Computation models for audiovisual attention decoding

Sara Enander

Louise Karsten

LUND UNIVERSITY

Department of Automatic Control

MSc Thesis
TFRT-6165
ISSN 0280-5316

Department of Automatic Control
Lund University
Box 118
SE-221 00 LUND
Sweden

Printed in Sweden by Tryckeriet i E-huset
Lund 2022

Abstract

When being in a noisy environment, a normal hearing person can manage to sort out background noise and focus on the attended source. This is something that a person with impaired hearing will struggle with, even when wearing a hearing aid. Research for developing intelligent hearing aids has not yet come up with a solution for solving this problem and more research is needed. This thesis uses data from experiments where a noisy environment is simulated. The test subjects are exposed to a monologue and a dialogue at the same time but are told to only focus on one of them. Using EEG and eye gaze data collected from these experiments as an input, different machine learning models are implemented to solve a binary classification task to predict whether a subject is attending to a monologue or a dialogue. The investigated models are support vector machine, multilayer perceptron, and convolutional neural network. The input to the models is time series arrays from either EEG signals or eye gaze data. For the support vector machine and the multilayer perceptron models, more compact representations of the time series arrays are used as inputs. The convolutional neural network performs best overall and reaches an average prediction score of 87% for all subjects when using inputs from all electrodes at the same time. When using one electrode at the time as input, and then averaging over all electrodes, the support vector machine performs best with an average accuracy of 78%. There is however a clear pattern in what regions of electrodes that succeed best with the classification task for all models. These are the electrodes at the temporal lobe as well as the sides of the front of the frontal lobe. It varies how long the trials need to be to get a decent accuracy for each model when EEG data is used. The support vector machine and the multilayer perceptron performs best for longer trials while the convolutional neural network performs best for shorter trials. For the eye gaze data, the support vector machine reaches the highest average score of 99%. The accuracy for the eye gaze data is not affected remarkably by decreasing the length of trials.

Popular science summary

Sara Enander & Louise Karsten

Intelligent hearing aids hope to amplify only the sound a person is attending to. This thesis uses machine learning to see if there is a connection between brain signals, eye gaze and the attended source.

Imagine being in a crowded space, surrounded by lots of talking people. Standing in the same location, people with normal hearing can easily shift focus from one talking person to another. Somehow their brains manage to sort out the sound they want to listen to. The ability of the brain to perform this selection has been investigated but it remains unclear how it succeeds to do it. If a person has impaired hearing the ability of sorting out unwanted sounds is affected. This leads to hearing problems in noisy situations even though the person is using a hearing aid. Research for developing intelligent hearing aids has not yet come up with a solution for solving this in a good way and more research is needed.

One step towards intelligent hearing aids is to find a way to understand what a person is attending to. This thesis uses data from brain signals (EEG) and eye gaze to see if a machine learning model can predict whether a person is attending to a monologue or a dialogue. It is investigated what model is most successful with this prediction. The data comes from experiments where the test subjects are exposed to a monologue and a dialogue at the same time but are told to only focus on one of them. For the application of integrating this into a hearing aid device it is also interesting to investigate how few electrodes of the EEG that can be used, and also how short the trials that are being presented to the model can be. Another thing being investigated is what parts of the brain that are most important when predicting what a person is attending to. Three different models are investigated for this task: support vector machine, multilayer perceptron and convolutional neural network.

A convolutional neural network performs best overall and reaches an average prediction score of 87% for all subjects when using inputs from all electrodes at the same time. When using one electrode at the time, it performs worse than the support vector machine and multilayer perceptron that reaches average scores above 70%. There is however a clear pattern in what regions of electrodes that succeed best with the classification task. These are the electrodes at the temporal lobe as well as the sides of the front of the frontal lobe. It varies how short the trials can be before the accuracy decreases for each model when EEG data is used. The support vector machine and the multilayer perceptron performs best for longer trials while the convolutional neural network performs best for shorter trials. For the eye gaze data, the accuracy is not affected remarkably by decreasing the length of trials.

Acknowledgements

We would like to thank all our supervisors: Bo Bernhardsson, Emina Alickovic, Martin Skoglund and Johannes Zaar for the help and support throughout this project. Bo was our academic supervisor. Besides always showing a lot of interest in this project, he has helped with many practical issues such as getting access to a computing resource, and knowing what and when to hand in things. Emina, Johannes and Martin were our supervisors at Eriksholm Research Centre. Emina helped us with her insight in EEG and previous research. She also managed to reduce the number of figures in the result section from 26 to 11 which we thought was very impressive. Johannes gave us great insight in the data, we are especially thankful for the time he noticed that all our labels were wrong. Martin helped us overall, with mathematical notations among other things. It was always nice to start every week with a Monday meeting with all of you.

Our examiner Kristian Soltesz seems to have appreciated our Friday updates that we mailed out every Friday. We are thankful for the always positive spirit in his responses. Other recipients of the Friday updates were, except for everyone mentioned above, our friends and PhD students Johanna and Oskar. We got great help and inspiration from both of them.

Two other master thesis students that worked with the same dataset were Nelly and Viktor. We would like to thank them for great teamwork throughout the project but mostly for helping us with our computer resource struggles. The computations were enabled by resources provided by the Swedish National Infrastructure for Computing (SNIC) at C3SE partially funded by the Swedish Research Council through grant agreement no. 2018-05973. We would also like to thank all the nice people at the Control department at LTH, particularly the genius who programmed the coffee machines to always have fresh coffee at 10 and 3 o'clock.

Besides all the people helping us with this thesis project, we would like to thank our friends and families for love and support! We would especially like to thank some of our classmates that have made all these 5 years of studying fun. Ebba, Felicia, Isabelle, and many others; We did it!

Contents

Acronyms

AAD	Auditory attention decoding
ANN	Artificial neural network
CNN	Convolutional neural network
EEG	Electroencephalography
EOG	Electrooculography
LOOCV	Leave-one-out cross validation
MEG	Magnetoencephalography
MLP	Multilayer perceptron
STD	Standard deviation
SVM	Support vector machine

Electrode labels

C	Central
F	Frontal
Fp	Pre-frontal

O	Occipital
P	Parietal
T	Temporal

1

Introduction

Imagine being in a crowded space, surrounded by lots of talking people. Standing in the same location, people with normal hearing can easily shift focus from one talking person to another. It is also possible for them to attend to a group discussion even though there is more noise in the background. Somehow their brains manage to sort out the sound they want to listen to. The ability of the brain to perform this selection has been investigated but it remains unclear how it succeeds to do it. This problem is usually referred to as the cocktail party problem [Cherry, 1935]. If a person has a hearing disorder or impaired hearing, the ability to neglect unattended sounds is also affected. This leads to hearing problems in noisy situations even though the person is using a hearing aid. To reduce background noise and improve the signal-to-noise ratio, there are different methods used in hearing aids today. Adaptive bilateral beamforming microphone arrays that can enhance sounds from different directions [Picou and Ricketts, 2019] and single-microphone noise reduction (also known as digital noise reduction) [Chong and Jenstad, 2018] are two examples of these methods. They have been shown to improve listening comfort and reduce effort of listening and tiredness in noisy conditions. This works quite well when the noise is different from the attended sound but falls short in multi-talker environments where the noise has the same sound profile as the attended sound [Andersen, 2021]. This is one reason why many people with hearing loss still choose not to wear their hearing aid [Accessible et al., 2016]. Consequently, there is a need to keep improving hearing aids and make them more comfortable to improve speech understanding further [Lunner et al., 2020].

One step towards resolving the cocktail party problem for hearing aids is to understand how the brain manages to sort out the sound of the attended sources from the sound of the unattended sources. This process is referred to as auditory attention decoding (AAD) [O'Sullivan et al., 2015]. Research has been focused on measuring the changes in cortical activity that track the changes in speech stimulus. This has been done by mapping the amplitude envelope of speech to brain responses recorded with electroencephalography (EEG) instruments [O'Sullivan et al., 2015].

Several methods have been investigated to register the cortical activity, such as electrocorticography [Mesgarani and Chang, 2012], magnetoencephalography (MEG) [Ding and Simon, 2012] and magnetic resonance imaging [Satheesh Kumar and Bhuvaneswari, 2012]. These methods are however expensive, not very common nor portable. In comparison, EEG is a very accessible, cheap, and easy method to use. EEG can be integrated in everyday devices, which is a huge advantage for research of hearing aids and is a main reason why research in the field of AAD has been focused on EEG. O'Sullivan et al. (2015) showed that unaveraged single trial EEG data could be used to decode auditory attention (i.e., to identify the attended talker in an environment with multiple competing speakers). The authors also presented that the speech representation strength in the EEG was correlated with the performance of the subject for a task concerning the cocktail party problem.

Another method for measuring responses from how subjects react to difficult listening situations is to measure eye gaze instead of brain signals. Studies have been conducted to investigate whether there is a measurable effect of the eye position in an environment where a cocktail party problem can occur [Lu et al., 2021; Shiell et al., 2021]. The performance of attending to a source in this situation has been proven to improve when the eye fixation is on the attended source and reduced when it is not [Best et al., 2020]. Measuring eye gaze can be done by wearing eye tracking glasses and collecting data from the way the eyes move. Eye features such as how fast the eyes move and the changes in the direction of the movement can then help in figuring out what kind of source the person is attending to [Groner and Groner, 1989].

If there are patterns in brain and eye activity combined with what conversation the subject is focusing on, there is a possibility of being able to predict what sounds that come from attended sources and unattended sources respectively [Bednar and Lalor, 2020]. This could be a part of making hearing aids better at filtering out unwanted noise and enhancing the attended sounds. To succeed with this, the possible patterns must be found, and a valid prediction has to be made from unseen data. A common way of finding and predicting patterns is by using different machine learning models [Schirrmeister et al., 2017]. This is done by training a model on available data to adapt the model to the specific kind of data so that it will be able to make correct predictions when new data is presented. Different kinds of machine learning methods are good at predicting and finding different kinds of patterns. Therefore, it is important to investigate what kind of model is best suited for the specific task.

Previous research on the AAD has shown that a neural network performs better than linear models when predicting speech envelope from attended and unattended speakers [Taillez et al., 2020]. Machine learning methods have also been used in other fields related to EEG classification, such as epilepsy seizure prediction, with quite good results [Alickovic et al., 2018; Rasheed et al., 2021]. To investigate how attention decoding was affected by the position of attended and unattended speakers, a more realistic setup was created by Das et al. (2018) where the sound sources were positioned at different locations. Additional noise was added and varied as well. It was reported that both the location of the sound sources and the level of noise impacted the accuracy. It has also been shown that the number of electrodes needed in order to obtain a good accuracy could be decreased to an extent if there is sufficient training data [Mirkovic et al., 2015; Montoya-Martínez et al., 2021].

Evidently, studies regarding hearing in noisy environments are of great importance to hearing aid research [Lunner et al., 2020]. Machine learning algorithms applied to data from brain and eye activity could be a way to improve today's hearing aid technology so that speech understanding is improved. This thesis aims to use machine learning to investigate if it is possible to find correlation between brain activity recorded using EEG as well as eye gaze data and whether the subject is attending a monologue or a dialogue. It focuses on answering the following questions:

1. *If all scalp electrodes (*$K = 64$ *in this study) are used, what model makes the best overall prediction?*
2. *How short can each trial be before the accuracy decreases significantly?*
3. *How few electrodes can be used in order to make a good prediction, and in what region of the scalp should these electrodes be located?*

2

Auditory attention decoding

2.1 Brain signal processing

The brain receives information from all our senses, it connects and processes the information and then sends impulses to the rest of the body. Depending on what senses that are activated and how they are processed, different parts of the brain are used. The largest part of the brain is the cerebrum, and this is split in two hemispheres, one on the left and one on the right. Each hemisphere is used for different tasks. The left hemisphere is used more for logical reasoning and abstract representations, while the right hemisphere interprets spatial conditions and emotions [Nationalencyklopedin, 2022b]. The outer neural layer of the cerebrum is called the cerebral cortex and is divided into four lobes in each hemisphere. These lobes are named frontal, parietal, temporal, and occipital. The frontal lobe is the biggest one and is, as the name suggests, located in the front and top part of the brain. The parietal lobe is located in the back top of the brain, the temporal lobe is the area on the sides and the occipital lobe is located in the far back. The cognitive functions as well as voluntary movements are connected to the frontal lobe and the parietal lobe handles touch and temperature. Hearing and vision are connected to the temporal and the occipital lobe respectively [Nationalencyklopedin, 2022b].

When a person sees someone speaking, a signal from the receiving retina in the eye is sent to the visual centre in the occipital lobe. To understand what the colours, contours, and movements that the visual centre distinguishes, signals are sent further on to the temporal lobe [Nationalencyklopedin, 2022c]. If wanting to understand what the person is saying, these signals are sent to the left hemisphere since this is used for interpreting speech and language. The right side is in charge of more cognitive functions such as visuospatial and social cognition [Bernard et al., 2018].

When a nerve impulse in the brain is activated it means that there is a change in the membrane of the sending and receiving nerve cell as well as connected cells. This leads to a flow of positive ions such as sodium and potassium [Nationalencyklopedin, 2022b]. Each of these ions contribute to a small electrical activity. Since large groups of nerve cells are synchronized and oriented perpendicular to the scalp, the outer nerve cells of the cerebral cortex get a summed potential difference from activated nerve cells beneath. A method to measure this summed electric activity present in these outer nerve cells is EEG [Constant and Sabourdin, 2012].

2.2 EEG

To measure the brain activity with EEG, electrodes are placed on the scalp. These are often used together with conducting gel or electrolytic water [Humanities lab, Joint faculties of humanities and theology at Lund University, 2022]. The electrodes are then able to sense the potential difference from activated nerve cells beneath and perpendicular to the scalp. This technique can detect signals continuously in the frequency interval 0.3-40 Hz that can be divided into five frequency rhythms: delta, theta, alpha, beta and gamma [Abo-Zahhad et al., 2015].

The first interval, the delta rhythm, has frequencies between 0.5-4 Hz and is observed from deep sleep. A bit higher frequencies, 4-8 Hz, are reached when the subject sleeps lightly and this span is called the theta rhythm. The next frequency interval is the alpha rhythm that contains the frequencies 8-14 Hz. This rhythm is observed from a relaxed subject, for example when meditating. The fourth interval, the beta rhythm, has frequencies between 14-30 Hz and is reached when a subject is actively thinking. All frequencies above 30 Hz are in the so called gamma rhythm. These frequencies are connected to visual stimulation [Abo-Zahhad et al., 2015].

There are different variations within EEG, where it is possible to vary the number of electrodes, the placement of them and the method how to place them on the scalp, depending on the objective. Figure 2.1 illustrates an EEG method called the 10:20 electrode system. Each presented combination of letter and number represents an electrode. The letters stand for what parts of the brain they detect signals from, Fp is pre-frontal, F is frontal, T is temporal, P is parietal, O is occipital, and C is central. The electrodes labelled with A are located between Fp and F [Lotte et al., 2015].

As seen in figure 2.1, the distance between the electrodes is 10% or 20% of the distance between the front and the back of the skull. A common EEG setup is a 10:20 system with 64 electrodes. This means an extra row of electrodes between each of the illustrated lines seen in the top of head view in figure 2.1 as well as added electrodes between the already existing lines. The 10:20 system is an internationally recognized method and due to the standardization, it is possible to reproduce conducted studies and compare subjects to each other [Khazi et al., 2012].

