n, W = (w_{ij}) \in \mathbb{R}^{n\times m}$ and $b \in \mathbb{R}^m$.

3.1 Definition (affine function) Given $\vartheta, m, n \in \mathbb{N}, s \in \mathbb{N}_0, \Theta = (\Theta_1, ..., \Theta_\vartheta) \in \mathbb{R}^\vartheta$ and $\vartheta \geq s + mn + m$ we can define a function $\mathcal{A}^{\Theta,s}_{m,n} : \mathbb{R}^n \to \mathbb{R}^m$ by

$$\begin{aligned}\mathcal{A}^{\Theta,s}_{m,n}(x) &= \begin{pmatrix} \Theta_{s+1} & \Theta_{s+2} & \cdots & \Theta_{s+n} \\ \Theta_{s+n+1} & \Theta_{s+n+2} & \cdots & \Theta_{s+2n} \\ \Theta_{s+2n+1} & \Theta_{s+2n+2} & \cdots & \Theta_{s+3n} \\ \vdots & \vdots & \ddots & \vdots \\ \Theta_{s+(m-1)n+1} & \Theta_{s+(m-1)n+2} & \cdots & \Theta_{s+mn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{pmatrix} + \begin{pmatrix} \Theta_{s+mn+1} \\ \Theta_{s+mn+2} \\ \Theta_{s+mn+3} \\ \vdots \\ \Theta_{s+mn+m} \end{pmatrix} \\ &= \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_m \end{pmatrix}\end{aligned}$$

Hence we get

$$y_j = \sum_{i=1}^{n} x_i \Theta_{s+(m-1)n+i} + \Theta_{s+mn+j}$$

for $j = 1, \cdots, m$. We call $\mathcal{A}$ an *affine function associated to* (Θ, s).

3.2 Example Let $\vartheta = 6, m = 1, n = 2, s = 2$ and $\Theta = (1, 2, -3, 4, 2, 0)$ be given. Than it holds that $6 \geq 2+2+2$ and we have a function $\mathcal{A}^{(1,2,-3,4,2,0),2}_{1,2}$:

$\mathbb{R}^2 \to \mathbb{R}$ as in 3.1 with

$$
\begin{aligned}
\mathcal{A}^{(1,2,-3,4,2,0),2}_{1,2}\begin{pmatrix}1\\2\end{pmatrix} &= (-3,4)\begin{pmatrix}1\\2\end{pmatrix}+4\\
&= 5+4\\
&= 9.
\end{aligned}
$$

3.3 Definition (Vectorized description of artificial neural networks) Now let $\psi_k : \mathbb{R}^{l_k} \to \mathbb{R}^{l_k}$ be functions with $L \in \{2,3,\cdots\}, k \in \{1,2,\cdots,L\}$ and $l_0, l_1, ..., l_L \in \mathbb{N}$. Furthermore let ϑ satisfy

$$\vartheta \geq \sum_{k=1}^{L} l_k(l_{k-1}+1)$$

and $\Theta \in \mathbb{R}^{\vartheta}$. Then we call the composition

$$
\begin{aligned}
\mathcal{N}^{\Theta,l_0}_{\psi_1,\psi_2,\cdots,\psi_L} : \mathbb{R}^{l_0} &\to \mathbb{R}^{l_L}\\
x &\mapsto (\psi_L \circ \mathcal{A}^{\Theta,\sum_{k=1}^{L-1} l_k(l_{k-1}+1)}_{l_L,l_{L-1}} \circ \psi_{L-1} \circ \mathcal{A}^{\Theta,\sum_{k=1}^{L-2} l_k(l_{k-1}+1)}_{l_{L-1},l_{L-2}}\\
&\quad \cdots \circ \psi_2 \circ \mathcal{A}^{\Theta,l_1(l_0+1)}_{l_2,l_1} \circ \psi_1 \circ \mathcal{A}^{\Theta,0}_{l_1,l_0})(x) \qquad (1)
\end{aligned}
$$

the *realization of the fully connected feedforward artificial neural network associated to* Θ with $L-1$ layers with dimensions $(l_0, l_1, \cdots, l_L)$ and activation functions $(\psi_1, \psi_2, \cdots, \psi_L)$, input dimension l_0, output dimension l_L and output activation function ψ_L.

We will now specify the function defined in the introduction to look at it in the context of artificial neural networks.

3.4 Definition (Weights and biases of artificial neural networks) Let $L \in \{2,3,\cdots\}$, $\nu_0, \nu_1, \cdots, \nu_{L-1} \in \mathbb{N}_0$, $l_0, l_1, \cdots, l_l \in \mathbb{N}$, $\Theta \in \mathbb{R}^{\nu}$, ϑ as above, satisfy

$$\nu_k = \sum_{i=1}^{k} l_i(l_{i-1}+1),$$

for all $k \in \{0, 1, \cdots .L-1\}$. Then for all $k \in \{1,2,\cdots,L\}$ we have an affine function

$$
\begin{aligned}
\mathcal{A}^{\Theta,\nu_{k-1}}_{l_k,l_{k-1}} : \mathbb{R}^{l_{k-1}} &\to \mathbb{R}^{l_k}\\
x &\mapsto W_k x + b_k
\end{aligned}
$$

where $W_k \in \mathbb{R}^{l_k \times l_{k-1}}$ and $b_k \in \mathbb{R}^{l_k}$ satisfy for all $k \in \{1, 2, \cdots, L\}$ that

$$W_k = \underbrace{\begin{pmatrix} \Theta_{\nu k-1+1} & \Theta_{\nu_{k-1}+2} & \cdots & \Theta_{\nu_{k-1}+l_{k-1}} \\ \Theta_{\nu k-1+l_{k-1}+1} & \Theta_{\nu_{k-1}+l_{k-1}+2} & \cdots & \Theta_{\nu_{k-1}+2l_{k-1}} \\ \Theta_{\nu_{k-1}+2l_{k-1}+1} & \Theta_{\nu_{k-1}+2l_{k-1}+2} & \cdots & \Theta_{\nu_{k-1}+3l_{k-1}} \\ \vdots & \vdots & \ddots & \vdots \\ \Theta_{\nu_{k-1}+(l_k-1)l_{k-1}+1} & \Theta_{\nu_{k-1}+(l_k-1)l_{k-1}+2} & \cdots & \Theta_{\nu_{k-1}+l_k l_{k-1}} \end{pmatrix}}_{\text{weights}}$$

and

$$b_k = \underbrace{\begin{pmatrix} \Theta_{\nu_{k-1}+l_k l_{k-1}+1} \\ \Theta_{\nu_{k-1}+l_k l_{k-1}+2} \\ \Theta_{\nu_{k-1}+l_k l_{k-1}+3} \\ \vdots \\ \Theta_{\nu_{k-1}+l_k l_{k-1}+l_k} \end{pmatrix}}_{\text{biases}}.$$

Let $\psi_k : \mathbb{R}^{l_k} \to \mathbb{R}^{l_k}$, $k \in \{1, 2, \cdots, L\}$, be functions. Then it holds that

$$\mathcal{N}^{\Theta,l_0}_{\psi_1,\psi_2,\cdots,\psi_l} = \psi_L \circ \mathcal{A}^{\Theta,\nu_{L-1}}_{l_L,l_{L-1}} \circ \psi_{L-1} \circ \mathcal{A}^{\Theta,\nu_{L-2}}_{l_{L-1},l_{L-2}} \circ \cdots \circ \psi_1 \mathcal{A}^{\Theta,\nu_0}_{l_1,l_0}.$$

3.2 Activation function

There are many ways to choose an activation function. For an image classification the ReLU function is appropriate [HH18, p. 22]

3.5 Definition (ReLU) We define the *rectified linear unit function* as $ReLU : \mathbb{R} \to \mathbb{R}$ which satisfies for all $x \in \mathbb{R}$ that

$$ReLU(x) = max\{0, x\}.$$

To describe deeper networks, we define the concept of a multidimensional function and apply it to activation functions:

3.6 Definition (Multidimensional function) Let $\psi : \mathbb{R} \to \mathbb{R}$ be a function. Then we can define by $\mathcal{M}_{d,\psi}$ with $d \in \mathbb{N}$ the $d-$dimensional version of ψ via:

$$\begin{aligned} \mathcal{M}_{d,\psi} : \mathbb{R}^d &\to \mathbb{R}^d \\ x &\mapsto (\psi_1(x), \psi_2(x), \cdots, \psi_d(x)). \end{aligned}$$

3.7 Example Let $d \in \mathbb{N}$. For $\psi = ReLU$ we set $ReLU_d := \mathcal{M}_{ReLU,d}$ the $d-$dimensional $ReLU$ function with $ReLU_d : \mathbb{R}^d \to \mathbb{R}^d$.

3.8 Example (An artificel neural network using ReLU) We will now explain the mechanisms of a neural network in more detail using an example. In doing so, we will also discuss the motivation behind the development of the algorithm.
Let $\mathcal{I}, \mathcal{H} \in \mathbb{N}, \Theta = (\Theta_1, \Theta_2, \cdots, \Theta_{\mathcal{H}\mathcal{I}+2\mathcal{H}+\mathcal{I}}), x = (x_1, x_2, \cdots, x_{\mathcal{I}}) \in \mathbb{R}^{\mathcal{I}}$ and let $ReLU$ be the activation function as in 3.5. We now look at the function $\mathcal{N}^{\Theta,\mathcal{I}}_{\mathcal{M}_{ReLU,\mathcal{H}}, id_{\mathbb{R}}}(x)$. The activation function $ReLU$ acts as a function that mimics the behaviour of neurons in the human brain. It will fire, if a certain threshold is reached, i.e. the output of the function is x. Otherwise, it will remain inactive, i.e. the output is equal to zero. The composition with an affine function provides a weighting of the output of the activation function, so that we obtain the following equation:

$$\begin{aligned}
\mathcal{N}^{\Theta,\mathcal{I}}_{\mathcal{M}_{ReLU,\mathcal{H}}, id_{\mathbb{R}}}(x) &= ((id_{\mathbb{R}}) \circ \mathcal{A}^{\Theta,\mathcal{H}\mathcal{I}+\mathcal{H}}_{\mathcal{I},\mathcal{H}} \circ \mathcal{M}_{ReLU,\mathcal{H}} \circ \mathcal{A}^{\Theta,0}_{\mathcal{H},\mathcal{I}})(x) \\
&= \mathcal{A}^{\Theta,\mathcal{H}\mathcal{I}+\mathcal{H}}_{\mathcal{I},\mathcal{H}}(\mathcal{M}_{ReLU,\mathcal{H}}(\mathcal{A}^{\Theta,0}_{\mathcal{H},\mathcal{I}}(x))) \\
&= \Big[\sum_{k=1}^{\mathcal{H}} \Theta_{\mathcal{H}\mathcal{I}+\mathcal{H}+k}\ ReLU\Big(\Big[\sum_{i=1}^{\mathcal{I}} x_i \Theta_{(k-1)\mathcal{I}+i}\Big] + \Theta_{H\mathcal{I}+k}\Big)\Big] \\
&\quad +\Theta_{\mathcal{H}\mathcal{I}+2\mathcal{H}+1} \\
&= \Big[\sum_{k=1}^{\mathcal{H}} \Theta_{\mathcal{H}\mathcal{I}+\mathcal{H}+k}\ max\Big\{0, \Big[\sum_{i=1}^{\mathcal{I}} x_i \underbrace{\Theta_{(k-1)\mathcal{I}+i}}_{\text{weights}}\Big] + \underbrace{\Theta_{H\mathcal{I}+k}}_{\text{biases}}\Big\}\Big] \\
&\quad +\Theta_{\mathcal{H}\mathcal{I}+2\mathcal{H}+1}.
\end{aligned}$$

The input is altered by scaling (weighting) and shifting (biasing) the argument. As a consequence, a single real number is generated by each neuron in each layer, which is then passed on to the neurons in the next layer. A simplified representation of the network is shown in figure 8.

The example presented here has only one hidden layer. Deep learning is a loosely-defined term, it refers to a network with multiple hidden layers.

3.3 Loss function

Let's have a look at a general setup of a neural network to touch on more conceptional ideas. Let $M \in \mathbb{N}$ be the number of available input-output

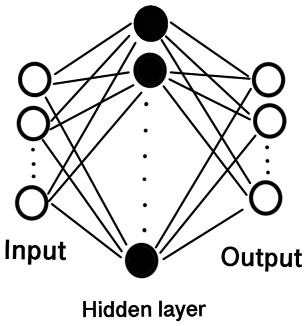


Figure 8 Illustration of a neural network with one input layer, one hidden layer and one output layer.

data pairs, with input dimension $d \in \mathbb{N}$. The basic idea behind machine learning is that a yet unknown function $\mathcal{E} \in C(\mathbb{R}^d, \mathbb{R})$ exists that matches the ith input data $x_i \in \mathbb{R}$ to the corresponding ith output data $y_i \in \mathbb{R}$ for $i \in \{1, 2, \cdots, M\}$, i.e.

$$y_i = \mathcal{E}(x_i), \text{ for all } i \in \{1, 2, \cdots, M\}.$$

With the knowledge of the M input-output pairs $(x_1, y_1) = (x_1, \mathcal{E}(x_1))$, $(x_2, y_2) = (x_2, \mathcal{E}(x_2)), \cdots, (x_M, y_M) = (x_M, \mathcal{E}(x_M))$, and without explicit knowledge of the function $\mathcal{E}$, we want to compute the output $\mathcal{E}(x_{M+1}) \in \mathbb{R}$ of the $(M+1)$th input data $x_{M+1} \in \mathbb{R}^d$. To do this, we need to solve the optimization problem of finding the global minimum of the function $\Phi : C(\mathbb{R}^d, \mathbb{R}) \to [0, \infty)$ which satisfies for all $\phi \in C(\mathbb{R}^d, \mathbb{R})$ that

$$\Phi(\phi) = \sum_{m=1}^{M} |\phi(x_m) - y_m|^2. \tag{2}$$

Note that $\Phi(\mathcal{E}) = 0$, so that the function $\mathcal{E}$ is the global minimizer of Φ. The function Φ is called *loss function* and indicates the quality of the prediction. We want to take a closer look at an example from book [Mur12, p. 56-57]. This function is also used later in the algorthmus.

3.9 Example (binary cross entropy) We can define the *entropy* $\mathbb{H}(p)$ of a discrete random variable X with distribution function p and sample space

$\Omega = \{1, ..., M\}$ as

$$\mathbb{H}(p) = -\sum_{m=1}^{M} p(X = M)\log(p(X = m)).$$

The *cross entropy* $\mathbb{H}(p, q)$ between two distribution functions p and q is then defined as

$$\mathbb{H}(p, q) = -\sum_{m=1}^{M} p(y_m)\log(q(y_m)),$$

where y_m is the output data. Then the *Kullback-Leibler divergence* provides a measure of the dissimilarity of two distribution functions via the equation

$$\mathbb{KL}(p||q) \quad = \quad -\mathbb{H}(p) + \mathbb{H}(p, q).$$

With the notation above now assume that we have a binary classification problem, i.e. $y_1, \cdots, y_M \in \{0, 1\}$. Then we can first state that $p(y_m) = \frac{1}{M}$ for all $m \in \Omega$ and the *binary cross entropy* function can be defined as follows

$$\mathbb{H}(p, q) \quad = \quad \frac{1}{M}\sum_{m=1}^{M} y_m\log(q(y_m)) + (1 - y_m)\log(1 - q(y_m)).$$

3.4 Gradient descent

We now want to repeat the already known concept of gradient descent in order to be able to look at a specific optimizer.
To do this, we recall the above-mentioned optimization problem from chapter 3.3 and thus look for the minima of the equation (2). Because of $\Phi \in C(\mathbb{R}^d, \mathbb{R})$ it is not amenable for discrete numerical computations, so we assume a spatially discretized version of the problem. More precisely, consider Φ restricted to a set of realization functions of neural networks.
Let $d, h \in \mathbb{N}$ and $l_1, l_2, \cdots, l_h$ such that

$$d = l_1(d + 1) + [\sum_{k=2}^{h} l_k(l_{k-1} + 1)] + l_h + 1.$$

We define $\mathcal{R}$ a subset of the Banach space $C(\mathbb{R}^d, \mathbb{R})$ as the set of realization functions of the neural network, i.e.

$$\mathcal{R} = \{(x \mapsto \mathcal{N}^{\theta,d}_{\mathcal{M}_{l_1},\mathcal{M}_{l_2},\cdots,\mathcal{M}_{l_h},id_{\mathbb{R}}}(x) \in \mathbb{R})|x, \theta \in \mathbb{R}^d\}$$

Here think of $\mathcal{N}^{\theta,d}_{\mathcal{M}_{l_1},\mathcal{M}_{l_2},\cdots,\mathcal{M}_{l_h},id_{\mathbb{R}}}(x)$ as described in the map (1). Now equation (2) can be reformulated and we want to approximately compute the minimal of the function

$$\begin{aligned} f:\mathbb{R}^d &\rightarrow [0,\infty) \\ \theta &\mapsto \sum_{m=1}^{M} \Big|(\mathcal{N}^{\theta,d}_{\mathcal{M}_{l_1},\mathcal{M}_{l_2},\cdots,\mathcal{M}_{l_h},id_{\mathbb{R}}}(x_m) - y_m)\Big|^2. \end{aligned}$$

Using the now generated approximation of the Banach space $C(\mathbb{R}^d,\mathbb{R})$, we can start solving the optimization problem.

3.10 Definition (Gradient descent) [JKea21, p.89 ff.] Let $d \in \mathbb{N}$ and $\xi \in \mathbb{R}^d$. Let $(\gamma_n)_{n\in\mathbb{N}} \subseteq [0,\infty)$. Let $(\gamma_n)_{n\in\mathbb{N}}$ be a sequence in $\mathbb{R}_{\geq 0}$, $f:\mathbb{R}^d \rightarrow \mathbb{R}$ a function and $\Theta = (\Theta_t)_{t\in[0,\infty)} : [0,\infty) \rightarrow \mathbb{R}^d$ be a continuously differentiable function which satisfies

$$\Theta_t = (\nabla f)(\Theta_t)$$

for all $t \in [0,\infty)$. Then $\theta = (\theta_n)_{n\in\mathbb{N}_0} : \mathbb{N}_0 \rightarrow \mathbb{R}^d$ is called the *gradient descent process* for the function f with generalized gradient Θ and initial value ξ if and only if θ satisfies for all $n \in \mathbb{N}$ that

$$\theta_0 = \xi \quad \textit{and} \quad \theta_n = \theta_{n-1} - \gamma_n(\nabla f)(\theta_{n-1}).$$

In a general setup we call $(\gamma_n)_{n\in\mathbb{N}}$ the *step size*, but in the context of machine learning it is also called the *learning rate*.

Now, as already announced, we will study a specific example of a gradient descent optimizer. We will look at the adaptive moment estimation optimizer, which is also called the Adam optimizer in the literature. As also quoted in the lecture [RN21, Lecture-notes 20: Deep Learning: Optimization Techniques], the Adam optimizer appears to be one of the best choices for gradient descent optimization in machine learning algorithms.

We want to look at an example of such an optimizer and use source [JKea21, p.130].