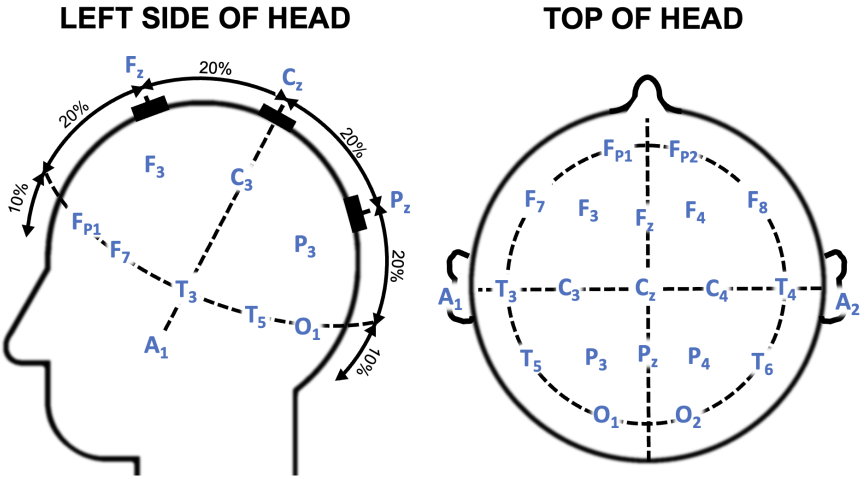


Figure 2.1 Placement of electrodes in the 10:20 system of EEG. The number of electrodes used can vary, for a 64 electrode system an extra row of electrodes between each of the illustrated lines seen in the top of head view are added as well as added electrodes between the ones on the already existing lines.

A drawback of EEG as a method to investigate brain signals from sensory inputs, compared to for example MEG, is that it collects a lot of noise. For example, blinking or movement of eyes or muscles will appear in the collected data. This leads to a need of pre-processing the data as a first step before it is ready to be used in further analysis [Palendeng, 2011].

The analysis of EEG data is usually visualised by means of brain electrical topography. Topography is done by creating a heatmap where the activity of each channel is represented by a colour on its corresponding place on a 2D plot of a head. Depending on the strength of the electrical activity measured from the channels, the colours are distributed along a spectrum between red and blue.

2.3 Eye tracking

Eye movements can be measured in different ways, such as screen based remote devices, wearable devices [Tobii, 2022a] and electrooculography (EOG) [Creel, 2019]. Remote devices can for example stand on a desk and be connected to a screen. They are advantageous to use in a setting where a subject is reacting to or interacting with the screen. These kinds of trackers are also quite robust to head movements and are therefore usually a good option for people with certain medical conditions and for young children. These eye trackers also have a wide range of sampling rates. Since a very high sampling rate is possible, a lot of very detailed data can be collected [Tobii, 2022a]. Wearable devices include glasses or VR-headsets. These are best used when moving around in an environment and doing activities such as interacting with others, shopping in a store or playing sports [Tobii, 2022a]. The quality of the data collected from wearable eye tracking devices is usually lower than for desktop devices since they need to be compact and light. They are also prone to shifting if the subject is moving around a lot [Tam, 2019]. Both remote and wearable devices are performing eye tracking based on invisible near-infrared light that is directed towards the eye. The direction of the reflection of this light is then recorded and from that the exact position of the eye and the direction where the eye is focusing can be determined [Tobii, 2022a]. Another eye tracking method is EOG, where electrodes are placed on the skin on each side of the eyes. The potential difference between these varies when the eyes move, and therefore eye movements can be detected [Creel, 2019].

When measuring the eyes' movements, the direction can be represented by yaw and pitch angles. The yaw angle represents horizontal rotation of the eyes, and the pitch represents vertical rotation. These angles can be used to detect for example when a person is switching gaze from one person to another in a conversation.

2.4 Signal representation

EEG data consists of time series data, $X_t = \{x_1, x_2, .., x_T\}$ where $t = \{1, 2, ..., T\}$ denotes each measured time point. The time series data belong to a channel each, $C = \{X_{t1}, X_{t2}, .., X_{tK}\}$ where K denotes the number of electrodes, and each electrode is represented by one channel. Every channel is a part of a trial, $Z_i = \{C_1, C_2, .., C_L\}$ where $i = \{1, 2, ..., L\}$ and L is the total number of trials. All trials are used for all subjects, $S_j = \{Z_{i1}, Z_{i2}, .., Z_{iM}\}$ where M denotes the number of subjects. The eye gaze data also consists of data stored as similar time series X_t, containing the values of the yaw angle. It only consists of one channel, $K = 1$ and has the same number of trials, and same number of subjects as the EEG data.

When measuring the brain signals there is also a lot of noise that gets in the way. Therefore, the EEG signal does not fully represent the true signals from the brain. The signal acquired by the electrodes can be modelled as presented in equation (2.1). Here the true brain activity is denoted as a_t, B is the linear mapping from the brain activity to the EEG, e_t is the noise from measurement and X_t is the measured signal.

$$X_t = Ba_t + e_t \tag{2.1}$$

To use the data for classification, there are many different approaches. For EEG, it is possible to investigate the behaviour of all electrode channels, C, or only one at a time, C_i, both using the full time series array. The data can also be split into smaller sizes $X_{T/n}$ or be converted into a more compact representation, by creating different features. These features could be calculating the mean value of the time series or the spread of the data to detect how it behaves more generally. The eye gaze data does not contain multiple channels, but the eye gaze data can also be converted into features in the same way as the EEG data.

3

Machine learning methods for audiovisual attention decoding

3.1 Concept

To understand the concept of machine learning, it can help to compare it with how humans learn. From the moment we are born the learning process of learning a language begins. We keep receiving information from all our senses and use this to understand the world around us. By listening to the language continuously and by starting to use words, one at a time, we slowly but steady build up our own dictionary. This learning process continues throughout our entire lives, as we encounter new surroundings where new combinations of words are needed to grasp and explain the world. Not only does the library of words that we carry help with understanding the world, all experiences that we have had during our lifetimes will also play a part in making decisions of language usage. We will perhaps understand that a certain scenario requires a more polite sentence than other informal words containing the same message.

Machine learning is when this learning process is implemented for a machine instead. Starting from zero, it receives information and creates an understanding of its world. If given enough knowledge and training, it will manage to make valid predictions. These predictions are done on previously unseen data, such as finding a good word describing an object the machine encounters. Depending on each situation, a different amount of knowledge as well training is needed. To make the learning process as efficient as possible, a lot of varied knowledge together with as little training as possible is preferred [Bhagat, 2021].

3.2 Classification and prediction

A common machine learning task is to predict between different class labels y_i, for inputs that can for example be a feature vector f_i, an image or time series X_t. For example, a model could get an image of a handwritten digit as an input and the task to classify what digit the image represents. The classes are then numbers between 0 and 9, so the class labels are $y_i = (0,1,2,...,8,9)$. Each digit consists of many different features such as circles, corners, or straight lines that could be included in the feature vector f_i. The entire image can also be used as input, without creating features first.

The feature vector, image or time series data is then the input to a model, that performs a nonlinear processing, here denoted h, to get one class prediction as output, y_i. The kind of processing depends on what model is used. This is described in equation (3.1).

$$y_i = h(z), \quad z = f_i \;\; or \;\; X_t \tag{3.1}$$

By letting the model study a lot of data, it is possible for it to gain an understanding of what features are needed to classify between the classes. In the same example as earlier, the images of handwritten digits, some digits might be more cursive, thicker, thinner, or a bit vague. Therefore, it is important to have a large and varied dataset to train the algorithm with as well as validating on unseen data.

A problem a machine learning algorithm might experience is that it gets too much training. This leads to an algorithm that is not generic enough to predict classes correctly when being exposed to previously unseen data, because it is too adapted to the already seen data. This concept is called overfitting. To avoid this, it is important to have both a decent amount of training data as well as validation data. There are multiple different methods on how to perform this split of the training dataset. Two methods that can be used to prevent overfitting are cross validation and early stopping.

Cross-validation

The method of splitting the dataset into a training and a validation part can be done in many ways. By cross-validating the data, the datasets are crossed over so that each data point has a chance of being validated against [Refaeilzadeh et al., 2016]. The performance of cross-validation can be measured by using different metrics, in this thesis only accuracy is used. The accuracy is the proportion of correctly classified samples among all samples. When cross-validation is used this metric is calculated from the average of all validation accuracies.

A simple version of cross validation is the so called k-fold cross validation and is illustrated in figure 3.1. Here, the sample called *Test* is used for validation and the rest of the data for training. In k-fold cross validation, the data is iterated over itself k times. Each time, the data is split into k parts, where one part is left out during training and then used for validation. After this, a new part is selected, and a new validation is performed. This is iterated until all parts of the data have been used both as train and validation data.

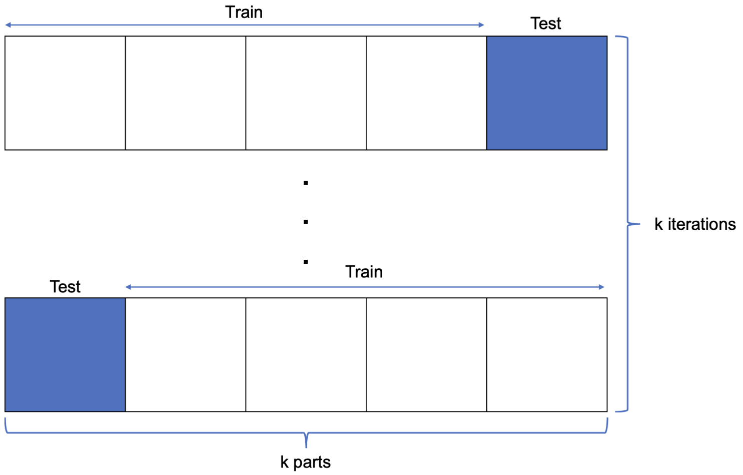


Figure 3.1 Illustration of k-fold cross validation, where the data is iterated over itself k times.

Another version of cross validation is the leave-one-out cross validation (LOOCV). It is a special case of k-fold cross validation where k is set to the number of data points in the dataset. This means that only one sample is used for validation each iteration, and the rest for training. The data is then iterated over until every data point has been used for validation. When using LOOCV there are more available training samples for each fold, which is good, especially if there is limited training data available. The drawback of LOOCV is that the computational cost and time increases since the data is iterated over many times.

Early stopping

When training a model, the accuracy will start at a very low level, and then increase the more the data it is trained on. The model can however get too familiar with the data and start to classify using features and connections very specific to the training data, that do not appear in the validation data. If this happens, the accuracy on the training data will be very high, but when the network or model is presented to new data that is seemingly similar to the training data it will perform badly.

To avoid this, it is a good idea to monitor the error for the training and validation data during training. The error measures how much the prediction deviates from the ground truth. When the training error is still decreasing, but the validation error starts to increase the training should stop to prevent overfitting from happening. An illustration of this is seen in figure 3.2.

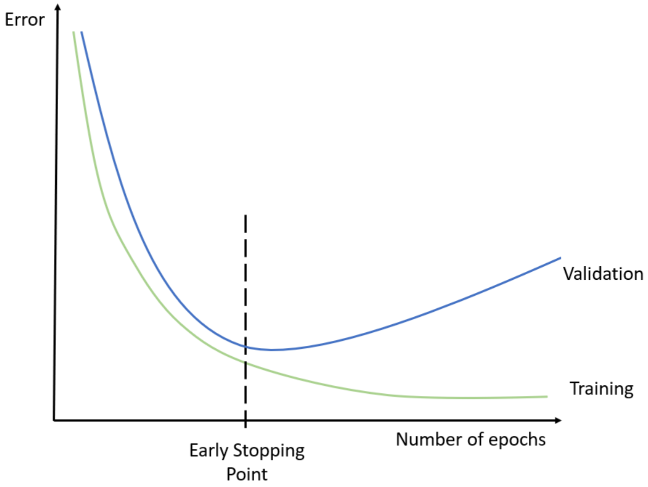


Figure 3.2 Example of early stopping of the training when the training error is still decreasing, but the validation error starts to increase.

Machine learning models

The machine learning models used in this thesis are support vector machine (SVM) and artificial neural networks (ANNs). For the ANNs, both multilayer perceptron (MLP) and convolutional neural network (CNN) are used. A CNN usually contains more layers than a simple MLP, as well as different types of layers. SVM has been used widely for EEG data classification in many previous studies [Richhariya and Tanveer, 2018], and it is therefore of interest to investigate the SVM in this thesis too. Previous research shows that a CNN is a successful model using EEG data as an input [Lawhern et al., 2018; Waytowich et al., 2018; Schirrmeister et al., 2017] and this thesis therefore analyses five different CNNs. As mentioned earlier, to avoid overfitting a machine learning model should be as generalised as possible. It is also better to use a simpler model when possible to reduce computational cost. Therefore, an MLP, which is a more simple version of an ANN than a CNN, is also implemented in this thesis.

3.3 Support vector machine

Many machine learning algorithms aim to classify the data by separating it into different classes. SVM is one of these methods and it uses a hyperplane to separate the data. A hyperplane is a plane with a dimension that is one order lower than the surrounding space. For example, if the surrounding space is of two dimensions, then the hyperplane is of one dimension, consequently a line. If the surrounding space is of three dimensions, then the hyperplane is a plane since it will be in two dimensions [Subasi and Gürsoy, 2010].

Figure 3.3 shows a 2D example of the minimum distance **d** from any point **x** to the hyperplane that is defined by equation (3.2). As presented in the figure, **w** in equation (3.2) is a weight vector and together with the constant *b* and the variable *x* it forms a line [Weinberger, 2018].

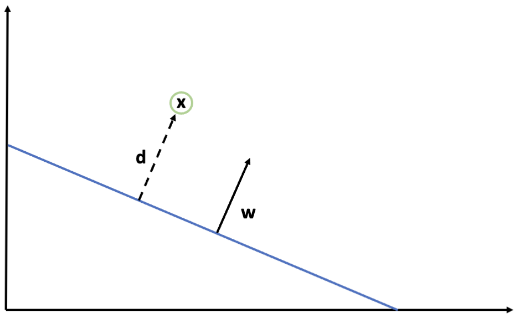


Figure 3.3 A 2D example of the minimum distance from any point to a hyperplane. **d** is the minimum distance from the point **x** to the hyperplane (blue). **w** is the weight vector that together determines the slope of the line.

$$\mathbf{w}^T x + b = 0 \tag{3.2}$$

Calculating the distance **d** from any point **x** to the hyperplane is done according to equation (3.3) [Weinberger, 2018]. Here, $||\mathbf{d}||_2$ and $||\mathbf{w}||_2$ are the Euclidean norms of the distance and weight vectors respectively. The distance between the closest data points of each class is called the margin. The points that are closest to the hyperplane are called support vectors and are the most difficult points to classify [Gandhi, 2018].

$$||\mathbf{d}||_2 = \frac{|\mathbf{w}^T x + b|}{||\mathbf{w}||_2} \tag{3.3}$$

The optimal hyperplane is when the margin is maximised, in other words when the distance from the points to the hyperplane is minimised. By finding the point **x** that minimises the distance from equation (3.3), the maximum margin can be found [Weinberger, 2018]. When it is impossible to separate the data linearly, a so called kernel can be used.

A kernel manages to map the data to a higher dimension where it may be possible to separate the data linearly. The mapping is explained by the definition in equation (3.4), where K is the Kernel function that maps the points x and y from the original space via the feature map ϕ. It returns the dot product of the vectors in the feature space [Wilimitis, 2018]. The mapping done by the kernel function is illustrated in figure 3.4. Here the kernel function is used to map the data points to a higher dimension where the data points are linearly separable.

$$K(x,y) = \phi(x) \cdot \phi(y) \tag{3.4}$$

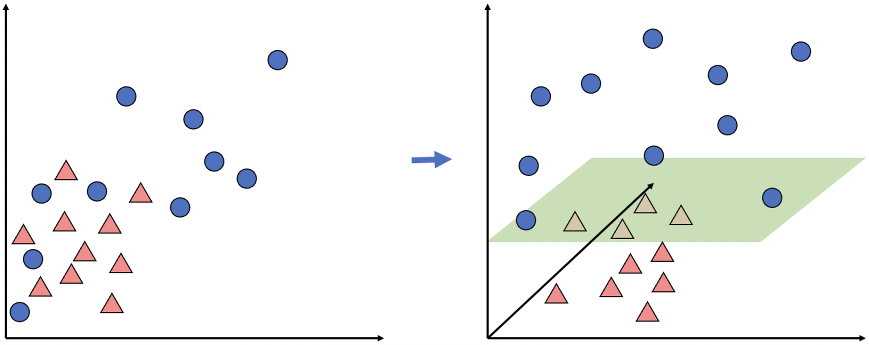


Figure 3.4 Mapping points to a higher dimension to make it possible to linearly separate the data.

Proposed SVM model

To investigate the ability of an SVM to classify whether a person is attending to a monologue or a dialogue, the module scikit-learn was used [Pedregosa et al., 2011]. This module provided components, so that the SVM model did not have to be implemented from scratch. The classifier was created by using the module's grid search component and the built in LOOCV function. By trying all possible kernels with the grid search, the optimal SVM kernel based on the input could be identified. The input consisted of a feature representation, f_i, of the time series arrays, X_t. Then, an SVM classifier with the found parameters from the grid search could be created. This was then used to train and validate on the data using cross-validation.

3.4 Artificial neural network

An ANN is a computational model that is inspired by how computations happen in the brain. It uses many of the same concepts as the brain, but in a reduced number [Walczak and Cerpa, 2003]. An artificial neuron fires when a linear combination of the inputs to that neuron exceeds a predecided threshold. When many of these neurons are connected in a network an ANN is created [Russell and Norvig, 2016].

An ANN is built from nodes connected by edges. The edges propagates the signal, from one node to the next. The signal is denoted a_i, where i represents the nodes the signal comes from. The strength and sign of the connection is decided by a numeric weight, w_i. In each node the weighted sum of the inputs, s, is computed. This computation for node j is presented in equation (3.5). An activation function, φ, is then applied to this sum, to obtain the output from that node, a_j, as presented in equation (3.6) [Russell and Norvig, 2016].

$$s_j = \sum_{i=0}^{n} w_{i,j} a_i \tag{3.5}$$

$$a_j = \varphi(s_j) = \varphi\left(\sum_{i=0}^{n} w_{i,j} a_i\right) \tag{3.6}$$

The activation can look different depending on the function of the network. A hard threshold or a logistic function can be used, but it is an advantage to use a nonlinear function. If the activation function is linear the network will not be able to learn complex patterns [Sharma et al., 2020]. The activation functions used in this thesis are presented in equation (3.7).

$$
\begin{aligned}
\text{Sigmoid} \quad & \varphi(x) = \frac{1}{1+e^{-x}} \\
\text{Hyperbolic tangent} \quad & \varphi(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}} \\
\text{Rectified linear unit} \quad & \varphi(x) = max(0,x) \\
\text{Softmax} \quad & \varphi(x) = \frac{e^{x}}{\sum_{j=1}^{n} e^{x_j}} \\
\text{Softplus} \quad & \varphi(x) = log(1+e^{x}) \\
\text{Elu} \quad & \varphi(x) = \begin{cases} x & x < 0 \\ a(e^{x}-1) & x \leq 0 \end{cases} \\
\text{Selu} \quad & \varphi(x) = \begin{cases} s \cdot x & x < 0 \\ s \cdot a(e^{x}-1) & x \leq 0 \end{cases} \\
\text{Exponential} \quad & \varphi(x) = e^{x} \\
\text{Softsign} \quad & \varphi(x) = \frac{x}{(|x|+1)}
\end{aligned}
\tag{3.7}
$$

The sigmoid function transforms values into the range between $0-1$ and is for example good to use in the output layer of a binary classifier. Similar to the sigmoid function is the hyperbolic tangent function, also called tanh, that instead transforms values between -1 to 1. The rectified linear unit, or ReLU, is linear for $x > 0$, and zero for $x < 0$ [Sharma et al., 2020] and is the most common activation function used in convolutional and deep learning models [Bharath, 2020]. It is more effective than other functions because all neurons will not fire at the same time [Sharma et al., 2020], and the fact that it is close to linear also makes optimisation simpler [Bharath, 2020]. The Selu and Elu functions are the same function, just scaled differently with the predefined parameters a and s [Keras, 2022c]. The three most commonly used functions of these are sigmoid, tanh and ReLU, and plots of these are presented in figure 3.5.

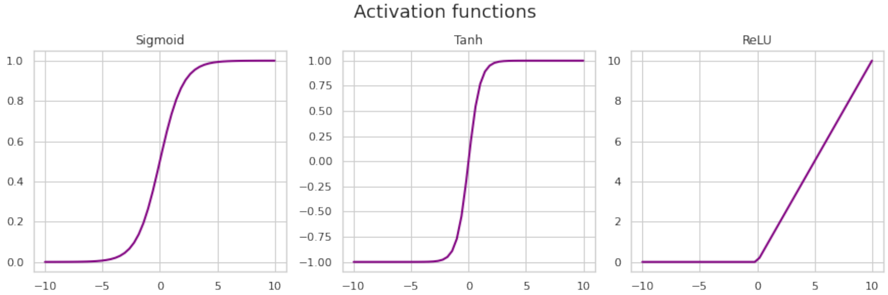


Figure 3.5 Plots of sigmoid, hyperbolic tangent (tanh), and rectified linear unit (ReLU), the most commonly used activation functions.

This thesis uses so called feed-forward networks. This means that the links in the network only go in one direction, from the input nodes to the output nodes. There are no links to previous nodes or any loops. In these kinds of networks, the nodes are usually arranged in layers. The nodes in each layer will therefore only get inputs from the nodes in the layer directly before. If there is only one input layer that directly connects to an output layer that network is called a perceptron.

Multilayer perceptron

When layers are added between the input- and output layers these are called hidden layers, and the network is called a multilayer perceptron. An example of an MLP with 3 input nodes, 2 output nodes and one hidden layer with 4 nodes is presented in figure 3.6.

Proposed MLP model

A few different versions of MLPs were created and tested on the data. The networks were created using Keras [Chollet et al., 2015] and contained one fully connected layer, also called dense layer. The number of nodes in the input layer were matched to the input data, and the output had one node to perform binary classification. To investigate which hyperparameters that would fit best to the data KerasTuner Hyperband tuner was used [O'Malley et al., 2019]. The hyperparameters that were tested were number of units and activation function. The number of units ranged from 1 to 100, activation functions that were tested were rectified linear unit, hyperbolic tangent, sigmoid, softmax, softplus, Selu, Elu, exponential and softsign. This was done for each channel, C_i for each subject S_k to get an optimal network for each channel. Early stopping was used to avoid overfitting. It was done by stopping the training if the validation error did not decrease for 5 epochs of training and was implemented by using the Keras callback EarlyStopping [Keras, 2022b]. After performing this Hyperband Tuning it was investigated which hyperparameters were chosen by the tuner most frequently and a new network was created from these hyperparameters.

The network that was kept for further analysis was a network with one hidden layer containing 80 nodes, ReLU activation function in the hidden layer and a sigmoid activation function for the output layer.

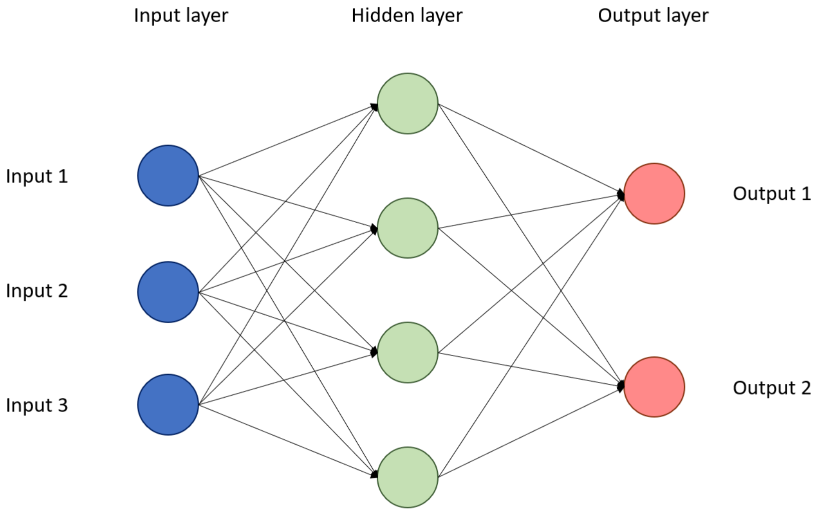


Figure 3.6 An example of a multilayer perceptron with one hidden layer and three input nodes, four nodes in the hidden layer and two output nodes.

Convolutional neural networks

CNNs are often used to classify images, because they manage to understand how the image is connected by capturing the spatial and temporal dependencies very well [Saha, 2018]. They can also perform well for other tasks, for example signal processing.

As the number of layers increases, the computational cost gets very high fast for fully connected layers. Therefore, instead of using only fully connected layers, CNNs use two additional types of layers: convolutional layers and pooling layers [O'Shea and Nash, 2015]. Convolutional layers are layers where each node only gets input from a specific region of nodes in the previous layer. To reduce computational costs further the same weights can be used for each of these regions. The layers work as filters or sliding windows that map the region of input neurons to one output neuron. This is depicted in figure 3.7. Depending on the weights, specific features can be detected in the input layer, regardless of position. Therefore, these layers are called convolutional layers [Albawi et al., 2017]. Each layer can then use different filters and the network can be specialised to its purpose. To make the computations faster, batch normalisation can be used. This method finds mean and variance of the current input and normalises it [Huber, 2020].

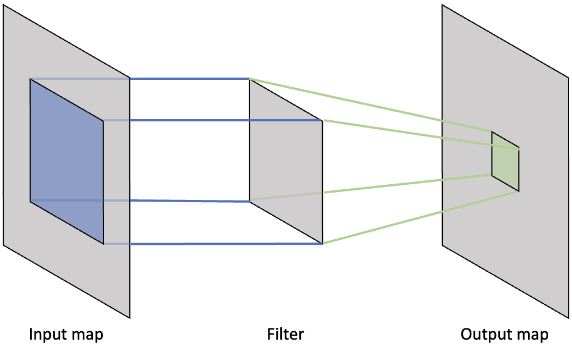


Figure 3.7 Example of a convolutional layer. It consists of an input map, a filter and an output map.

There are different convolutional layers such as depthwise convolutional layers and separable convolutional layers. For depthwise convolutional layers each input channel gets a single convolutional layer applied to it [Wang, 2022]. For a separable layer a depthwise convolution is followed by pointwise convolution. Pointwise convolution uses a kernel with a depth of the amount of input channels. This kernel iterates through all data points [Wang, 2022]. Convolutional layers can be implemented using for example Keras, where a common version is called Conv2D [Keras, 2022a]. It is also possible to reshape the input so that it fits the next layer. This can be done using different methods, two of these are using a permute layer and a flatten layer. The permute layer takes a pattern as an input and permutes the input dimensions accordingly [Keras, 2022d]. The flatten layer flattens the input so that it can be sent into a dense (fully connected) layer [Dumane, 2020].

The architecture of a very simple CNN is presented in figure 3.8, where a convolutional layer is followed by a pooling layer and then a fully connected layer. Another common way to structure a CNN is to use two convolutional layers before a pooling layer, and then repeating before having the fully connected layers last.

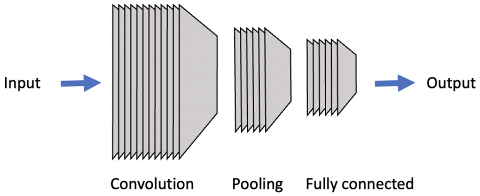


Figure 3.8 Typical architecture of a CNN where a convolutional layer is followed by a pooling layer and then a fully connected layer.

Pooling layers have the role of reducing the dimension of the network. Typically, these layers take a region of for example 2×2 nodes and map it to one node in the next layer, which results in a down scaling to 25% of the size of the previous layer [O'Shea and Nash, 2015]. To prevent the models from getting overfitted, dropout layers can be used. If using a regular dropout layer, some input elements are randomly set to 0. If instead using a spatial dropout layer, feature maps are dropped randomly [Keras, 2022e].

The dense or fully connected layers have the functions that are present in a normal ANN, they calculate class scores from activations to perform classification. Therefore, fully connected layers are typically put in the end of a CNN, since that is where the classification will take place [O'Shea and Nash, 2015].

Proposed CNN models

Five different CNNs were implemented from the code presented in the Army Research Laboratory EEGModels project [Army Research Laboratory, 2022]. All models were able to train on data from all 64 electrode channels at the same time. The channel number could be modified to 1 channel when using either one electrode at the time or eye gaze data as an input. All parameters and weights given from the source were used to be able to compare easier between the results in this study with the one presented in the article.

The first of the five models had the structure shown in figure 3.9. It consisted of two main blocks, where each block contained at least one convolutional layer. The first of the blocks contained a depthwise convolutional layer and the second one a separable convolutional layer. The model was implemented to match an input signal that was sampled at 128Hz [Lawhern et al., 2018].

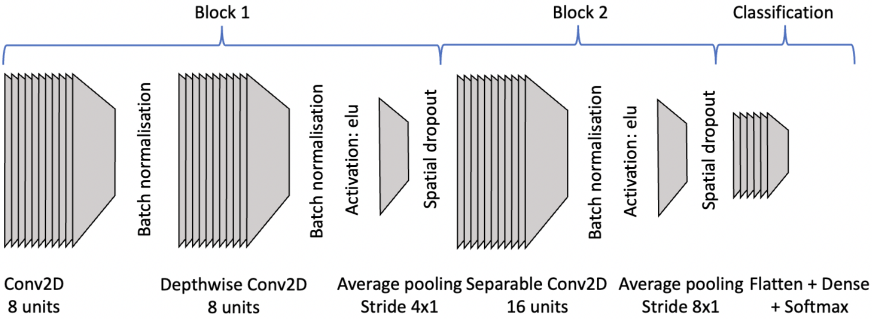


Figure 3.9 An illustration of the first CNN model presented in the Army Research Laboratory EEGModels project [Army Research Laboratory, 2022].

The second CNN model was the same as the first one except for the dropouts not being spatial dropouts but regular ones. The number of units in the convolutional layer was also higher for this model. The architecture is presented in figure 3.10 [Waytowich et al., 2018].