3.11 Definition (The Adam gradient descent optimization method) Consider again the setup as in Definition 3.10 and in addition let $\epsilon \in \mathbb{N}$, $(\alpha_n)_{n\in\mathbb{N}}$, $(\beta_n)_{n\in\mathbb{N}} \subseteq [0,1)$ be given. Then we call $\theta = (\theta^{(1)},\theta^{(2)},\cdots,\theta^{(d)})$:

$\mathbb{N}_0 \to \mathbb{R}^d$ the *Adam gradient descent process* for the function f with generalized gradient Θ, learning rates $(\gamma_n)_{n\in\mathbb{N}}$, momentum decay factors $(\alpha_n)_{n\in\mathbb{N}}$, second moment decay factors $(\beta_n)_{n\in\mathbb{N}}$ and initial value ξ. if and only if there exists functions $m = (m^{(1)}, m^{(2)}, \cdots, m^{(d)})$, $\mathbb{M} = (\mathbb{M}^{(1)}, \mathbb{M}^{(2)}, \cdots, \mathbb{M}^{(d)}) : \mathbb{N}_0 \to \mathbb{R}^d$ such that the function θ satisfies for all $n \in \mathbb{N}, i \in \{1, 2, \cdots, d\}$

$$\begin{aligned} \theta_0 = \xi, \qquad & m_0 = 0, \qquad && \mathbb{M}_0 = 0, \\ m_n = \quad & \alpha_n m_{n-1} + (1-\alpha_n) && \Theta(\theta_{n-1}), \\ \mathbb{M}_n^{(i)} = \quad & \beta_n \mathbb{M}_{n-1}^{(i)} + (1-\beta_n) && |\Theta_i(\theta_{n-1})|^2, \end{aligned}$$

$$\text{and } \theta_n^{(i)} = \theta_{n-1}^{(i)} - y_n \Big[\epsilon + \Big[\frac{\mathbb{M}_n^{(i)}}{1-\prod_{l=1}^n \beta_l}\Big]^{\frac{1}{2}}\Big]^{-1} \Big[\frac{m_n^{(i)}}{1-\prod_{l=1}^n \alpha_l}\Big].$$

4 Convolutional neural network

In this chapter we will look at the architecture of a convolutional neuronal network (CNN), based on [AH17, Chapter 3]. A CNN can consist of several layers that filter the input by applying different functions. Since the filters are only applied repetitively to subsets of the input, the weight matrix is extremely sparse compared to a standard neural network. A CNN is inspired by the human visual cortex, hence the entries in the model correspond matrix are called neurons. In the following chapter, we will always assume a grayscale image with width $W \in \mathbb{N}$ and height $H \in \mathbb{N}$.

4.1 Convolutional layer

CNNs are mostly applied to images. So consider as input a grayscale image, then it has $W \cdot H = K \in \mathbb{N}$ pixel. We assume that there are N neurons in the first hidden layer. The neurons can be rearrange into l blocks of $M_1 \times M_2$ matrices, such that $M_1 \cdot M_2 \cdot l = N$ and $M_1, M_2, l \in \mathbb{N}$. Considering that each pixel in the image is correlate with its neighbors, than with pixels that are far a way, we can connect each neuron of one block to an $R_1 \times R_2$ pixel region in the image. This matrix is called the *receptive field* of the neuron and implies that neurons in one block cover all the pixel of the image. Now we choose a weight $w^i_{a,b}$ with $i \in \{1, 2, \cdots, l\}$, $a \in \{1, 2, \cdots R_1\}, b \in \{1, 2, \cdots R_2\}$ for each block, so that all neurons in one block share the same weight.

If we define a neuron by its position in the $M_1 \times M_2$ matrix, i.e. denoted by (p, q) and a pixel by its position in the $R_1 \times R_2$ matrix, we get a function $f^l(p, q)$, that compute the output of the l-th matrix. Specifically, we define a function

$$f^l(p, q) = (\mathcal{M})\Big(\sum_{i=0}^{R_1-1} \sum_{j=0}^{R_2-1} \operatorname{im}(p+i, q+j) w^l_{i,j} \Big).$$

Here $\mathcal{M}$ is a activation function and im is a function that refers to the pixel $(p + i, q + j)$ of the image. The function f^l describes a convolution which is denoted by $*$ in the literature. A convolution layer consists of l-many $M_1 \times M_2$ filters which are convolved with the image. The output of the layer results in l images on which the activation function $\mathcal{M}$ is applied and is called *feature map*. If the image is multi-coloured, a third dimension is added so that initially a $W \times H \times C$ matrix is considered. Here, the variable C represents the number of channels.

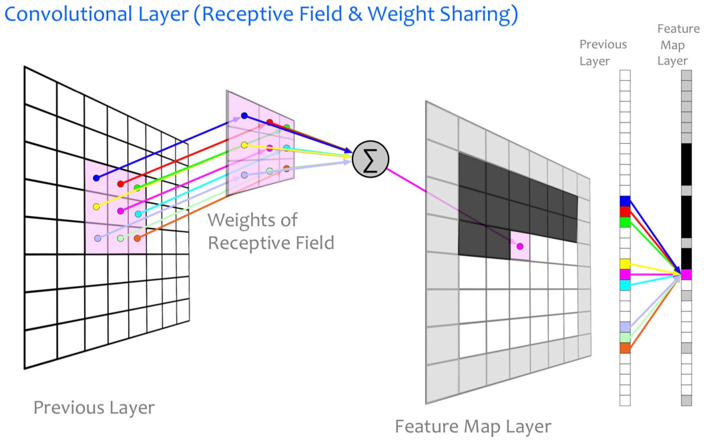

Convolutional Layer (Receptive Field & Weight Sharing)
Previous Layer
Feature Map Layer
∑
Weights of Receptive Field
Previous Layer
Feature Map Layer

Figure 9 [RN21, Chapter 19] Visualization of weight sharing in the feature map.

4.2 Filter

4.1 Definition (Filter) Let $\mathbf{X} \in \mathbb{R}^{W \times H}$ be an image. Then the convolution of the image with a filter $f \in \mathbb{R}^{P \times Q}$ is given by

$$(\mathbf{X} * f)(m,n) = \sum_{i=0}^{P-1} \sum_{j=0}^{Q-1} \mathbf{X}(sm+i, sn+j) f(i,j), \tag{3}$$

with $sm \in \{0, 1, \cdots, H-1\}$, $sn \in \{0, 1, \cdots, W-1\}$ and $s \in \mathbb{N}$. We refer to $f \in \mathbb{R}^{P \times Q}$ as *kernel* and s denote the *stride* of the layer. The variable $s, P, Q \in \mathbb{N}$ has to be choosen, such that $\frac{W-P}{s} + 1 \times \frac{H-Q}{s} + 1 \in \mathbb{N}$. The resulting ouput matrix lies in $\mathbb{R}^{\left(\frac{W-P}{s}+1\right) \times \left(\frac{H-Q}{s}+1\right)}$.

4.2 Notation (Pooling) Pooling, also called downsampling, is the process of dividing the image $\mathbf{X} \in \mathbb{R}^{W \times H}$ in a $d \times d$, $d \in \mathbb{N}$ region every $s \in \mathbb{N}$ pixels. The application of a function g on the region leads to just one entry in the feature map of the respective region. The parameter s of the pooling layer is denoted as stride and $d \times d$ refers to the kernel. If $s \leq d$ holds, the pooling regions overlap.

Overall, the use of a pooling layer leads to a reduction of the spatial dimension of the feature maps without any significant loss of information. By applying a function g to the regions in the image, information from the

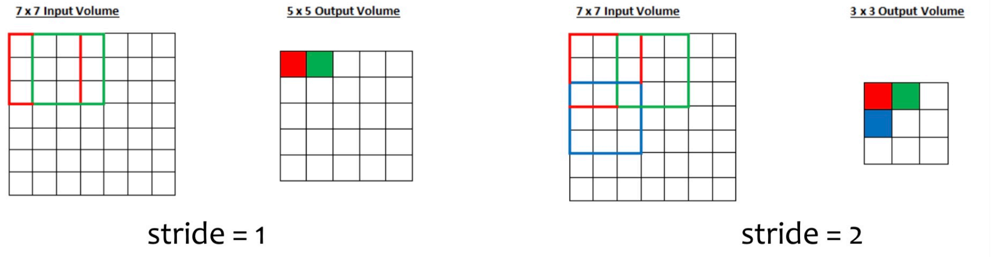


Figure 10 [JRea21][Chapter 9] Example of different strides s in a layer.

neighbouring pixels is included. The choice of the function g specifies the pooling layer.

4.3 Example

1. If $g = \max_{i \in \{0,1,\cdots,d-1\}} x_i$, then we speak of *max pooling*. This type of pooling is often used in the literature and leads to good results [AH17][p.97].
2. Choosing $g = \frac{1}{d} \sum_{i=0}^{d-1} x_i$ leads to *average pooling*.

4.4 Notation (Zero-padding) Padding describes the extension of the image matrix $\mathbf{X}_{(p,q)} \in \mathbb{R}^{W \times H}$ by k zero rows and columns. More precisely let (p, q) denote the position of a pixel, then

$$\text{padding}(\mathbf{X}_{(p,q)}) = \begin{cases} 0 & \text{, for } p \in \{0,1,\cdots,k\} \cup \{W+1, W+2, \cdots, W+k\} \text{ or} \\ & q \in \{0,1,\cdots,k\} \cup \{H+1, H+2, \cdots, H+k\} \\ \text{im}(p,q) & \text{, otherwise.} \end{cases} .$$

Padding is used to keep the size of the image constant after applying filters [AH17][p.140].

Now that all the important elements for creating a CNN are defined, they need to be combined. The source [AH17][Chapter 3.3, Chapter 3.5.1] was used as a reference for the next chapter.

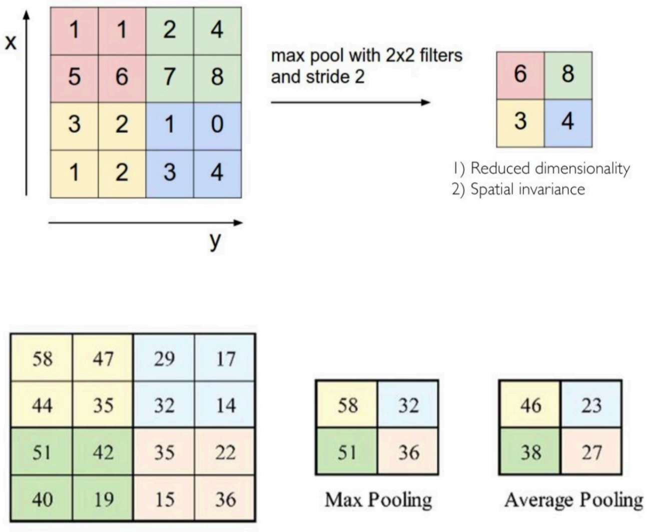


Figure 11 [JRea21][Chapter 9] Visualization of max and average pooling in one layer.

4.3 Network architecture

There are many options for building a CNN and there exists no golden rule for designing one. Usually, the design of the network starts with a first architecture and afterwards adjusting the parameters.Through literature research, one can discover which architectures already have been successful for the same type of input data. Furthermore, there are already many available network architectures that are widely used, for example LeNet or VGG.
As a rule of thumb, a CNN consists of several layers, with the activation function placed before the pooling layer. Furthermore, it ends with a fully connected layer in order to perform a classification. When choosing the parameters, it should be ensured that the feature maps passed have suitable dimensions for the next layer.

Performance parameters are used to classify such networks. The following introduction to the topic is based on [Tra19, chapter 3.2.].

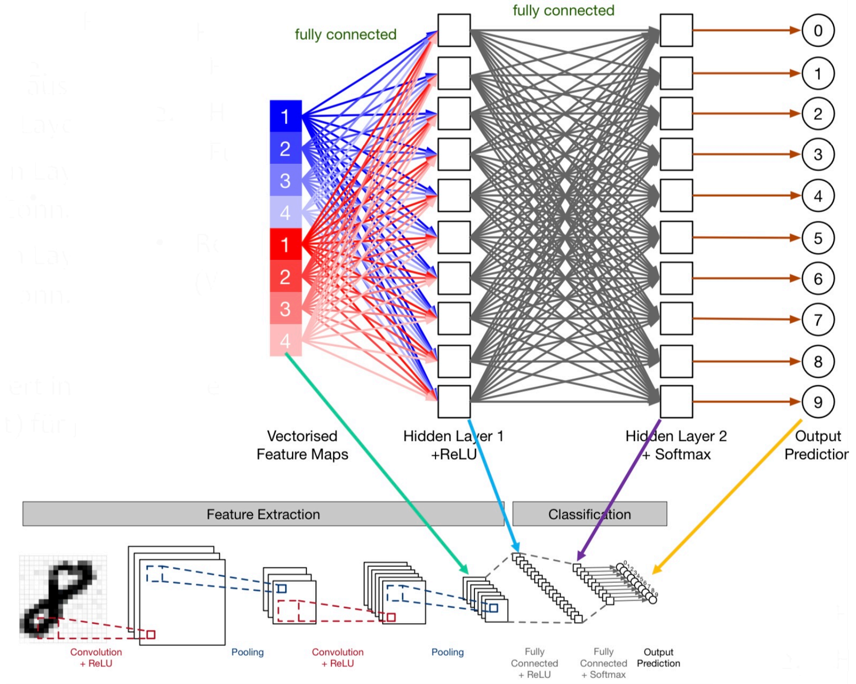


Figure 12 [RN21][Chapter 19] LeNet: Example of a CNN architecture.

4.4 Performance parameter

For binary classification problems, the quality of the network is easily measured by the output of the functions $accuracy$, $specificity$ and $sensitivity$. To calculate these functions, the true positive (TP), true negative (TN), false positive (FP) and false negative (FN) values must be computed after the network has been created. With the scikit-learn library [Sl] the above parameter can be visualized by a confusion matrix.

```
import sklearn
import seaborn
#Predicting the output values
y_pred = model.predict(x_test)
y_test_pred = np.argmax(y_pred, axis=1)
#Calculating the entries for the confusion matrix
Matrix_prediction = sklearn.metrics.confusion_matrix(y_test, y_test_pred)
#Visualize matrix
cf_matrix= seaborn.heatmap(Matrix_prediction, annot=True, cmap=’Blues’)
```

```
cf_matrix.set_title('Confusion Matrix with labels\n\n')
cf_matrix.set_xlabel('\nPredicted Values')
cf_matrix.set_ylabel('Actual Values')
cf_matrix.xaxis.set_ticklabels(['False','True'])
cf_matrix.yaxis.set_ticklabels(['False','True'])

plt.show()
```

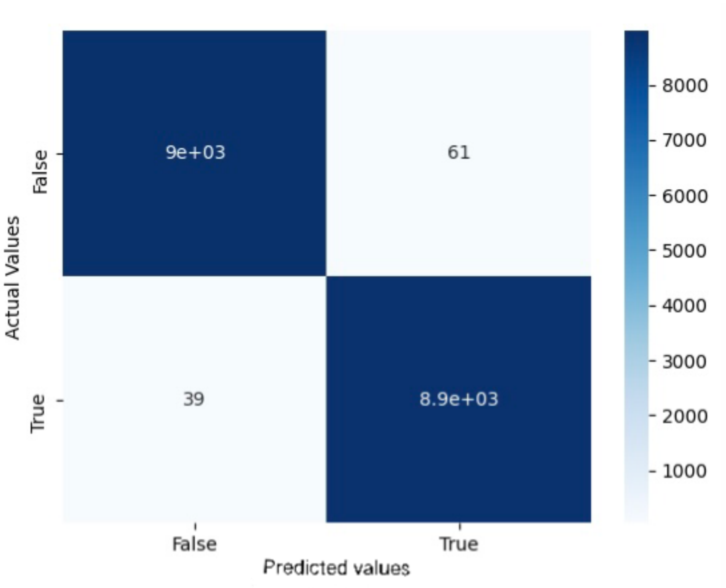


Figure 13 Confusion matrix implementd with the above code

4.5 Definition (Accuracy) The *accuracy* is a function that outputs the percentage of correct classification:

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

4.6 Definition (Sensitivity) The sensitivity function measures the percentage of correctly positive predictions compared to all positive predictions:

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

4.7 Definition (Specificity) The $specificity$ function measures as a counterpart of the sensitivity the percentage of correctly negative predictions compared to all negative predictions:

$$specificity = \frac{TN}{TN + FP}$$

5 Collected Dataset

In this chapter, the used datasets are presented. The first section explains how IEDs, or short 'spikes', were simulated using the software Curry [Cur] and MATLAB [MAT]. The following section deals with data collected from a patient at the Institute of Biomagnetism and Biosignalanalysis in Münster. Details of the studies can be found in the used source [ARWea19] and [Kai22]. These datasets were pre-processed with the MATLAB based open source software Brainstorm [Bra].

5.1 Pre-Processing

The pre-processing of the datasets was based on the paper by Lorenco et al. [?] who were able to successfully detect IEDs with EEG data. In summary, the following steps were performed:

- Application of a bandpass filter between 0.5 and 30 Hz
- Lowering the sample frequency to 125 Hz
- Applying a longitudinal bipolar montage
- Splitting the data into two second non-overlapping samples

The steps applied correspond to the visualisation techniques used by physicians in daily clinical practice. In [ED11, p. 187], Ebner recommends using a filter between 1.6 and 30 Hz for the pattern detection of spikes. Furthermore, the use of a longitudinal bipolar montage increases the visibility of a phase inversion, which also occurs as a classification feature in epileptic activity [ED11, p. 22]; 2.3.1. Lowering the sample frequency only acts as a minimization of the data memory.
If not stated otherwise, this process was carried out in Brainstorm and applied to all recordings.

5.2 Simulated Data

In order to test the algorithm later, simulated IED data were first created. This provides a theoretical basis in which various noise-signal-ratios can be tested. The basis of the noise file was still real data which was clinically inconspicuous.

5.2.1 Recording

An EEG and an MEG measurement of a subject from a studie at the Institute for Biomagnetism and Biosignalanalysis were taken simultaneously in 2022. Afterwards, the MEG was examined by a physician for abnormalities and assessed as clinically inconspicuous.

5.2.2 Electrodes

Since the measurement was carried out with only 54 electrodes instead of 84, the default available longitudinal montage was not applicable in Brainstorm and thus the only standard pre-processing skipped. In order to simulate natural noise, the electrodes closest to the actual montage were selected. FIG illustrates the choice of electrodes. The selection of the electrodes and the associated subtraction was carried out in a separate Matlab file.

5.2.3 Spike simulation

The following steps for the spike simulations were produced with the software Curry. Curry contains a tool for simulating data, for which an empty file of variable duration is first created. To match the noise data two seconds were used. A sample file with 28 electrode positions is included in the standard Curry version and was used for this simulation. A preferred dipole position was then selected in the default brain. The selected position is frontal, in the area of electrode F3, as is often the case for epilepsy patients [ED11, Chapter 12.2.2]. With the dipole position now selected, a spike can be simulated. A spike in the software is defined by its strength, shape and duration. I used the average values of each category from chapter 2.3.1 as a guide, so that in summary the spike wave complex has a strength of 80 μAmm at the spike peak and 20 μAMm at the wave peak and lasts 125 ms. The starting time of the spike wave complex was shifted by 402 ms, starting at 0, so that 6 files were created. For a better overview, the exact parameters can be read from the file names. The file name is divided into the following parts $Dipole\_Startingtime\_Duration\_x - y - z - Coordinates\_Spike - Strength$.