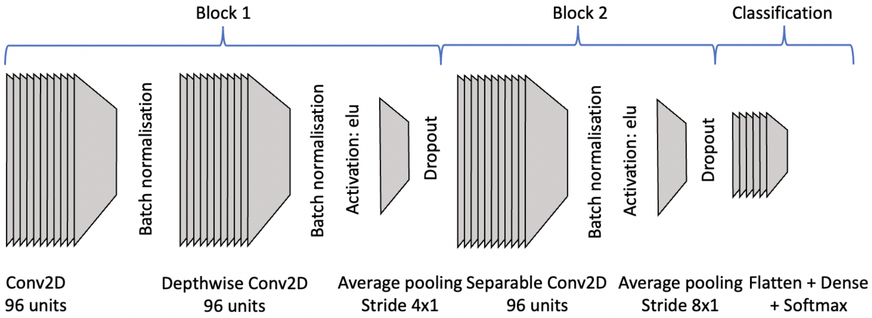


Figure 3.10 An illustration of the second CNN model presented in the Army Research Laboratory EEGModels project [Army Research Laboratory, 2022].

In the third model, there were three blocks containing convolutional layers before the classification that was done by flattening, then a dense layer and then using softmax. The convolutional layers contained 16, 4 and 4 units respectively and were separated by batch normalisation, the *Elu* activation function and dropout layers. Instead of using a pooling layer, the model used a permute layer, where permutations were done according to the pattern 2, 1, 3 [Lawhern et al., 2018].

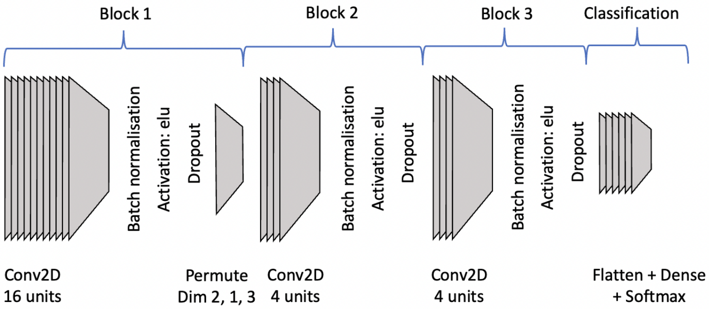


Figure 3.11 An illustration of the third CNN model presented in the Army Research Laboratory EEGModels project [Army Research Laboratory, 2022].

The fourth model was the most complex ones of the five different models since this one contained most layers. It was built up by four blocks, each containing at least one convolutional layer. The first block had two convolutional layers with 25 units each and the other had one convolutional layer with 50, 100 and 200 units respectively. Max pooling was used in between together with batch normalisation and the *Elu* activation function. The implementation matched a signal of 2 seconds sampled at 128 Hz [Schirrmeister et al., 2017].

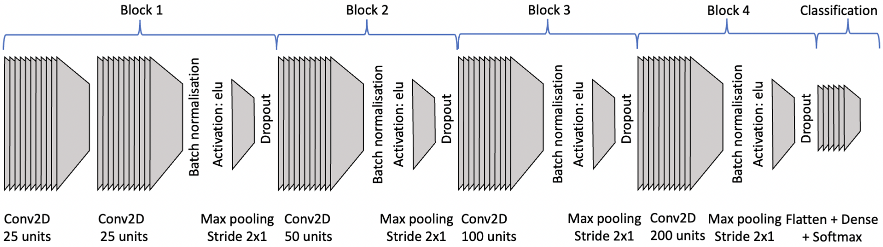


Figure 3.12 An illustration of the fourth CNN model presented in the Army Research Laboratory EEGModels project [Army Research Laboratory, 2022].

The fifth model consisted of one block containing two convolutional layers. It used batch normalisation, two different activation functions as well as an average pooling layer. The classification was linear and was done by flattening, and then using a dense layer and softmax. Just like the fourth model, the implementation was done to match an input signal of 2 seconds that was sampled at 128Hz [Schirrmeister et al., 2017].

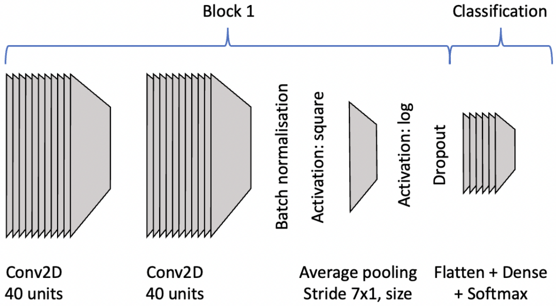


Figure 3.13 An illustration of the fifth CNN model presented in the Army Research Laboratory EEGModels project [Army Research Laboratory, 2022].

4

Dataset

The dataset used in this thesis was presented in the master thesis *Decoding Attention in Real-world listening* [Bilert, 2020]. The experiment was conducted at Eriksholm Research Centre and the data is not publicly available.

4.1 The experiment

There were $M = 18$ subjects in this experiment, 7 of those were female, and two left-handed. The handedness has a connection to what half of the cerebral hemisphere that is more dominant for different tasks, such as language. For about 95% of all right-handed people, language is left lateralised. For left-handed people, this percentage is instead 70% [Nationalencyklopedin, 2022a]. All subjects in the experiments were normal hearing and native Danish. The age range was 18 to 57 years, with all but two subjects in the range 18 to 32 years. The experiment was done in approval of Science Ethics Committees for the Capital Region in Denmark in accordance to the Declaration of Helsinki [Bilert, 2020].

The audiovisual (AV) stimuli were video clips where actors were sitting facing the camera. One person was having a monologue while simultaneously the two others were having a dialogue. These configurations of the actors were changed for each trial between two female and two male actors, so there were four actors in total, but only three in each trial. The clips were cut to be approximately 120 seconds each, which represents one full trial.

The subjects were seated in front of a big TV-screen where the videos of the actors were shown, and audio was played through 10 speakers in front of the screen. There were two different types of experiments conducted: baseline and main experiments, which used different videos. For the baseline experiments videos of a monologue and a dialogue were shown separately. During the main experiments the monologue and the dialogue were shown simultaneously, and multi-talker babble noise was added. Before the main experiment the subjects were told to focus on either the monologue or the dialogue and ignore the other. For the baseline experiment they were just told to focus on the film and pay attention to the kind of trial that was shown. After each trial a question about the content in the video was asked to the subject, to make sure that the subject would pay attention. Then the subject had to make a subjective rating of how the audibility of the trial was. Each subject did 34 trials each, of which 10 were baseline experiments and 14 main experiments.

The EEG was recorded using the ActiveTwo BioSemi device [BioSemi, 2022] with $K = 64$ electrodes placed using the 10:20 system as described in section 2.2. Head and eye movements of the subjects were also recorded during the experiment. This was done using the Vicon motion capture system with integrated Tobii pro glasses II for eye tracking [Vicon, 2022; Tobii, 2022b].

4.2 Pre-processing

The pre-processing of the data was conducted by Sascha Bilert when the data was collected and analysed for his master thesis [Bilert, 2020]. Both the EEG and the eye data had to be pre-processed to be usable for further analysis. A summary of how the pre-processing was done is presented below.

EEG

The EEG channels were re-referenced using the average of external mastoid electrodes, one on the left side and one on the right side. The data was then downsampled from 8192 Hz to 128 Hz. The power line artifacts were removed, and the EEG data was further down sampled to 64 Hz. The DC component was removed, and the eye-activity was controlled by referencing the Fp2 and AF4 electrodes (over the eyes). This made it possible to identify and remove eye-blinks artifacts. After this, the EEG signals from the trials were band-pass filtered between 1 and 9 Hz to match the frequency bands mainly driven by attention [Bilert, 2020].

Eye gaze

The eye gaze data was first separated from the head movements, and then eye-blinks were removed by using linear interpolation over the blink artifacts. After this, the data from each eye was analysed and either the left or the right eye was chosen for further analysis based on the quality of the data. In some cases, the quality index of both eyes was below 50% and in those cases that specific trial was excluded entirely. The data was then downsampled from 100 Hz to 64 Hz by using a low pass filter. From the unit gaze vector coordinates the yaw and pitch angles (horizontal and vertical rotational angles respectively) were calculated [Bilert, 2020]. Figure 4.1 shows the yaw angle over time from one of the subjects when attending to a dialogue. The time series of the yaw angle was the data used further in this thesis.

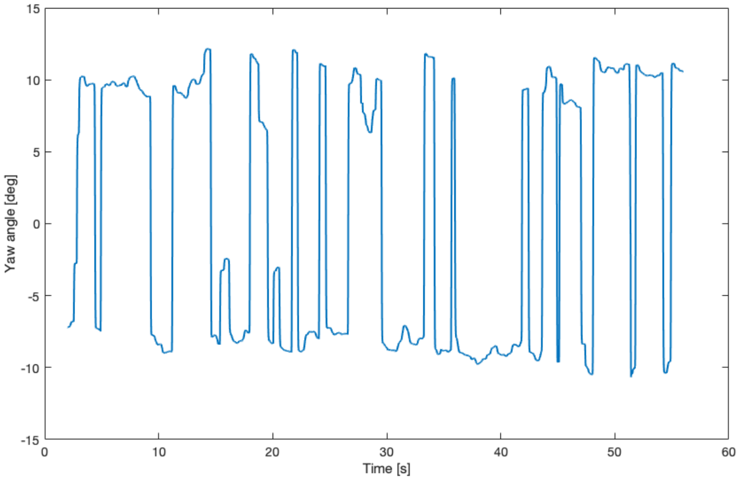


Figure 4.1 The yaw angle from subject 2 plotted over time when attending to a dialogue.

4.3 Selecting data

Subject 1 was used as a pilot subject to make everything run correctly, so it was removed entirely from the analysis. Subjects 3, 5 and 15 were not included due to bad quality of the EEG data. The subjects that remained and had good quality EEG data were subjects 2, 4, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16, 17 and 18. This sums up to $M = 14$ subjects in total with good EEG that were used for the experiments with the EEG data. For some subjects the eye data for some trials was not included, so when investigating the predictions from eye gaze data, these subjects were ignored. The subjects that were kept for eye gaze data were 2, 6, 7, 8, 10, 11, 13, 14, 16, 17, which is $M = 10$ subjects in total.

The EEG signals were expected to be stronger and more distinct when the subjects were focusing on either a monologue or a dialogue in the main experiment, compared to only one of them being present in the baseline experiment. Therefore, only the $L = 24$ main trials were used for each subject in this thesis.

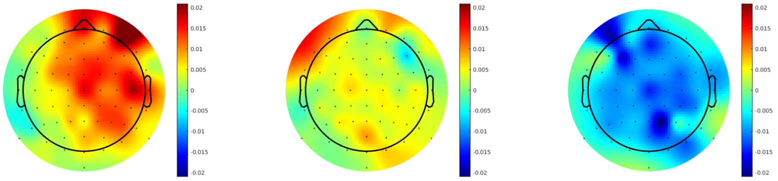
4.4 Structure of the data

For all $L = 24$ trials the EEG data was collected for approximately 120 seconds, for $K = 64$ channels. The data for each channel was stored in a time series array $X_t = \{x_1, x_2, .., x_T\}$ where $t = \{1, ..., T\}$, $T = 6913$. So for each trial Z_i there were 64 channels C: $Z_i = \{C_1, C_2, .., C_L\}$, where $C = \{X_{t1}, X_{t2}, .., X_{tK}\}$. The eye gaze data only has $K = 1$ channel but is otherwise structured in the same way as the EEG data, with the same number of trials.

4.5 Visualisation

EEG

Using brain electrical topography, it is possible to easily visualise what electrodes that are receiving electrical activity. This can be done subject by subject, trial by trial. Figure 6.5 presents a visualisation of subjects 2, 6 and 7, where the mean values of each electrode for the first trial are plotted in three topographic plots. The figure shows three heads seen from above where the nose is pointing forward. Each dot represents an electrode. As seen to the right of each head, the colours represent the values at each electrode compared to each other. These specific mean values will not be further investigated, but this way of visualising values for each electrode is used in this thesis.

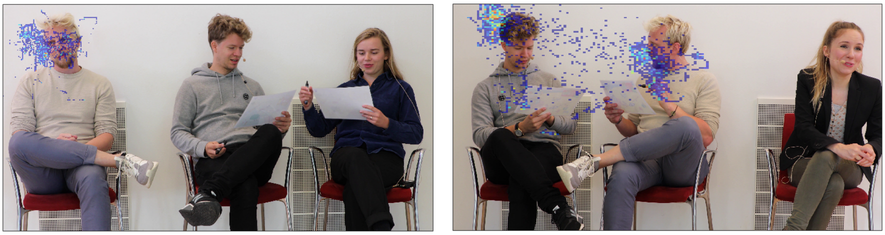


(a) Subject 2 **(b)** Subject 6 **(c)** Subject 7

Figure 4.2 Topographic plots for three subjects' electrodes' mean values for the first trial.

Eye gaze

The eye gaze was visualised using the finished script that was used to extract features in the original data [Bilert, 2020]. Figure 4.3 shows the eye gaze data plotted on top of images corresponding the trials from the main experiments. Here the subject was watching both a dialogue and a monologue at the same time, but was told to focus on either the monologue or the dialogue. Note that the images are just one frame from the videos and since the persons in the videos move their heads, the points do not correspond perfectly to the images.



(a) Monologue

(b) Dialogue

Figure 4.3 Eye gaze measurements for subject 2 plotted on top of one frame from the videos for the main experiment. The eye gaze data is clearly positioned on the attended source, and the movement pattern of the eyes also varies significantly depending on whether the subject is attending to a monologue or a dialogue.

From figure 4.3 it is evident that the eye gaze data is positioned on the attended source. It can also be seen that the movement pattern of the eyes varies significantly depending on whether the subject is attending to a monologue or a dialogue. The plotted data is from the entire trial length, so an investigation of whether a shorter time window would lead to the same clear results could be interesting. The eye gaze data could also be valuable as an extra check whether a model designed for another dataset as an input is working. When using the eye gaze data as an input to machine learning models in this thesis, only the movement of the eyes was used, without the audio or the videos.

5

Method

By letting machine learning models train on EEG and eye gaze data it was investigated if it would be possible to predict if the test subject was attending to a monologue or a dialogue. First an SVM model was tested, then MLP models and finally five different CNN models. All scores were validated using LOOCV. The overall methodology was the same for all models, but more specific implementations differed. The trials used were the ones from the main dataset, where a monologue and a dialogue were presented at the same time to the subject. Each model was trained separately for each subject.

5.1 Feature extraction

To compress the time series data from each channel of the EEG data and the eye gaze data, several features were extracted and used as input to the SVM and MLP. Extracting features is a way of sorting out parts of the data that are not important for classification, and to focus the models on what is most important. As a first step many features were extracted and evaluated. The choice of the features were based on available methods from NumPy [Harris et al., 2020]. The five features that made the best predictions were analysed more in detail. The features were extracted by reducing the entire time series for each channel into one value in different ways. The features that were kept for further analysis were the mean value, the standard deviation, the 25^{th} and 75^{th} percentiles and the mean value of the angle in the frequency domain. Then each combination of these five features were investigated and the combination that gave the highest prediction accuracy was kept and used as input to the models.

Mean value

The mean value was calculated from the time series for each channel as presented in equation (5.1).

$$\overline{X_t} = \frac{1}{T}\sum_{t=1}^{T} x_t \tag{5.1}$$

Standard deviation

The standard deviation was calculated similarly as the mean value for each channel from the time series. The calculation is showed in equation (5.2).

$$std = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(X_t - \overline{X_t})^2} \tag{5.2}$$

Percentiles

The 25^{th} percentile is the value where 25% of the points in the time series have a lower value. Similarly, the 75^{th} percentile is the value where 25% of the examples in the time series have a higher value.