5.2.4 Simulated Spike file

The merging of the simulated spike-wave complex and the noise file were completed in python. At the beginning, the selected spike file should be classified by specifying the start time of the spike. Then both files are loaded and a signal-to-noise ratio (SNR) can be selected. Before combining the two matrices, five channels of the noise file are permuted, so that the number of samples increases by a factor of 120. The resulting file is created in a for loop by adding the spike to each of the 250 noise files. Both matrices can be multiplied in this step by any factor chosen beforehand. The result of the addition is stored in the vector *epoch_all*. In parallel, a classification vector *event_all* is created, which receives the entry one for each of these resulting matrices. The control data consisted of noise data without an additional spike. These were also stored in the *epoch_all* vector and received the entry zero in the *event_all* vector. In summary, the permutation of the electrode channels created 30000 samples containing a spike and 30000 without.

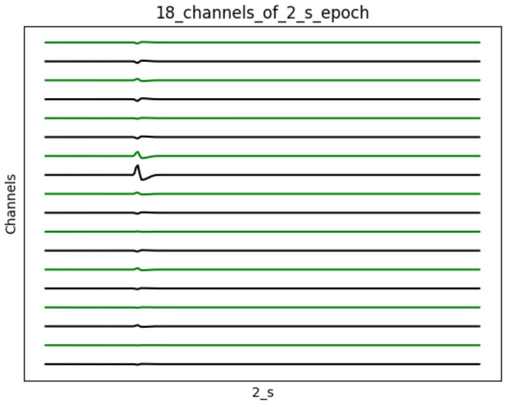


Figure 14 Simulated Spike data

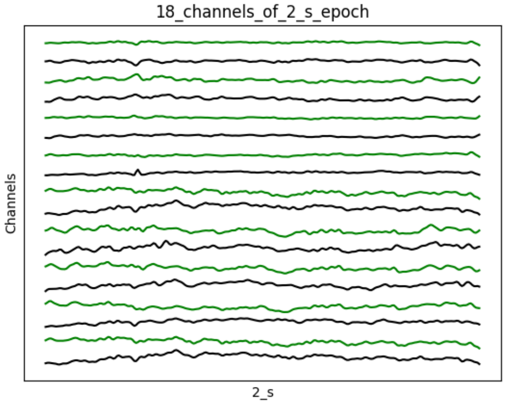


Figure 15 Simulated Spike with a SNR of 1

5.3 Patient history

The patient is a 20-year-old female, who was diagnosed with epilepsy at the age of 14. Her seizure semiology is described as distributed thinking and the inability to speak or follow conversations. Motor symptoms or impairment of awareness are not present. Despite several antiepileptic drug(AED) treatments, the patient did not become seizure free, so presurgical evaluation was done, including the implantation of four invasive electrodes in the frontal lobe. During the examinations a FCD(Focal cortical dysplasia) was found near Broca's area. Resective epilepsy surgery was not considered and the electrodes were removed since only typical spread patterns instead of seizure onset patterns were recorded. After another unsuccessful attempt to alleviate the seizures with medication, the patient was re-examined in 2019. In order to perform a better source analysis, the medication was temporarily discontinued in 2019 in consultation with a physician. The EMEG recording with 1050 IEDs discussed in this paper was then recorded.

5.4 Recording 2019

Simultaneous EEG and MEG measurement was performed for a total of 47 minutes. The measurement is divided into six parts. The first measurement is only seven minutes long and contains somatosensory evoked potentials and fields (SEP/SEF). Hence it was omitted. The remaining 40 minutes are

equally divided into five sessions of eight minutes each. All measurements had a sample frequency of 2400 Hz. The EEG system consisted of 80 electrodes and the MEG system (VMS MedTech Ltd) included of 275 third order axial gradiometers. The patient remained in a supine position. She was then instructed to close her eyes and relax during the recording.
After the measurements, an epileptologist marked 1050 IEDs with the highest negativity in the EEG channel F3.

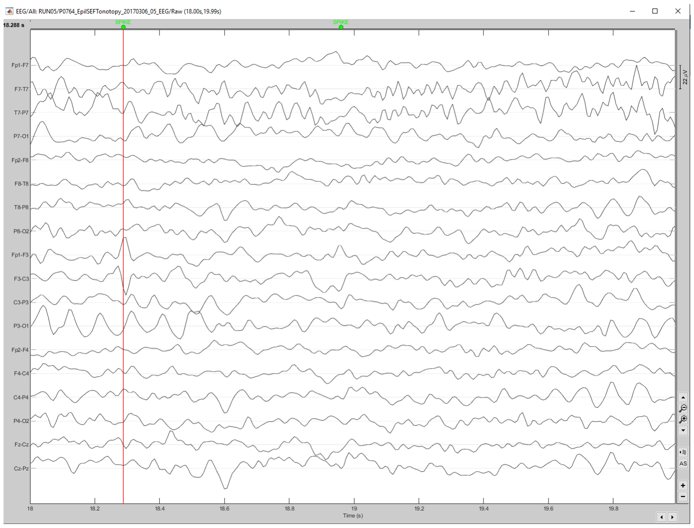


Figure 16 Preprocessed file with spike annotation

5.4.1 Data multiplication

In order to be able to train the machine learning algorithm, a large number of data samples is needed. Since only 997 samples were generated after pre-processing the data, the number was artificially multiplied in this step, following the example of C. Lourenç et al. [LTCvP21]. The first step was to apply two additional montages: Since brainstorm contains three different variants of a longitudinal montage, the other two available were applied: "Longitudinal 1" and "Longitudinal 2". This led to a threefold multiplication of the number of samples. In addition, the time window of the samples was shifted forward by one second on each of the three montages. This lead to an overall temporal and spatial change of the IED location and to a dataset that is about six times larger than before.

Montage	# samples	# IED samples	% IED samples/ samples
Montage_all	6961	4044	58,1%
Longitudinal_1	1991	1183	59,42%
Longitudinal_2	2206	1260	57,12%
Longitudinal_3	2764	1601	57,92%

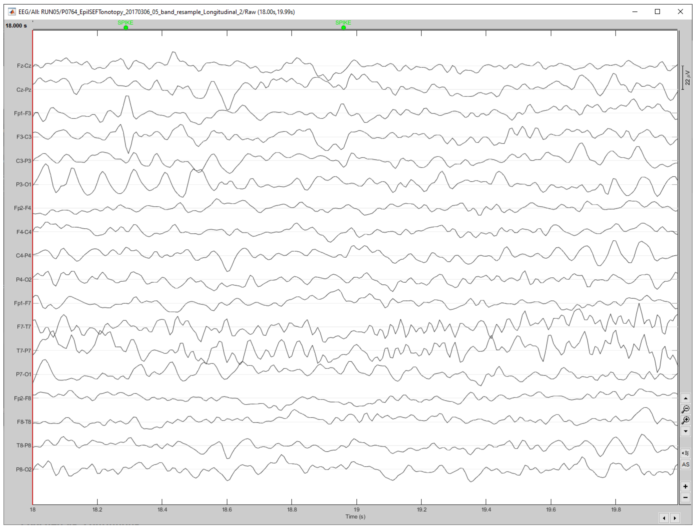


Figure 17 Same sample as figure 16 with a different montage

5.5 Recording 2021

At the time of measurement in 2021, the patient was 23 years old and participated in a double-blind sham-controlled pilot/feasibility clinical trial to investigate the effect of personalized and dCMI-optimized tDCS on IED frequency. The study took place at the University Hospital Münster. An activ sham method was used to control the effect, which included stimulation with adjacent cathodes and anodes. For the study, the patient was stimulated twice a day for twenty minutes with a 20 minute break between blocks. A one-hour EEG was recorded before and after the stimulation. This was carried out on five days within one week. For a better baseline measurement, a two hour long EEG measurement was taken on the first day before stimulation instead of one hour. After five weeks, the active sham method was applied. It is structured in the same way as the stimulation week, with the stimulation blocks being replaced by the activ sham blocks. The tDCS

IED Marker	# IED samples	% IED samples/ samples
SPIKE_1	1647	4,32%
SPIKE_2	1258	3,3%
SPIKE_3	5472	14,35%
SPIKE_all	568	1,49%
SPIKE_1 or SPIKE_2	2825	7,41%
SPIKE_3 or SPIKE_2	6466	16,96%
SPIKE_1 or SPIKE_3	5607	14,7%
SPIKE_1 or SPIKE_2 or SPIKE_3	6597	17,3%

used was a model from Starstim developde by neuroelectrio and the headcap choosen is the neo19 neoprene with 39 predefined positions based on a subset of 10-20 EEG system. The 19 electrodes of the EEG were etched according to the template of the cape. The EEG had a sampling frequency of 200 Hz.

5.5.1 IED Marking

Three epileptologists independently performed the marking of IEDs. The study was completely blinded, i.e. the epileptologists were not able to see the time of recording the EEG data or similar.

5.6 Pre-processing of the IED Marking

For pre-processing this recordings the same steps were applied as for the 2019 recordings. Since fewer electrodes were available, a two-second epoch now contains only 12 channels instead of 18. The markers of the three epileptogens were saved as "SPIKE_1", "SPIKE_2" and "SPIKE_3" respectively as labels of the file. An additional marker was defined under the name "SPIKE_all" which is defined as the intersection of the annotations of all three epileptologists with a tolerance of 8 ms. Splitting the data set in two-second epochs, a total of 38 132 samples were generated.

5.7 Marking Agreement

When comparing the label of the two seconds epochs, the following distribution of the labeled data emerges.

# Spike_1 ∩ Spike_2	# Spike_1 ∩ Spike_3	# Spike_2 ∩ Spike_3
80	1512	264

Table 1

Only 5% of the other IED markings match the results from epileptologists with annotation 'SPIKE_2'. Evaluators with the marking 'SPIKE_1' and 'SPIKE_3', on the other hand, share about 91% of the same labeld IED epochs in the case of 'SPIKE_1'. This variance may result from inaccurate marking and other used classification methods.

6 IED Detection

6.1 Related work

The detection of IEDs in the EEG using neural networks has been studied in previous works. More recently, in 2021, C. Lourenço et al. [LTCvP21] used a VGG network (abbreviation for Visual Geometry Group) on a dataset of 166 patients. Time shifting and montage changes have further increased the number of epochs with IEDs. The network is visualised in Figure 18. An Adam optimizer with learning rate $2 * 10^{-5}$ and $ReLU$ as activation function was used. In addition, the dataset was divided so that 80% belonged to the training set and 20% to the test set. The results were a specificity of up to 99% and a sensitivity of up to 96%.

In a work two years earlier a specificity of 98% on the training data and 93% on the test data was achieved with the same network.[LTCvPea19] Here, the author used 112747 two-second epochs as input files. The training and testing of such a deep network required more than 75 hours.

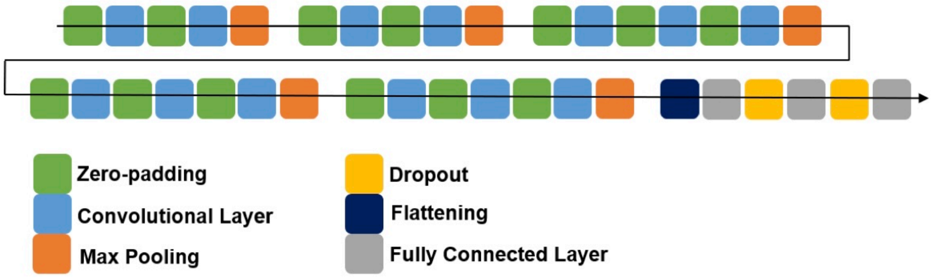


Figure 18 [LTCvP21] Visualization of a VGG convolutional neural network, that was used for detecting IED in EEG data

6.2 Algorithm

The following CNN was built in Python 3.8 using the tensorflow [Ten] and scikit-learn [Sl] library.
The architecture of the network follows the design already successfully tested by Lourenço et al.[?]. Since a much larger data set was chosen in the referred studies in order to train the CNN, which was not available in this study, viewer layers were used. By reducing the number of layers, only 1052 parameters were trained, and the time required to train and test the data was limited to a maximum of 30 minutes. Besides the already mentioned parameters in previous chapters, one could also determined a *dropout_rate*. If the network can only predict the given data effectively, but cannot be adapted to other data, we speak of *overfitting*. By using a droput layer, this problem can be reduced. The principle here is that the dropout sets a certain rate of arbitrary neurons to zero for the next calculation step [HH18, Chapter 7.2]. The following code sample shows the preselection of the parameters:

```
filter = 2
kernel_size = 2
pool_size = 2
opt = Adam(lr=0.00002)
dropout_rate = 0.5

if Data_2021_load == 1:
    input_shape =(12,250)
else:
    input_shape=(18,250)
```

In another document, the CNN was then defined as a function.

```
#Function call of the CNN model
import CNN_Function as CNN
model = CNN.Modell(dropout_rate = dropout_rate, input_shape=
                input_shape,pool_size = pool_size, filter =filter,
                kernel_size =kernel_size)

from tensorflow import keras
from keras.layers import Dense, Flatten, Dropout, Conv1D, MaxPool1D
from keras import Sequential
from keras.optimizers import Adam

def Modell(filter, kernel_size,pool_size,dropout_rate, input_shape):
    model = Sequential()

    model.add(ZeroPadding1D(padding=1, input_shape = input_shape))
    model.add(Conv1D(filters = filter, kernel_size = kernel_size,
            activation = "relu"))
    model.add(MaxPool1D(pool_size= pool_size, strides= 1))
    model.add(Flatten())
    model.add(Dropout(rate = dropout_rate))
    model.add(Dense(units = 2, activation="relu"))

    opt = Adam(lr=0.00002)
    model.compile( loss = 'binary_crossentropy', optimizer = opt,
                metrics = ['accuracy'])

    return model
```

Notice that at the end the network receives a *flatten-layer* which transforms the input feature map into a one dimensional array ([tf.b]). Then the last layer, called *dense*, operates as a fully connected layer where *unit* represents the size of the output ([tf.a]). During the subsequent traning of the model, only as much tranings data is used per run as the specified *batch_size* indicates. This process is repeated according to the number of *epochs*.

```
model.fit(x_train, y_train, batch_size= batch_size, epochs=epochs,
            verbose=0, validation_data=(x_test,y_test),
            callbacks=[history])
```

7 Results

This chapter presents the results of the CNN introduced in section 6.2. We distinguish between the three datasets from chapter 5, with the last chapter explaining the use of multiple datasets. Generally, the number of epochs, the batch-size and the train-test-ratio were modified. The test size is presented as a decimal number to represent its proportion to the total sample data. Additional data-specific changes are described in the respective sub-chapters.

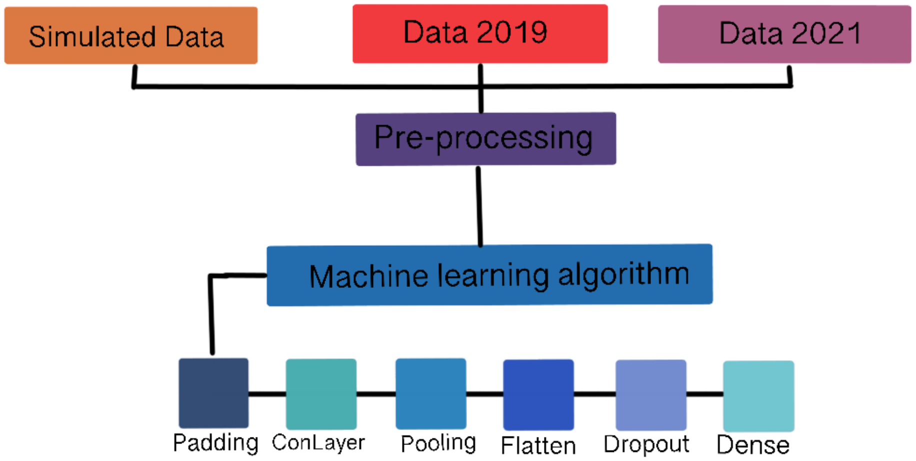


Figure 19 Study design and architecture of the CNN. Here, ConLayer refers to a convolutional layer.

7.1 Results Simulation

The data generated in chapter 5.2 is considered to be the input file of the algorithm to test it under controlled conditions. Selecting the spike start time fixed as zero, the following results were calculated.
In general, an improvement of the accuracy with a simultaneous increase of the signal-to-noise ratio can be noted. Otherwise, parameter changes do not lead to significant reductions. The only outlier arises from the performance results of the signal-to-noise ratio 100 with an accuracy that has decreased by approx. 50% in comparison between test size 0.4 and size 0.3.

Signal-Noise-ratio	test_size	epochs	batch_size	accuracy	sensitivity	specificity
100	0.4	250	32	92,99%	87,67%	100%
100	0.3	250	32	42,33%	0%	84,14%
100	0.2	250	32	100%	100%	100%
10	0.4	250	32	99,46%	99,08%	99,84%
10	0.3	250	32	99,44%	99,56%	99,33%
10	0.2	250	32	99,8%	100%	99,64%
1	0.4	250	32	49,87%	100%	0%
1	0.3	250	32	57,09%	71,53%	42,83%
1	0.2	250	32	51,01%	1,88%	98,67%

Signal-Noise-ratio	test_size	epochs	batch_size	accuracy	sensitivity	specificity
100	0.4	250	64	100%	100%	100%
100	0.4	250	32	92,99%	87,67%	100%
10	0.4	250	64	95,31%	97,68%	92,96%
10	0.4	250	32	99,46%	99,08%	99,84%
1	0.4	250	64	50,29%	0,19%	99,77%
1	0.4	250	32	49,87%	100%	0%

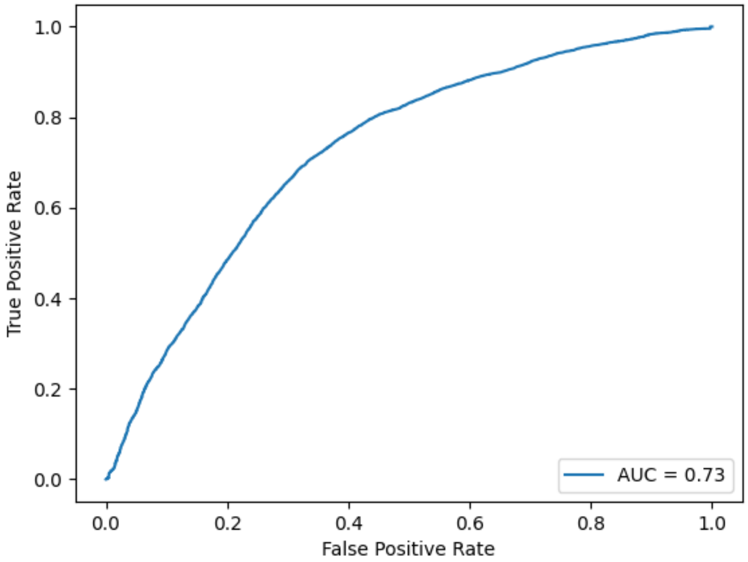


Figure 20 This curve is denoted as ROC curve and is used for evaluation of the performance of the algorithm. The area below the curve is abbreviated as AUC. If the AUC is equal to one, the algorithm operates perfectly. Accordingly, the closer it is to one, the higher is the performance of the CNN. Here, the ROC curve is calculated with the simulated data and a SNR of one.