Mean value of angle in the frequency domain

The mean value of the angle in the frequency domain was calculated by applying a fast Fourier transform [Harris et al., 2020], that was defined as presented in equation (5.3) and then taking the mean of the angle of the resulting complex argument.

$$A_k = \sum_{t=0}^{T-1} X_t e^{-2\pi i \frac{tk}{T}} \qquad k = 0, 1, ..., T-1 \tag{5.3}$$

Standardisation of features

When the features had been extracted, they were standardised to have a similar impact on the classification. This was done using scikit-learn's StandardScaler [Pedregosa et al., 2011], which subtracts the mean of the training samples and divides by the standard deviation of the sample as presented in equation (5.4) where f_{ij} is the specific feature sample, f_j a vector of samples for that specific feature.

$$s = \frac{f_{ij} - \overline{f_j}}{\sqrt{(f_j - \overline{f_j})^2}} \tag{5.4}$$

5.2 Input and output to the models

The input to the SVM and MLP models was a feature vector, f_i. For the CNNs, the input was the entire time series array, X_t. The time series array could either be split up into different channels or an entire matrix containing all the channels. The input went into the models, was processed, and the output was either 0 or 1 depending on which class the model predicted the input to belong to. The class vector was $y_i = (0,1)$ depending on whether the subject was told to focus on a monologue or a dialogue.

5.3 Electrode regions

To investigate what parts of the brain that had a big impact on predicting whether the subject was attending to a monologue or a dialogue, the electrodes were divided into six different groups. This also made it possible to see whether fewer electrodes could be used to make a good prediction. These groups contained 4, 5 or 6 electrodes each and were grouped together depending on what region of the brain they belonged to. The six regions were: temporal left and right, frontal left, right, and center, central, parietal and occipital. Exactly what electrodes that belonged to each group is presented in figure 5.1. The models were trained using data from one electrode at a time as input and was then averaged over the electrodes in each group and over subjects.

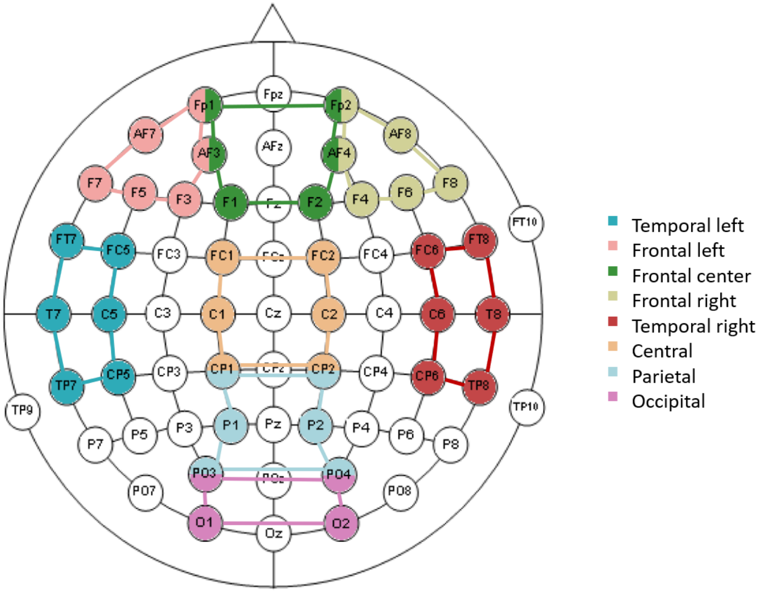


Figure 5.1 Plot of the position of the electrodes belonging to different regions. The regions investigated are: temporal left and right, frontal left, right, and center, central, parietal and occipital.

5.4 Process

All the data from the EEG and eye gaze measurements was initially stored as time series arrays, X_t. The number of channels C depended on if it was EEG or eye gaze data being investigated as an input. $K = 64$ for EEG when all electrodes were used, and $K = 1$ when the electrodes were used individually as well as when using eye gaze data. For the eye gaze data, the yaw angle was used to identify whether the subject was attending to a monologue or a dialogue. Since the yaw angle represents horizontal eye movements, it was expected that a subject, attending to a dialogue would have much more horizontal eye movements than a person attending to a monologue.

The first step was to select what models to use from the different variants of SVM, MLP and CNN. Only the best versions were used for further analysis. For the SVM model the first step was to find good features, and for the MLP and CNN models it was investigated which of the versions described in section 3.4 that reached the highest accuracy. The final step of these selection processes was to change the time window length of the trials. Each 120 seconds trial was split into n windows of equal length (n = 1, 2, ..., 10). Each split led to more trials, each with less data in it.

After this, each electrode was used individually for each subject to predict between monologue and dialogue. The averages of the resulting scores from this were then compared between each subject. These mean values were also presented in topographic plots done in MATLAB using the EEGLAB add-on. The coordinates for the electrode locations were selected using the BESA file for 4-shell dipfit spherical model [Delorme and Makeig, 2004].

The next step was to find the five electrodes that performed best at the classification task for each model. When the best electrodes for each subject had been identified, the overall best electrodes for each model were calculated by summing up the scores from all subjects. It was then investigated what regions of electrodes that performed best as presented in section 5.3. After this, the optimal number of electrodes needed for a successful prediction were analysed. This was done by sorting the electrodes after the mean accuracy of all trials of all subjects. The best electrodes were then chosen from this list and taken away gradually.

The final part was to investigate the time dependency of the five best electrodes of each model as well as the eye gaze data. Just like the when selecting models, this was done by splitting the time window length from 120 seconds into n windows of varying length (n = 1, 2, ..., 10).

During all experiments, the validation of the models was done with LOOCV. Here each sample left out consisted of all the data from one trial at the time. This was done to make sure that the validation was always done on data from another trial.

6

Results

6.1 Selecting models

Support vector machine

The experiments done with the SVM calculated each subject's electrode's ability to predict whether a subject was attending to a monologue or a dialogue. From the time series arrays, X_t, the standard deviation, the 25^{th} and the 75^{th} percentiles, the mean, and the mean of the phase in the frequency domain were calculated. These were combined in all possible ways to choose which features to move forward with. The resulting mean scores for each subject's electrode's prediction accuracy are presented in table 6.1. Here, STD stands for standard deviation and $Freq\ mean$ for the mean of the phase in the frequency domain. The EEG column presents the mean of all subjects' electrodes' accuracy. The Eye column presents the mean of all subjects' accuracy when using eye gaze data as input. X in the table denotes the used features.

The highest mean accuracies for the EEG and eye gaze data were 0.78 and 0.99 respectively and are marked with a red colour in table 6.1. The best EEG result was reached with the combination standard deviation and the 25^{th} and 75^{th} percentiles. The resulting feature vector as input to the model was therefore $f = (std, 25^{th}, 75^{th})$ and this was used during further experiments with the SVM.

The original time window length for each time series array was 120 seconds. Splitting this into shorter time intervals and thus increasing the amount of trials instead resulted in the mean prediction accuracies seen in figure 6.1. Here, each 120 seconds trial was split into n windows of varying length (n = 1, 2, ..., 10). All lines reached their highest value at 120 seconds, in other words with no split of the data. Therefore, no time split was used for the rest of the SVM calculations.

Table 6.1 Mean scores of the SVM model when combining different features. The highest accuracy for EEG was reached using the feature vector $f = (std, 25^{th}, 75^{th})$. X denotes the used features.

STD	Percentile 25	Percentile 75	Mean	Freq mean	EEG	Eye
X					0.69	0.97
	X				0.68	0.97
		X			0.67	0.93
			X		0.54	0.63
				X	0.53	0.54
X	X				0.75	0.85
X		X			0.75	0.96
X			X		0.66	0.91
X				X	0.66	0.78
	X	X			0.74	0.88
	X		X		0.65	0.98
	X			X	0.65	0.85
		X	X		0.65	0.96
		X		X	0.64	0.93
			X	X	0.56	0.58
X	X	X			0.78	0.89
X	X		X		0.71	0.98
X	X			X	0.71	0.79
X		X	X		0.71	0.87
X		X		X	0.70	0.96
X			X	X	0.64	0.97
	X	X	X		0.70	0.97
	X	X		X	0.70	0.95
	X		X	X	0.64	0.95
		X	X	X	0.64	0.87
X	X	X	X		0.75	0.98
X	X	X		X	0.74	0.98
	X	X	X	X	0.69	0.96
X	X		X	X	0.69	0.99
X		X	X	X	0.69	0.97
X	X	X	X	X	0.73	0.97

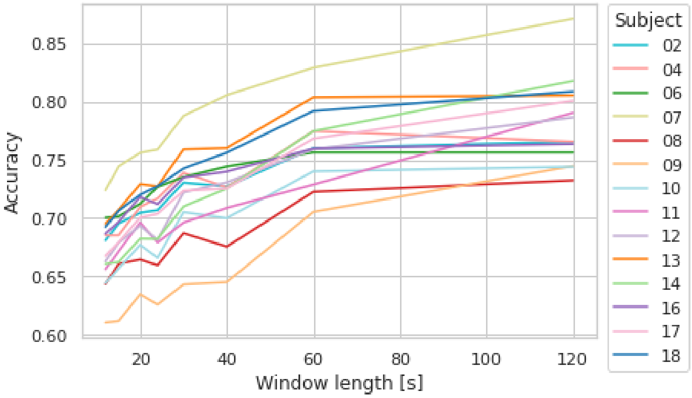


Figure 6.1 The scores of using EEG to predict whether a subject is attending to a monologue or a dialogue using the SVM model with the feature vector $f = (std, 25^{th}, 75^{th})$. Each 120 seconds trial was split into n windows of varying length (n = 1, 2, ..., 10) and evaluated for each time split. Using no time split but keeping the trials at 120 seconds resulted in the highest scores.

Multilayer perceptron

Different kinds of MLPs were tested and the accuracies for the two best ones are presented in table 6.2. Here ft is short for features. The two best MLP models both contain one hidden layer. The first model was created using KerasTuner [O'Malley et al., 2019], where the hyperparameters were tuned for each channel as described in section 3.4. For this model all five features presented in table 6.1 were used as input. The other model was a fixed model based on the most frequent hyperparameters that were found with KerasTuner, using 80 units, ReLU as activation function in the hidden layer and the sigmoid activation function in the output layer. This model was trained both using five features as input and the three features that were best for the SVM model, $f = (std, 25^{th}, 75^{th})$. This model was used for both the EEG data and the eye gaze data.

As seen in table 6.2, the fixed model with three features as input was the best model for the EEG data and is marked with red colour in the table. Therefore, this model was selected for further analysis. For the eye gaze data, the fixed model performed equally good for five and three input features.

Splitting the original time window length for each time series array into shorter time intervals resulted in the mean prediction accuracies seen in figure 6.2. Just like the SVM model, all lines reached their highest value at 120 seconds. No time split was therefore used for the rest of the MLP experiments.

Table 6.2 Resulting mean scores for all subjects when each electrode predicts between a monologue or a dialogue using two different MLPs with 1 hidden layer. One is created with KerasTuner and the other is fixed network with 80 neurons. The fixed network uses ReLu as activation function in the hidden layer and sigmoid as activation function in the output layer. The MLP with KerasTuner had 5 features as input and the fixed MLP had either all 5 features or the feature vector $f = (std, 25^{th}, 75^{th})$. Here ft is short for features.

Subject	Tuner, 5ft	Fixed, 5ft	Fixed, 3ft	Fixed eye, 5ft	Fixed eye, 3ft
2	0.65	0.64	0.74	0.63	0.75
4	0.59	0.64	0.68	-	-
6	0.60	0.58	0.68	0.60	-
7	0.63	0.71	0.83	0.88	0.88
8	0.59	0.56	0.59	0.79	0.67
9	0.57	0.59	0.74	-	-
10	0.55	0.58	0.63	0.76	0.71
11	0.57	0.51	0.66	0.73	0.67
12	0.64	0.59	0.67	-	-
13	0.60	0.61	0.73	0.70	-
14	0.66	0.69	0.74	0.67	0.67
16	0.64	0.58	0.69	0.83	0.88
17	0.56	0.60	0.75	0.79	0.71
18	0.67	0.64	0.74	-	-
Mean	0.61	0.61	0.71	0.74	0.74

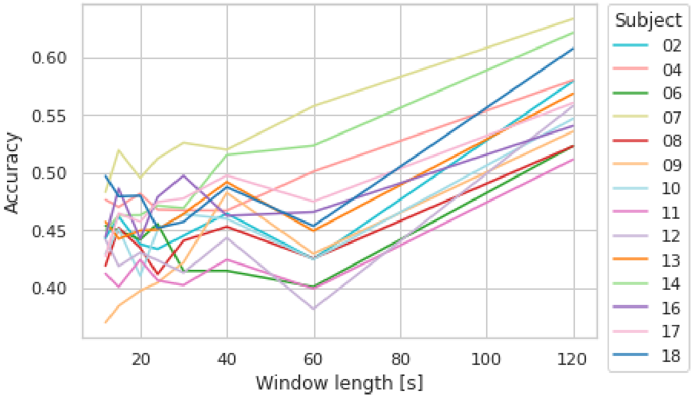


Figure 6.2 The scores of using EEG to predict whether a subject is attending to a monologue or a dialogue using the MLP model. Each trial was split into different number of seconds and evaluated for each time split. Using no time split resulted in the highest scores.

Convolutional neural network

Five different CNN models were implemented and run on each subject's data. The models are described thoroughly in section 3.4. Figure 6.3 shows how the models performed during different time window length splits. Since models 1 and 5 were best overall and peaked at the time around 24 seconds, this time split was used for further experiments with CNN.

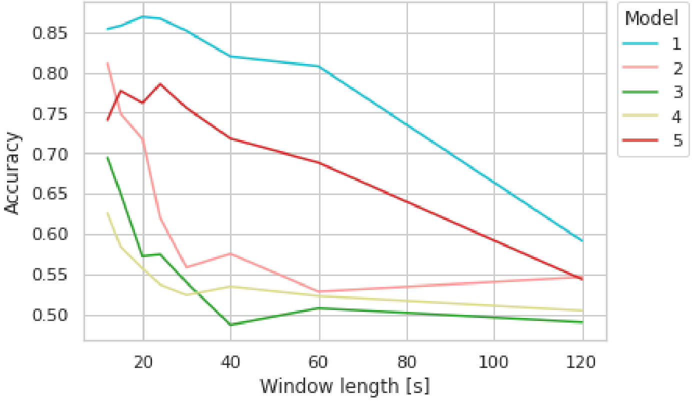


Figure 6.3 The scores of using EEG to predict whether the subject is attending to a monologue or a dialogue using the CNN models. Each trial was split into different number of seconds and evaluated for each time split. Using a time split of around 24 seconds resulted in the highest scores for both of the two best models.

The resulting prediction scores for using each subject's EEG data as an input is seen in table 6.3. The highest mean accuracy for EEG was reached with the first CNN model. This result is marked with a red colour in table 6.3. This network was the only one used in further experiments with CNNs. The scores for this model using each subject's eye data as an input is presented in the last column in table 6.3.

Table 6.3 The table presents the resulting scores when predicting between a monologue or a dialogue from EEG data for the CNN models. The first CNN model performs best, and the last column presents the scores from this model but using eye gaze data as an input.