7.2 Results Data 2019

The data set from 2019 receive a maximum accuracy of 57% on the tranings data and 59% on the test data. This can be achieved with a test size of 2088 data, which represents 30% of the total input data.

test_size	epochs	batch_size	train_accuracy	test_accuracy
0.4	250	32	58,9%	56,88%
0.3	250	32	58,76%	56,53%
0.2	250	32	41,51%	43,5%

Changing the other parameters does not result in a significant improvement of the accuracy.

test_size	epochs	batch_size	train_accuracy	test_accuracy
0.3	1000	32	58,76%	56,53%
0.3	500	32	58,76%	56,53%
0.3	250	32	58,61%	57,05%

7.3 Results Data 2021

As already mentioned in chapter 5.5, the data set of 2021 contains four markers. In the first part of this chapter, we look at the results of the CNN, which results from applying the algorithm using the respective annotations. The second part analyses the annotations of all physicians, which is, by definition, a union of all markers.

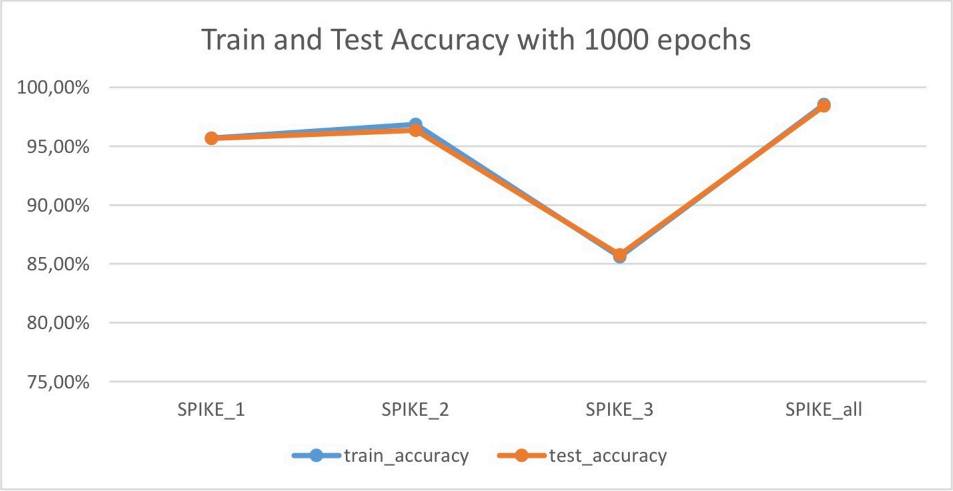
Keeping all parameters constant except for the test_size, the model achieves the best accuracy result on the test data for all classes except for 'SPIKE_1' with a test size of 0.4. Using the annotation 'SPIKE_1', the test size 0.3 performs slightly better (95.66%). The table visualises the results of all markings, with the number of epochs set to 250.

IED Marker	test_size	train_accuracy	test_accuracy
SPIKE_1	0.4	95,61%	95,63%
SPIKE_1	0.3	95,68%	95,66%
SPIKE_1	0.2	95,08%	94,98%
SPIKE_2	0.4	96,79%	96,37%
SPIKE_2	0.3	96,85%	96,36%
SPIKE_2	0.2	96,78%	96,37%
SPIKE_3	0.4	85,31%	85,63%
SPIKE_3	0.3	85,04%	85,17%
SPIKE_3	0.2	85,49%	85,75%
SPIKE_all	0.4	98,48%	98,41%
SPIKE_all	0.3	98,52%	98,37%
SPIKE_all	0.2	98,45%	98,26%

By changing only the number of epochs, the accuracy does not increase significantly for each marker. The table shows the results of the different epochs with a test size of 0.3.

IED Marker	epochs	train_accuracy	test_accuracy
SPIKE_1	1000	95,69%	95,67%
SPIKE_1	500	95,65%	95,66%
SPIKE_1	250	95,68%	95,66%
SPIKE_2	1000	96,85%	96,36%
SPIKE_2	500	96,83%	96,38%
SPIKE_2	250	96,85%	96,36%
SPIKE_3	1000	85,6%	85,77%
SPIKE_3	500	85,36%	85,29%
SPIKE_3	250	85,04%	85,17%
SPIKE_all	1000	98,55%	98,43%
SPIKE_all	500	98,42%	98,24%
SPIKE_all	250	98,52%	98,37%

In the group comparison, the marker 'SPIKE_all' performs best with a highest train accuracy of 98.55% and test accuracy of 98.43%.



The sensitivity is very low in all cases, while the specificity is high. If only the batch_size is varied, a maximum sensitivity of 13.76% is reached by all classes for 'SPIKE_1'. Using the union of the markers, we get a very liberal marking with 6597 IED samples. If the test size is varied in this case, a test accuracy of 82.16% is obtained when dividing the data for testing and training in half. It can also be observed that all other test sizes achieve similar results.

IED Marker	test_size	epochs	batch_size	accuracy	sensitivity	specificity
SPIKE_1	0.4	250	64	80,72%	13,76%	83,72%
SPIKE_1	0.4	250	32	94,42%	2,29%	98,55%
SPIKE_2	0.4	250	64	95,34%	1,29%	98,82%
SPIKE_2	0.4	250	32	96,22%	0,37%	99,76%
SPIKE_3	0.4	250	64	83,04%	4,4%	96,01%
SPIKE_3	0.4	250	32	84,71%	1,62%	98,42%
SPIKE_all	0.4	250	64	97,38%	1,27%	98,89%
SPIKE_all	0.4	250	32	97,95%	0,42%	99,49%

test_size	epochs	batch_size	train_accuracy	test_accuracy
0.5	500	64	82,31%	82,16%
0.4	500	64	80,88%	79,96%
0.3	500	64	82,39%	81,86%
0.2	500	64	82,39%	81,86%

With a constant test size of 0.4 and varying number of epochs, the best train accuracy of 82.69% is achieved with 300 epochs. The maximum test accuracy (82.88%) is reached with 250 epochs.

test_size	epochs	batch_size	train_accuracy	test_accuracy
0.4	10000	64	80,97%	79,83%
0.4	1000	64	82,67%	82,76%
0.4	500	64	80,88%	79,96%
0.4	300	64	82,69%	82,67%
0.4	250	64	81,94%	82,88%
0.4	100	64	75,29%	75,29%

When comparing the batch size of 64 and 32, the latter performs better in terms of accuracy $(82,55\%)$. Note that with a batch site of 64, a much higher sensitivity $(29,10\%)$ is achieved.

test_size	epochs	batch_size	accuracy	sensitivity	specificity
0.4	250	64	66,06%	29,1%	74%
0.4	250	32	82,55%	0,34%	99,69%

7.4 Results mixed Data

For this section, we will follow the same procedure as for the previous section. First, we use only the individual markings of each physician for the CNN and then their union. Furthermore, the data set from the year 2019 is always added. As the 2019 dataset consists of 6 more channels than the dataset from 2021, only channels 3 to 14 of the 2019 dataset are selected. For clarity, the tables only show which IED marker was used.

If the test size changes, there is only a slight difference in ther perfomance within the respective markings used for the input file. The best results are obtained when using the marker 'SPIKE_2' and a test size of 0.4 ($train\_accuracy = 90,63\%$).

IED Marking	test_size	epochs	batch_size	train_accuracy	test_accuracy
SPIKE_1	0.4	250	32	89,92%	89,68%
SPIKE_1	0.3	250	32	89,93%	89,67%
SPIKE_2	0.4	250	32	90,63%	90,61%
SPIKE_2	0.3	250	32	90,38%	90,35%
SPIKE_3	0.4	250	32	81,05%	80,61%
SPIKE_3	0.3	250	32	81,46%	80,94%

No significant changes were observed when different batch sizes were used, except for the marker 'SPIKE_2', which showed an increase in accuracy when using a batch size of 32. This is also the highest achieved accuracy (96.38%) compared to the other markers.

IED Marking	batch_size	accuracy	sensitivity	specificity
SPIKE_1	64	89,08%	71,71%	91,65%
SPIKE_1	32	89,68%	71,40%	92,38%
SPIKE_2	64	88,3%	77,56%	89,76%
SPIKE_2	32	96,38%	77,00%	99,20%
SPIKE_3	64	79,61%	43,78%	89,40%
SPIKE_3	32	80,61%	43,36%	90,79%

Table 2 Performance parameter measured with epochs = 250 and test_size =0.5

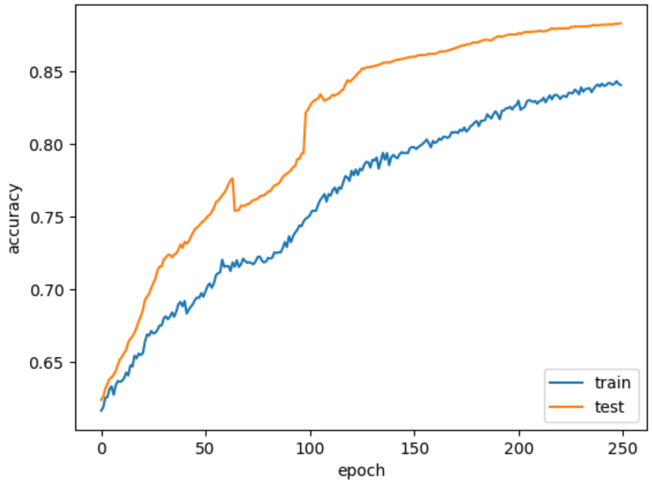


Figure 21 Accuracy trend as dependent on the number of epochs for the data set with the marker 'SPIKE_1' combined with the data set from 2019