Subject	**CNN 1**	**CNN 2**	**CNN 3**	**CNN 4**	**CNN 5**	**Eye CNN1**
2	0.93	0.75	0.59	0.64	0.88	0.94
4	0.96	0.63	0.61	0.55	0.83	-
6	0.99	0.74	0.77	0.55	0.84	0.74
7	0.92	0.54	0.63	0.50	0.87	0.89
8	0.96	0.59	0.53	0.51	0.86	0.92
9	0.52	0.52	0.43	0.50	0.54	-
10	0.75	0.60	0.63	0.52	0.66	0.96
11	0.89	0.63	0.55	0.56	0.83	0.92
12	0.96	0.56	0.59	0.52	0.84	-
13	0.86	0.56	0.55	0.51	0.75	0.87
14	0.97	0.51	0.53	0.50	0.83	0.85
16	0.83	0.67	0.61	0.53	0.73	0.93
17	0.69	0.70	0.45	0.53	0.80	0.80
18	0.92	0.68	0.58	0.56	0.76	-
Mean	0.87	0.62	0.57	0.53	0.79	0.88

6.2 Prediction accuracy from each electrode

For each subject, all electrodes were used individually to predict if the subject was attending to a monologue or a dialogue. The SVM model reached highest scores overall for this and the results are presented in the box plots in figure 6.4. The median values are represented with a horizontal line in each coloured box in the plot, the size of the box represents 50% of the data points.

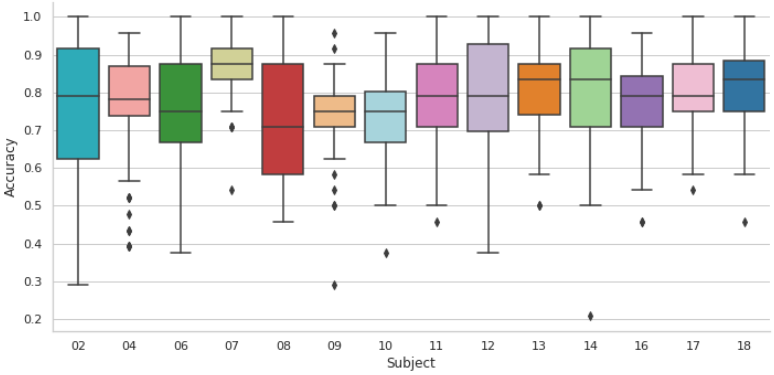
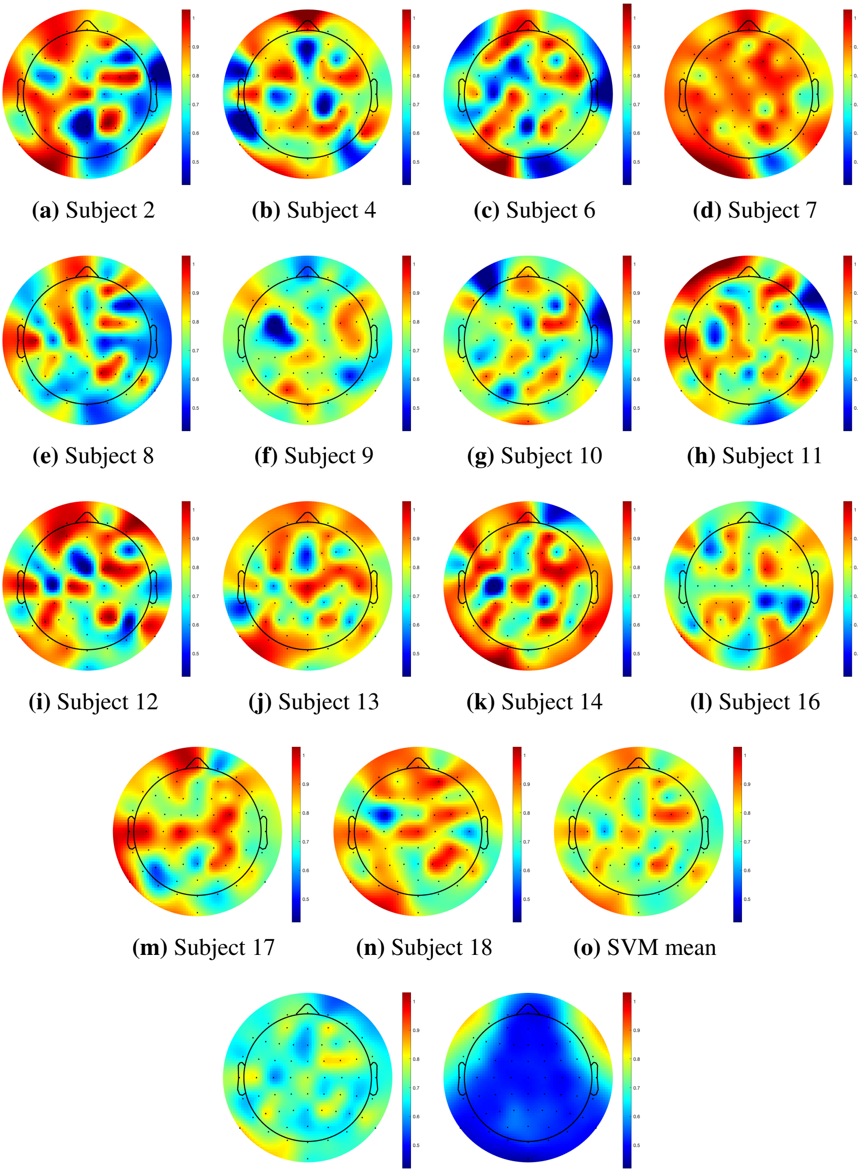


Figure 6.4 Box plot of the accuracy for each subject when predicting whether the subject is attending to a monologue or a dialogue. Each electrode was used separately as input to the SVM model, and the median and standard deviation for each subject is visible in the plot.

Each electrode was compared to each other depending on its placement on the subject's head. This was visualised by plotting topographic plots with the reached prediction accuracies for each subject. The model with the highest scores overall was, as already mentioned, SVM and performances from each subject using this model are presented in figure 6.5. The figure also includes topographic plots of the average scores for all three models. It is only possible to spot a simple pattern from the plot of the average scores from the CNN.



(a) Subject 2 **(b)** Subject 4 **(c)** Subject 6 **(d)** Subject 7

(e) Subject 8 **(f)** Subject 9 **(g)** Subject 10 **(h)** Subject 11

(i) Subject 12 **(j)** Subject 13 **(k)** Subject 14 **(l)** Subject 16

(m) Subject 17 **(n)** Subject 18 **(o)** SVM mean

(p) MLP mean **(q)** CNN mean

Figure 6.5 Topographic plots for each subject's electrodes accuracy of predicting between a monologue or a dialogue using SVM. The last subplots present the mean values of all subjects for the MLP and CNN models. All plots show activity in multiple parts of the brain, only from CNN's mean it is possible to spot a simple pattern.

The best electrodes for predicting between a monologue or a dialogue are presented in tables 6.4, 6.5 and 6.6. Here, *Electrode 1* represents the best electrode and *Electrode 5* the fifth best electrode. The electrode labels are presented together with the reached scores. The last rows of the tables present the electrodes with the highest accuracies after summing up the scores from all subjects. For the SVM model, electrodes FC4, P2, FC2, AF4, and FC6 were best. For the MLP with 1 hidden layer and only 3 features, the best electrodes were FC4, FC2, P5, C5 and FC6. The CNN model's best electrodes were F8, FT8, F7, AF7 and FT7.

Table 6.4 The five best electrodes for each subject when predicting between a monologue or a dialogue for a SVM. The best electrodes overall were FC4, P2, FC2, AF4 and FC6.

Subject	Electrode 1	Electrode 2	Electrode 3	Electrode 4	Electrode 5
2	FC6 : 1.0	FC4 : 1.0	Cz : 1.0	Fp1 : 0.96	AF7 : 0.96
4	FC3 : 0.96	FC1 : 0.96	C5 : 0.96	Fpz : 0.96	FC6 : 0.96
6	AF3 : 1.0	P5 : 1.0	O1 : 1.0	AF4 : 1.0	FC6 : 1.0
7	P9 : 1.0	Fpz : 1.0	F2 : 1.0	P2 : 1.0	AF3 : 0.96
8	C5 : 1.0	AF4 : 1.0	P2 : 1.0	FC1 : 0.96	C1 : 0.96
9	PO3 : 0.96	FC4 : 0.92	F7 : 0.88	Oz : 0.88	CPz : 0.88
10	FC4 : 0.96	FC2 : 0.96	P5 : 0.92	Iz : 0.92	F6 : 0.92
11	T7 : 1.0	AF8 : 1.0	AF4 : 1.0	FC4 : 1.0	P8 : 1.0
12	Fp1 : 1.0	AF3 : 1.0	C1 : 1.0	T7 : 1.0	P5 : 1.0
13	Cz : 1.0	FC3 : 0.96	P5 : 0.96	PO7 : 0.96	FC4 : 0.96
14	F7 : 1.0	FC5 : 1.0	P5 : 1.0	P9 : 1.0	AF4 : 1.0
16	P9 : 0.96	FC2 : 0.96	AF3 : 0.92	FC1 : 0.92	P1 : 0.92
17	C1 : 1.0	C5 : 1.0	T7 : 1.0	P9 : 1.0	FC4 : 1.0
18	AF4 : 1.0	FC4 : 0.96	Cz : 0.96	CP4 : 0.96	P2 : 0.96
All	FC4	P2	FC2	AF4	FC6

Table 6.5 The table presents the five best electrodes of each subject when predicting between a monologue or a dialogue for a MLP with 1 hidden layer and only 3 features. The best electrodes overall are FC4, FC2, P5, C5 and FC6.

Subject	Electrode 1	Electrode 2	Electrode 3	Electrode 4	Electrode 5
2	C5 : 0.96	T7 : 0.96	FC4 : 0.96	Fp1 : 0.92	AF3 : 0.92
4	FC5 : 0.92	CP4 : 0.92	P2 : 0.92	P8 : 0.92	C5 : 0.88
6	P2 : 0.96	P9 : 0.92	AF4 : 0.92	P5 : 0.88	Fpz : 0.88
7	P9 : 1.0	P2 : 1.0	AF3 : 0.96	C5 : 0.96	FC4 : 0.96
8	AF4 : 0.88	P2 : 0.88	AF3 : 0.83	C1 : 0.83	C5 : 0.83
9	F7 : 0.88	FC1 : 0.88	Iz : 0.88	Oz : 0.88	CPz : 0.88
10	P5 : 0.83	P9 : 0.83	FC2 : 0.83	Fp1 : 0.79	CPz : 0.79
11	T7 : 0.96	FC6 : 0.92	F5 : 0.88	FC2 : 0.83	Fp1 : 0.79
12	AF4 : 0.96	P2 : 0.96	AF3 : 0.92	P9 : 0.92	T7 : 0.88
13	P5 : 0.88	P9 : 0.875	AF4 : 0.875	FC2 : 0.875	P2 : 0.875
14	P5 : 1.0	FC2 : 1.0	P9 : 0.96	AF4 : 0.96	FC4 : 0.96
16	P3 : 0.83	P9 : 0.83	F6 : 0.83	F8 : 0.83	PO8 : 0.83
17	C5 : 0.92	Fpz : 0.88	F3 : 0.83	F5 : 0.83	FC5 : 0.83
18	FC4 : 0.96	C5 : 0.92	P5 : 0.92	FC2 : 0.92	T7 : 0.88
All	FC4	FC2	P5	C5	FC6

Table 6.6 The table presents the five best electrodes of each subject when predicting between a monologue or a dialogue for the CNN model. The best electrodes overall are F8, FT8, F7, AF7 and FT7.

Subject	Electrode 1	Electrode 2	Electrode 3	Electrode 4	Electrode 5
2	F8 : 0.90	F7 : 0.84	FT8 : 0.83	AF7 : 0.80	T8 : 0.78
4	FT7 : 0.71	FT8 : 0.70	F7 : 0.70	T8 : 0.69	F8 : 0.67
6	AF7 : 0.87	F7 : 0.83	AF8 : 0.82	F6 : 0.79	FC6 : 0.77
7	F8 : 0.87	AF7 : 0.84	F7 : 0.82	FC5 : 0.82	FT8 : 0.81
8	AF8 : 0.85	F8 : 0.81	F7 : 0.73	FT7 : 0.70	AF7 : 0.68
9	FT7 : 0.65	T7 : 0.62	F7 : 0.58	AF7 : 0.58	FC5 : 0.57
10	AF8 : 0.78	F8 : 0.75	FT7 : 0.73	F7 : 0.71	FT8 : 0.71
11	AF7 : 0.81	F8 : 0.69	FT8 : 0.69	F6 : 0.69	AF8 : 0.68
12	FT8 : 0.86	F8 : 0.84	F7 : 0.83	AF7 : 0.81	T8 : 0.75
13	F7 : 0.77	F8 : 0.76	POz : 0.75	AF7 : 0.70	F5 : 0.70
14	FT8 : 0.78	AF7 : 0.73	FT7 : 0.73	T8 : 0.73	PO3 : 0.71
16	AF7 : 0.74	F7 : 0.74	FT7 : 0.73	F5 : 0.73	FT8 : 0.72
17	FT8 : 0.84	F8 : 0.75	FC6 : 0.74	AF8 : 0.73	FT7 : 0.70
18	AF7 : 0.78	F7 : 0.75	FT7 : 0.75	FT8 : 0.74	F8 : 0.73
All	F8	FT8	F7	AF7	FT7

Electrode regions

The performance of the electrodes based on what region of the brain they belonged to was investigated and the results are presented in figures 6.6 and 6.7, 6.8. The box plots seen in these figures all have a median value marked with a horizontal line in each box and the size of the box represents 50% of the data points. All three models had the same four regions with the highest means. These regions were temporal left, frontal left, frontal center and temporal right.

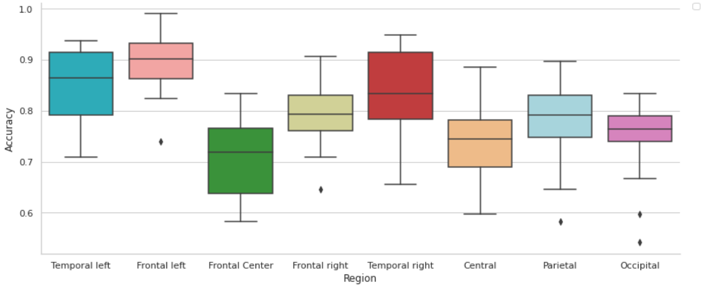


Figure 6.6 The performance for electrodes in different regions for SVM, over all subjects. The four regions with the highest medians are temporal left, frontal left, frontal center and temporal right.

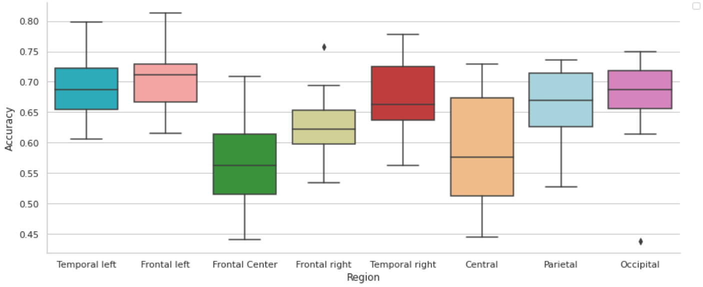


Figure 6.7 The performance for electrodes in different regions for MLP, over all subjects. The four regions with the highest medians are temporal left, frontal left, frontal center and temporal right.

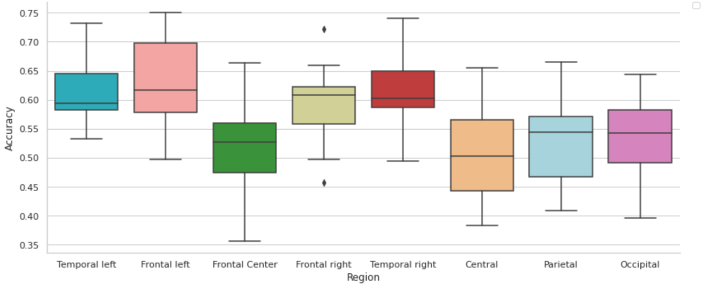


Figure 6.8 The performance for electrodes in different regions for CNN, over all subjects. The four regions with the highest medians are temporal left, frontal left, frontal center and temporal right.

6.3 Optimal number of electrodes

It was investigated how few electrodes could be used without the accuracy being too affected. Only the best electrodes for each model, presented in tables 6.4, 6.5 and 6.6, were kept. The model that had the highest accuracy was the SVM model and its result is presented in figure 6.9, but all models had a similar pattern. For the SVM, the accuracy stayed somewhat consistent until 12 electrodes remained. The accuracy improved for almost all subjects when only 4 electrodes were used.

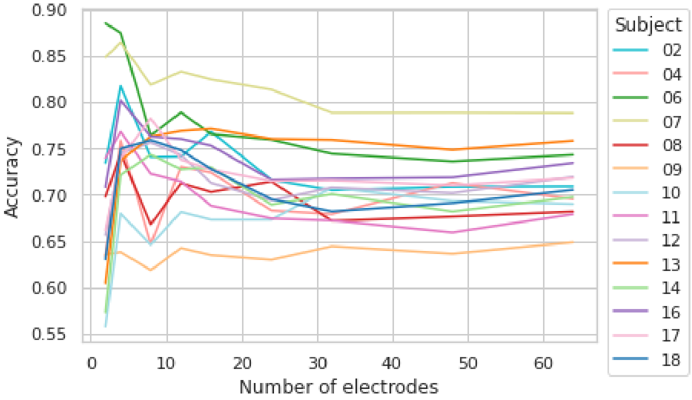
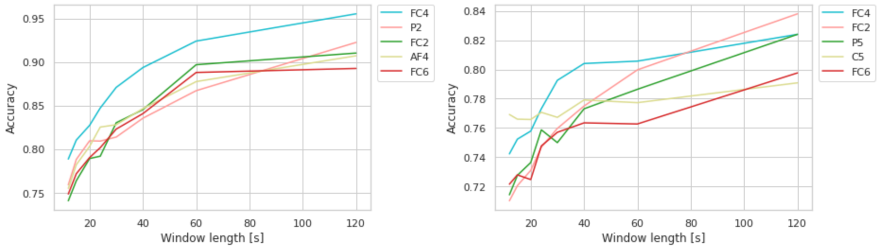


Figure 6.9 The performance for each subject, when decreasing number of electrodes, using SVM. The accuracy stayed somewhat consistent until 12 electrodes remained and improved for almost all subjects when only 4 electrodes were used.

6.4 Time dependency for best electrodes

Using the five best electrodes from each model, presented in tables 6.4, 6.5 and 6.6, the time dependency of their prediction scores were investigated. Changing the time window lengths resulted in accuracies following the same pattern as when using all electrodes, as seen in figures 6.1, 6.2 and 6.3. The two models performing best overall are presented in figure 6.10, where the mean scores of all subjects are plotted. It can be seen here that there is a decrease of accuracy for the five best electrodes when the time window lengths get shorter.



(a) SVM

(b) MLP

Figure 6.10 The mean scores of all subjects using the five best electrodes of SVM and MLP different amount of seconds. There is a decrease of accuracy when the time window lengths get shorter.

6.5 Time dependency for eye data

The final investigation was of how the performance of the eye data as an input was affected by changing the time window lengths. The same feature vector as above was used as input to the SVM and the MLP: $f = (std, 25^{th}, 75^{th})$. The SVM model reached scores that were highest overall and its result is presented in figure 6.11, but all models had similar patterns where the accuracy did not decrease remarkably for shorter time splits. For the SVM, all time windows of 10 seconds and above did not lead to a higher accuracy.

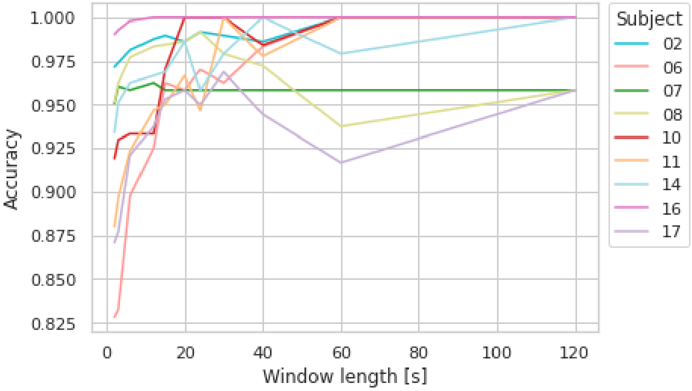


Figure 6.11 The scores when using eye data to predict a monologue and a dialogue for the SVM using different amount of seconds. For all time windows of 10 seconds and above, the longer time splits did not contribute to a higher accuracy.

6.6 Summary

The implemented CNN performed best overall for the EEG data and reached an average prediction score of 87% for all subjects when using inputs from all electrodes at the same time. The best performance of the CNN occurred when using a time window length of around 24 seconds. When using all electrodes individually as input, the CNN performed worse than the SVM and MLP. The best model was then instead the SVM that reached an average of 78%. Both the SVM and the MLP worked best without any time split. For the eye gaze data the SVM performed best, reaching an average accuracy of 99%.

For all models, the electrodes at the temporal lobe as well as the sides of the front of the frontal lobe performed best. Decreasing the number of electrodes led to varied performances of the models. The SVM model had a somewhat consistent accuracy until 12 electrodes remained, and an increased accuracy for 4 electrodes.

Using the five best electrodes from each model, the results varied regarding how short the trials could be before the accuracy decreased. The models behaved in a similar way as when investigating time dependency for all electrodes. When using eye gaze data as input the accuracy was not affected remarkably by decreasing the time window length.

7

Discussion

7.1 Selecting models

Support vector machine

The combination of features that resulted in the highest prediction performance was one that described how the data was spread. The fact that the features describing mean values were not selected indicates that the spread of the data was more distinctive between the two classes than the mean values. The classification between a monologue and a dialogue was in other words not better when using an average of the EEG data.

The features used to reach the highest accuracy when using the EEG data as an input to the SVM model did not match the features used for reaching the highest accuracy when using eye gaze as an input. The features that resulted in the highest accuracy for the EEG data resulted in an average accuracy of 89% when using eye gaze data. The best average accuracy for the eye gaze data was however 99% when using four features as presented in table 6.1. The classification task should be fairly easy from using the eye gaze data as an input since the subjects' eyes are mainly resting on the person they are attending to. Any model using the eye gaze data as an input should in other words be able to reach a high accuracy on this binary classification task. An accuracy of 89% was still quite high but because of this being so much lower than when using many other feature combinations, it is clear that the EEG and eye gaze data are quite different and that they require very different features in order to obtain a high accuracy.

Why 120 seconds was the best time split does not have an obvious explanation, since shorter time splits also mean more trials, Z, to train on, which could have been an advantage to the model. It seems however like longer time splits are advantageous for the training.

Multilayer perceptron

The fixed model was found to have the same mean accuracy across all subjects as the one that used KerasTuner to adapt the model to the data for each subject. This indicates that the fixed model could be used instead of tuning the model, and that the chosen hyperparameters did not have a great impact on the performance. The model was only tuned for the EEG data, and never for the eye gaze data, which means that the model was not specifically adapted to handle the eye gaze data. It was however seen that the model did get a higher accuracy for the eye gaze data than for the EEG signals. This was probably because the eye gaze data was much simpler than the EEG data, and that it was easier to distinguish between whether the subject was attending a dialogue or monologue based on the eye gaze data. The difference was larger when 5 features were used, and that was the same trend as for the SVM, so this result supports the fact that the result is very dependent on which features that are being used.

The MLP accuracy for the eye gaze data was not very high, especially when comparing to other models, so it was clear that something else was disturbing the performance. It might have been that the model was not good enough for this kind of data, lack of training data or that it was too adapted to the EEG data. The combination of features that gave the highest result on the eye gaze data for the SVM was not tried for the MLP model, but it would have been interesting to see if the model would have performed better for these features.

Just like the SVM model, the highest accuracy for the MLP model was reached for 120 seconds. As soon as the time window length was decreased to 60 seconds or lower, the scores were lower than 50%. Since there were only two classes to predict between, a score beneath 50% means a result worse than chance. In other words, this model is useless for other time window lengths than 120 seconds.

Convolutional neural network

All CNN models could use all 64 electrodes at the same time as an input. Splitting the data from 120 seconds for each trial into smaller sets had different impacts for different models but they all performed worst at the full time window length 120 seconds, as seen in figure 6.3. For models 2, 3 and 4, the highest accuracies were reached when the data was split the most. Since models 1 and 5 performed best overall it was decided to use the time split where they performed the best, 24 seconds. The first CNN model performed best of the CNN models, with a mean accuracy of 87%. All models were implemented to match an input signal sampled at 128 Hz, although the architectures were designed to match another EEG dataset. According to the source where code was taken from, the first CNN model was the best for their data as well. Since this was the case here too, only this model was used in further analysis of the CNN models.

The rest of the experiments were done by using one electrode at the time as an input to the CNN and then calculating the mean of the accuracies. Since the CNN model was designed for all electrodes together as an input instead of only one this means that its resulting accuracy was expected to decrease. Testing the performance when using one input channel made it possible to both investigate its generalisation but also easier to compare with the performance of the SVM and the MLP.

7.2 Prediction accuracy from each electrode

When using one electrode at the time as an input to the models, the SVM model reached highest accuracies in average for all subjects. The mean values were about 80% for many subjects and many electrodes got as much as 100%. The performance of the CNN model decreased significantly when using one electrode at a time as input to the models. This made, as already stated, sense because they were implemented to handle all electrodes at the same time and therefore had parameters in the architecture that matched this number of channels instead of only one.

Studying the topographic plots from the models' mean values makes it clear that it was a big difference between the SVM and MLP models that used features as input and the CNN model that used the time series as input. The models where features were used had very random topographic patterns, that also varied a lot depending on what features that were chosen. Topographic plots for these models when using other features are included in Appendix A. When no features were used, as seen in the CNN model, the topographic plot had a very generalised pattern. Here, the highest reached accuracies were from the electrodes in the front left and right of the head. It is however difficult to draw a conclusion about whether this pattern would look the same if other models were to be used with all electrodes as input. The pattern could also be a result of the model's characteristics.

Electrode regions

All four models resulted in the same pattern regarding how the electrodes in the different regions performed at the binary classification task. The frontal left was the region where the average score of all subjects always was the highest. The top four regions regarding the average scores were the temporal left, the frontal left, the frontal center and the temporal right. In other words, the regions at the side of the head, mostly at the front were the ones where the electrodes performed best. The result that these four regions were the ones performing best is reasonable, especially since the temporal lobe is in charge of hearing. The frontal lobe is in charge of the cognitive functions as well as voluntary movements. This brain part should therefore be active when focusing on a monologue or a dialogue in a noisy situation.

The regions with the lowest prediction scores were frontal center, central, parietal and occipital. These are placed along the center of the head. The parietal lobe handles touch and temperature, and it therefore makes sense that this region did not perform the best in the task. What was remarkable was that the occipital lobe did not perform better. This is the lobe in charge of vision and should be active during the experiments. A reason for the result could be that it was always as active independent of whether the subject was attending to a monologue or a dialogue. This would mean no ability for these electrodes' data to predict well what kind of conversation the subject was attending to.

7.3 Optimal number of electrodes

The accuracy for the SVM model stayed somewhat consistent until 12 electrodes remained. The accuracy then improved for almost all subjects when 4 electrodes were used compared to when more electrodes were used. Therefore, it seems like it could be a good idea to reduce the number of electrodes to reduce computational cost if using an SVM model. For less than 4 electrodes the accuracy decreased for almost all subjects, so to use less than 4 electrodes would most likely not be beneficial if not specifically adapted to a specific subject. This pattern was not as clear for the other models, so the same conclusion could not be drawn for them, but the general pattern was the same for all models.

7.4 Time dependency for best electrodes

There was a significant decrease of accuracy for the five best electrodes when the time window length got shorter for the SVM model. The accuracy also decreased more for each time split. Therefore, a specific time limit for when the accuracy was decreasing too much could not be set. The time dependency for the MLP shows a clear decrease in accuracy for shorter time splits, and especially after 40 seconds there was a more radical decrease. Therefore, it could be an idea to investigate further if the time splits could be set to 40 seconds for the 5 best electrodes instead. The accuracy down to 12 seconds is still significantly higher than chance, so the model could still perform some classification, but it would not be as accurate as for the longer time windows. The results from the CNN model were much lower overall and therefore not of great importance to investigate further.

7.5 Time dependency for eye data

The result of the investigation of time dependency for the eye data using the SVM model made it clear that for all time windows of 10 seconds and above the longer time splits did not contribute to a higher accuracy. The accuracy for the different time splits for CNN, as presented in Appendix A, seemed to vary less when the time splits were shorter, which indicated that the model might have been more robust for shorter time windows. This result agrees with the previous result for the CNN models from the EEG data, that it performed better for smaller time splits. As discussed above this is probably because that model was created for shorter time splits. In conclusion, it does not seem to decrease the accuracy to reduce the time splits to 10 seconds, it might in some cases even be advantageous for all models.

8

Conclusion

This thesis used data from experiments where a noisy environment was simulated. The test subjects were exposed to a monologue and a dialogue at the same time but were told to only focus on one of them. Using EEG and eye gaze data collected from these experiments as an input, different machine learning models were implemented to solve a binary classification task to predict whether the subject was attending to a monologue or a dialogue. The investigated models were SVM, MLP and CNNs. The input to the models was either time series arrays from EEG signals or eye gaze data. For the SVM and the MLP, more compact representations of the time series arrays were used as inputs. The project has led to the following conclusions:

1. If 64 electrodes are used together, a CNN model makes the best overall prediction. The average accuracy for all subjects is then 87%. When using one electrode at the time instead, the SVM model performs best with an average accuracy of 78%. This is done when using of all subjects' electrodes together with features that represented the time series arrays.

2. It varies how short the trials can be before the accuracy decreases for each model when EEG data is used. The SVM and the MLP performs best for longer trials while the CNN performs best for shorter but an increased amount of trials. For the eye gaze data, the accuracy is not affected remarkably by decreasing the length of trials.

3. For the SVM model, the best performing model, the accuracy is somewhat consistent until 12 electrodes remains. The electrodes should be located at the temporal lobe as well as the sides of the front of the frontal lobe.

9

Continuation

A valid prediction of what source a subject is attending to is of importance for developing intelligent hearing aids. To simplify an implementation of EEG sensors in these hearing aids, the amount of electrodes as well as the time sequence needed for a valid prediction needs to be reduced. Continued studies regarding decreasing the number of electrodes are therefore of interest. This could be for example to investigate how the models perform if only one electrode is used, and to adapt the models specifically to that one electrode.