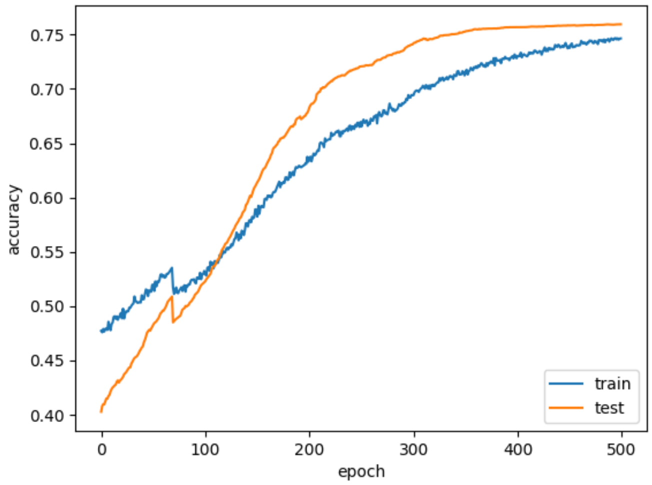


Figure 22 Accuracy trend as dependent on the number of epochs for the data set with the marker 'SPIKE_2' combined with the data set from 2019

Considering all markings, the maximum test accuracy is 76.46% and the train accuracy 76.88%. The results were reached using 500 epochs and a test size of 0.5. No significant differences in performance were observed when varying the test size.

test_size	epochs	batch_size	train_accuracy	test_accuracy
0.5	500	64	76,88%	76,46%
0.4	500	64	76,44%	75,95%
0.3	500	64	76,25%	75,24%
0.2	500	64	76,41%	75,4%

Modification of the epoch size leads to a decrease in accuracy but an general increase in sensitivity.

test_size	epochs	batch_size	accuracy	sensitivity	specificity
0.4	500	64	75,95%	0,23%	99,73%
0.4	250	64	74,96%	43,01%	85%
0.4	100	64	58,85%	18,49%	71,52%

8 Discussion

The created CNN reached high accuracy for both the simulated data and the collected data set. Considering only simulated data, the algorithm is able to achieve an accuracy, sensitivity and specificity of at least 99.44% when the signal-to-noise ratio (SNR) is greater or equal to ten. It offers further important research approaches, such as evaluating which minimal SNR can be used for a good performance. Furthermore, it can possibly explain which ratio of IED and normal samples should be used for a balanced classification. Thus, this simulated data may serve for performance evaluation of the CNN and potentially be combined with actual recorded IEDs. This would replace the method of increasing the data set described in the chapter 5.4.1 and possibly lead to an improvement in sensitivity and accuracy as described in the cited paper [LTCvP21].

Using the CNN on the data set from 2019, most of the parameter choices did not achieve high values. One reason for this may be that training 1052 neurons of the CNN with a maximum of 5569 samples is not sufficient for a complex IED detection. Nevertheless, an optimal overall accuracy of 82.75% was achieved in 1000 epochs.

The application of the algorithm to the data from 2021 leads to high specificity and accuracy. A maximum accuracy of 98.55% was achieved. However, the sensitivity is very low. A potential factor might be the low number of IED samples, so that the CNN is not able to learn the appearance of a spike.

The combination of both EEG recordings created a balanced data set. It contains enough data for training the algorithm and provides enough samples with an IED. Both potential described problems have been solved and a stable performance has been observed. Compared to the individual use of the data, the accuracy only decreased by a maximum of 8%, but the sensitivity increased by up to 77%. Besides, the performance parameters were in most

cases not significantly influenced by changes in the CNN parameters.
It is also important to note that learning depends very much on the labelled input data. The best results are achieved with the 'SPIKE_2' labels, which leads to an accuracy difference of up to about 15% (Table 2).

Overall, the presented CNN reaches a high accuracy for all data sets. Using the data from 2019 and from 2021 with the union of the marker, an accuracy of 76.88% is achieved, this being the lowest value compared to the other combined data sets, but is still a good achievement. The highest accuracy value was obtained with the data set from 2021 and applying the individual markers. It was higher than the accuracy achieved by physicians in the visual detection of IEDs (accuracy = 88%, [KABea17]). Automatic IED detection can save a significant amount of time in the everyday clinical routine. It has been observed that the markings within physicians differ widely [KABea17], Table 1, thus, such an algorithm could serve as a conservative evaluator. In the paper [KABea17], the lack of an automatic spike detection tool with high specificity is highlighted. Such a tool would allow physicians to subsequently review the IED samples that are flagged too conservatively as spikes.Therefore, it would be necessary to analyse only the samples that are classified as IEDs by the CNN.
Although the CNN is not very deep compared to other models applied in the context of spike detection, it achieves high performance parameter with a running time of only 30 minutes. Deeper networks require several hours of training [LTCvP21].

On of the limitation of the present study is that it is based on a single patient file. Fortunately, the algorithm was also able to classify simulated files. This suggests that other patient files can also be classified with the CNN. Consequently, one of the next steps would be to test the algorithm on different files.In addition, it could be helpful to analyse whether the CNN is able to distinguish patterns that look typical of epilepsy, such as wicket spikes, but do not occur in epilepsy. as a consequence, false diagnoses could be reduced.
Moreover, fine tuning techniques such as earlystopping by calculating an optimal epoch size can manifest the choice of parameters effectively. In this way, the algorithm could be optimised while simultaneously saving computational time.

References

[AH17] H.H. Aghdam and E. J. Heravi. *Guide to Convolutional Neural Networks, A Practical Application to Traffic-Sign Detection and Classification*. Springer, 2017.

[ARWea19] M. Antonakakis, S. Rampp, and C.H. Wolters et al. Individualized targeting and optimization of multi-channel transcranial direct current stimulation in drug-resistant epilepsy. *IEEE 19th International Conference on Bioinformatics and Bioengineering (BIBE)*, 2019.

[Bra] Brainstorm. `http://neuroimage.usc.edu/brainstorm`.

[BSea09] M. J. Brodie and S. C. Schachter et al. *Fast facts: epilepsy*. Abingdon, Oxford : Health Press Ltd., 2009.

[Cur] Curry. `https://compumedicsneuroscan.com`.

[ED11] A. Ebner and G. Deuschl. *EEG*. Thieme, 2011.

[HH18] C. F. Higham and D. J. Higham. Deep learning: An introduction for applied mathematicians. 2018.

[JHea19] J Jing and A. Herlopian et al. Interrater reliability of experts in identifying interictal epileptiform discharges in electroencephalograms. *American Medical Association*, 2019.

[JKea21] A. Jentzen and B. Kuckuck et al. Mathematical introduction to deep learning. summer term 2021.

[JRea21] X. Jiang and B. Risse et al. Mustererkennung. summer term 2021.

[KABea17] N. Kane, J. Acharya, and S. Beniczky et al. A revised glossary of terms most commonly used by clinical electroencephalographers and updated proposal for the report format of the eeg findings. *Clin Neurophysiol Pract*, pages 170–185, 2017.

[Kai22] F. Kaiser. Optimization of transcranial electrical stimulation montages to reduce seizure frequency and severity in patients with refractory focal epilepsy. 2022.

[LTCvP21] C. Lourenço, M. C. Tjepkema-Cloostermans, and M. J.A.M. van Putten. Efficient use of clinical eeg data for deep learning epilepsy. 2021.

[LTCvPea19] C. Lourenço, M. C. Tjepkema-Cloostermans, and M. J.A.M. van Putten et al. Deep learning for interictal epileptiform discharge detection from scapl eeg recordings. 2019.

[MAT] MATLAB. `https://de.mathworks.com/products/matlab.html`.

[MG14] J. W. Miller and H.P. Goodkin. *Epilepsy*. John Wiley & Sons, 2014.

[Mur12] K. P. Murphy. *Machine Learning*. Cambridge, MA :MIT Press., 2012.

[RN21] B Risse and A. Nienkötter. Computer vision. winter term 2020/21.

[Sl] Scikit-learn. `https://scikit-learn.org/stable/`.

[Ten] Tensorflow. `https://www.tensorflow.org`.

[tf.a] Tensorflow: tf.keras.layers.Dense. `https://www.tensorflow.org/api_docs/python/tf/keras/layers/Dense`.

[tf.b] Tensorflow: tf.keras.layers.Flatten. `https://www.tensorflow.org/api_docs/python/tf/keras/layers/Flatten`.

[Tra19] T.P. Trappenberg. *Fundamentals of Machine Learning*. Oxford University Press, 2019.

[Tre22] M. Trepel. *Neuroanatomie*. Amsterdam : Elsevier, 8 edition, 2022.

[WKea10] C. Watson and M. Kirkcaldie et al. *The brain: an introduction to functional neuroanatomy*. 2010.

Plagiatserklärung der / des Studierenden

Hiermit versichere ich, dass die vorliegende Arbeit über ____________________________
_____________________________________ selbstständig verfasst worden ist, dass keine anderen Quellen und Hilfsmittel als die angegebenen benutzt worden sind und dass die Stellen der Arbeit, die anderen Werken – auch elektronischen Medien – dem Wortlaut oder Sinn nach entnommen wurden, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht worden sind.

____
(Datum, Unterschrift)

Ich erkläre mich mit einem Abgleich der Arbeit mit anderen Texten zwecks Auffindung von Übereinstimmungen sowie mit einer zu diesem Zweck vorzunehmenden Speicherung der Arbeit in eine Datenbank einverstanden.

____
(Datum, Unterschrift)