There are numerous things left to investigate when it comes to the choice of machine learning models. In this project, only three sorts of models were analysed. The CNN model was not modified to fit to the dataset but still managed to perform well when using all electrodes at the same time as an input. If the parameters would be tuned for it to work well with one electrode at the time instead, the model has a chance of outperforming the other models. Working with completely different models than the three selected in this project is also of interest. There are many types of machine learning models and there is a possibility that other ones will perform better at this binary classification task.

The dataset also contained audio and video files of the experiment. This was not used due to lack of time. Since listening also involves lip reading, it would be interesting to combine the studies done in this project with the audio and video files. One step would be to create a speech envelope from the predictions, another to find patterns in the EEG and eye data with what is seen on the lips. There is evidently more that can be done with the dataset that will hopefully help shed light on the cocktail party problem.

Bibliography

Abo-Zahhad, M., S. Ahmed, and S. N. Seha (2015). “A New EEG Acquisition Protocol for Biometric Identification Using Eye Blinking Signals”. *International Journal of Intelligent Systems and Applications (IJISA)* **7**:6, p. 49. DOI: `10.5815/ijisa.2015.06.05`.

Accessible, ". on, A. H. H. C. for Adults; Board on Health Sciences Policy; Health, E. Medicine Division; National Academies of Sciences, D. Medicine" Blazer, S. Domnitz, and C. Liverman (2016). *Hearing Health Care for Adults: Priorities for Improving Access and Affordability*. Washington (DC): National Academies Press (US). URL: `https://www.ncbi.nlm.nih.gov/sites/books/NBK385313/`.

Albawi, S., T. A. Mohammed, and S. Al-Zawi (2017). “Understanding of a convolutional neural network”, pp. 1–6. DOI: `10.1109/ICEngTechnol.2017.8308186`.

Alickovic, E., J. Kevric, and A. Subasi (2018). “Performance evaluation of empirical mode decomposition, discrete wavelet transform, and wavelet packed decomposition for automated epileptic seizure detection and prediction”. *Biomedical signal processing and control* **39**, pp. 94–102.

Andersen, A. H. e. a. (2021). “Creating clarity in noisy environments by using deep learning in hearing aids”. *Seminars in hearing* **42**:3, pp. 260–281. DOI: `10.1055/s-0041-1735134`.

Army Research Laboratory (2022). *A collection of convolutional neural network (cnn) models for eeg signal processing and classification*. `https://github.com/vlawhern/arl-eegmodels`.

Bednar, A. and E. C. Lalor (2020). “Where is the cocktail party? decoding locations of attended and unattended moving sound sources using eeg”. *NeuroImage* **205**, p. 116283. ISSN: 1053-8119. DOI: `https://doi.org/10.1016/j.neuroimage.2019.116283`.

Bernard, F., J.-M. Lemée, A. Ter Minassian, and P. Menei (2018). "Right Hemisphere Cognitive Functions: From Clinical and Anatomic Bases to Brain Mapping During Awake Craniotomy Part I: Clinical and Functional Anatomy". DOI: 10.1016/j.wneu.2018.05.024.

Best, V., T. R. Jennings, and G. Kidd (2020). "An effect of eye position in cocktail party listening". *Proceedings of Meetings on Acoustics* **42**:1, p. 050001. DOI: 10.1121/2.0001344. eprint: https://asa.scitation.org/doi/pdf/10.1121/2.0001344.

Bhagat, P. M. (2021). "Artificial Intelligence in Healthcare". *International Journal of Scientific Research Engineering Trends* **7**:2, p. 796. URL: https://ijsret.com/wp-content/uploads/2021/03/IJSRET_V7_issue2_236.pdf.

Bharath, K. (7, 2020). *Understanding relu: the most popular activation function in 5 minutes!* URL: https://towardsdatascience.com/understanding-relu-the-most-popular-activation-function-in-5-minutes-459e3a2124f (visited on 2022-05-18).

Bilert, S. P. (2020). "Decoding Attention in Real-World Listening".

BioSemi (2022). *ActiveTwo*. URL: https://www.biosemi.com/products.htm (visited on 2022).

Cherry, C. (1935). "Some experiments on the recognition of speech, with one and with two ears". *Journal of the Acoustical Society of America* **25**, pp. 975–979. DOI: 10.1121/1.1907229.

Chollet, F. et al. (2015). *Keras*. URL: https://github.com/fchollet/keras.

Chong, F. Y. and L. M. Jenstad (2018). "A critical review of hearing-aid single-microphone noise-reduction studies in adults and children". *Disability and Rehabilitation: Assistive Technology* **13**:6. PMID: 29072542, pp. 600–608. DOI: 10.1080/17483107.2017.1392619.

Constant, I. and N. Sabourdin (2012). "The EEG signal: a window on the cortical brain activity". *Pediatric Anesthesia* **22**:6, p. 539. DOI: https://doi.org/10.1111/j.1460-9592.2012.03883.x.

Creel, D. J. (2019). "The electrooculogram". *Handbook of clinical neurology* **160**, pp. 495–499. DOI: 10.1016/B978-0-444-64032-1.00033-3.

Das, N., A. Bertrand, and T. Francart (2018). "Eeg-based auditory attention detection: boundary conditions for background noise and speaker positions". *Journal of neural engineering* **15**:6. DOI: https://doi.org/10.1088/1741-2552/aae0a6.

Delorme, A. and S. Makeig (2004). "Electroencephalography (eeg)-based brain-computer interfaces". *Journal of Neuroscience Methods* **134**, pp. 9–21. URL: https://sccn.ucsd.edu/eeglab/download/eeglab_jnm03.pdf.

Ding, N. and J. Z. Simon (2012). “Emergence of neural encoding of auditory objects while listening to competing speakers”. *Proceedings of the National Academy of Sciences* **109**:29, pp. 11854–11859.

Dumane, G. (2020). *Introduction to convolutional neural network (cnn) using tensorflow*. URL: https://towardsdatascience.com/introduction-to-convolutional-neural-network-cnn-de73f69c5b83.

Gandhi, R. (2018). *Support vector machine - introduction to machine learning algorithms*. URL: https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47.

Groner, R. and M. Groner (1989). “Attention and eye movement control: an overview”. *European archives of psychiatry and neurological sciences* **239**:1, pp. 9–16. DOI: 10.1007/BF01739737.

Harris, C. R., K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. Fernández del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant (2020). “Array programming with NumPy”. *Nature* **585**:7825, pp. 357–362. DOI: 10.1038/s41586-020-2649-2.

Huber, J. (2020). *Batch normalization in 3 levels of understanding*. URL: https://towardsdatascience.com/batch-normalization-in-3-levels-of-understanding-14c2da90a338.

Humanities lab, Joint faculties of humanities and theology at Lund University (2022). *EEG*. URL: https://www.humlab.lu.se/sv/utrustning/eeg/ (visited on 2022-02-10).

Keras (2022a). *Conv2d layer*. URL: https://keras.io/api/layers/convolution_layers/convolution2d/ (visited on 2022-05-23).

Keras (2022b). *Earlystopping*. URL: https://keras.io/api/callbacks/early_stopping/ (visited on 2022-05-30).

Keras (2022c). *Layer activation functions*. URL: https://keras.io/api/layers/activations/ (visited on 2022-05-23).

Keras (2022d). *Permute layer*. URL: https://keras.io/api/layers/reshaping_layers/permute/ (visited on 2022-05-23).

Keras (2022e). *Spatialdropout2d layer*. URL: https://keras.io/api/layers/regularization_layers/spatial_dropout2d/ (visited on 2022-05-23).

Khazi, M., A. Kumar, and J. VidyaM (2012). “Analysis of EEG Using 10:20 Electrode System”. *International Journal of Innovative Research in Science, Engineering and Technology* **1**:2, p. 185. URL: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.475.4944&rep=rep1&type=pdf.

Lawhern, V. J., A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance (2018). “Eegnet: a compact convolutional neural network for eeg-based brain–computer interfaces”. *Journal of Neural Engineering* **15**:5, p. 056013. URL: http://stacks.iop.org/1741-2552/15/i=5/a=056013.

Lotte, F., L. Bougrain, and M. Clerc (2015). “Electroencephalography (eeg)-based brain-computer interfaces”. *Wiley Encyclopedia of Electrical and Electronics Engineering*. DOI: 10.1002/047134608X.W8278.

Lu, H., M. F. McKinney, T. Zhang, and A. J. Oxenham (2021). “Investigating age, hearing loss, and background noise effects on speaker-targeted head and eye movements in three-way conversations”. *The Journal of the Acoustical Society of America* **149**:3, pp. 1889–1900.

Lunner, T., E. Alickovic, C. Graversen, E. H. N. Ng, D. Wendt, and G. Keidser (2020). “Three new outcome measures that tap into cognitive processes required for real-life communication”. *Ear and hearing* **41**:Suppl 1, 39S.

Mesgarani, N. and E. F. Chang (2012). “Selective cortical representation of attended speaker in multi-talker speech perception”. *Nature* **485**:7397, pp. 233–236.

Mirkovic, B., S. Debener, M. Jaeger, and M. De Vos (2015). “Decoding the attended speech stream with multi-channel EEG: implications for online, daily-life applications”. *Journal of Neural Engineering* **12**:4, p. 046007. DOI: 10.1088/1741-2560/12/4/046007. URL: https://doi.org/10.1088/1741-2560/12/4/046007.

Montoya-Martínez, J., J. Vanthornhout, A. Bertrand, and T. Francart (2021). “Effect of number and placement of eeg electrodes on measurement of neural tracking of speech”. *PLOS ONE* **16**:2, pp. 1–18. DOI: 10.1371/journal.pone.0246769.

Nationalencyklopedin (2022a). *Cerebral dominans*. URL: https://www.ne.se/uppslagsverk/encyklopedi/l%C3%A5ng/cerebral-dominans (visited on 2022-06-02).

Nationalencyklopedin (2022b). *Hjärna*. URL: https://www.ne.se/uppslagsverk/encyklopedi/l%C3%A5ng/hj%C3%A4rna (visited on 2022-02-10).

Nationalencyklopedin (2022c). *Hjärna*. URL: https://www.ne.se/uppslagsverk/bild/teckning/hj%C3%A4rna-(hej-2) (visited on 2022-03-30).

O’Malley, T., E. Bursztein, J. Long, F. Chollet, H. Jin, L. Invernizzi, et al. (2019). *Kerastuner*. https://github.com/keras-team/keras-tuner.

O’Shea, K. and R. Nash (2015). “An introduction to convolutional neural networks”. *CoRR* **abs/1511.08458**. eprint: 1511.08458. URL: http://arxiv.org/abs/1511.08458.

O'Sullivan, J., A. Power, N. Mesgarani, S. Rajaram, J. Foxe, B. Shinn-Cunningham, M. Slaney, S. Shamma, and E. Lalor (2015). "Attentional selection in a cocktail party environment can be decoded from single-trial eeg". *Cerebral cortex (New York, N.Y. : 1991)* **25**. DOI: 10.1093/cercor/bht355.

Palendeng, M. E. (2011). "Removing Noise From Electroencephalogram Signals For BIS Based Depth of Anaesthesia Monitors". *UNIVERSITY OF SOUTHERN QUEENSLAND*, p. ii. URL: https://eprints.usq.edu.au/23505/2/Palendeng_2011_whole.pdf.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay (2011). "Scikit-learn: machine learning in Python". *Journal of Machine Learning Research* **12**, pp. 2825–2830.

Picou, E. and T. Ricketts (2019). "An evaluation of hearing aid beamforming microphone arrays in a noisy laboratory setting." *Journal of the American Academy of Audiology* **30**:2, pp. 131–144. DOI: 10.3766/jaaa.17090.Epub2018Jan2.PMID:30461406.

Rasheed, K., A. Qayyum, J. Qadir, S. Sivathamboo, P. Kwan, L. Kuhlmann, T. O'Brien, and A. Razi (2021). "Machine learning for predicting epileptic seizures using eeg signals: a review". *IEEE Reviews in Biomedical Engineering* **14**, pp. 139–155. DOI: 10.1109/RBME.2020.3008792.

Refaeilzadeh, P., L. Tang, and H. Liu (2016). "Cross-Validation". DOI: https://doi.org/10.1007/978-1-4899-7993-3_565-2.

Richhariya, B. and M. Tanveer (2018). "Eeg signal classification using universum support vector machine". *Expert Systems with Applications* **106**, pp. 169–182. ISSN: 0957-4174. DOI: https://doi.org/10.1016/j.eswa.2018.03.053.

Russell, S. and P. Norvig (2016). *Artificial Intelligence: A Modern Approach*. Pearson Education Limited.

Saha, S. (2018). *A comprehensive guide to convolutional neural networks — the eli5 way*. URL: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53.

Satheesh Kumar, J. and P. Bhuvaneswari (2012). "Analysis of electroencephalography (eeg) signals and its categorization–a study". *Procedia Engineering* **38**, pp. 2525–2536. DOI: https://doi.org/10.1016/j.proeng.2012.06.298.

Schirrmeister, R. T., J. T. Springenberg, L. D. J. Fiederer, M. Glasstetter, K. Eggensperger, M. Tangermann, F. Hutter, W. Burgard, and T. Ball (2017). "Deep learning with convolutional neural networks for eeg decoding and visualization". *Human Brain Mapping* **38**:11, pp. 5391–5420. DOI: 10.1002/hbm.23730.

Sharma, S., S. Sharma, and A. Athaiya (2020). "Activation functions in neural networks". URL: https://www.ijeast.com/papers/310-316,Tesma412,IJEAST.pdf.

Shiell, M. M., T. Cabella, G. Keidser, D. C. Niehorster, M. Nyström, M. Skoglund, S. With, J. Zaar, and S. Rotger-Griful (2021). "Eye-movement patterns of hearing-impaired listeners measure comprehension of a multitalker conversation". *The Journal of the Acoustical Society of America* **149**:4, A77–A77.

Subasi, A. and M. I. Gürsoy (2010). "EEG signal classification using PCA, ICA, LDA and support vector machines". *Expert Systems with Applications* **37**:12, pp. 8659–8666. DOI: https://doi.org/10.1016/j.eswa.2010.06.065.

Taillez, T. de, B. Kollmeier, and B. T. Meyer (2020). "Machine learning for decoding listeners' attention from electroencephalography evoked by continuous speech". *European Journal of Neuroscience* **51**:5, pp. 1234–1241. DOI: https://doi.org/10.1111/ejn.13790. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/ejn.13790.

Tam, J. (15, 2019). *Choosing the right eye tracker*. URL: https://www.gazept.com/blog/visual-tracking/choosing-the-right-eye-tracker/ (visited on 2022-05-18).

Tobii (2022a). *How does an eye tracker work?* URL: https://www.tobiipro.com/blog/what-is-eye-tracking/.

Tobii (2022b). *Tobii Pro Glasses 2 - Discontinued*. URL: https://www.tobiipro.com/product-listing/tobii-pro-glasses-2/.

Vicon (2022). *Vero*. URL: https://www.vicon.com/hardware/cameras/vero/ (visited on 2022).

Walczak, S. and N. Cerpa (2003). *Artificial Neural Networks*. Ed. by R. A. Meyers. Third Edition. Academic Press, New York, pp. 631–645. ISBN: 978-0-12-227410-7. DOI: https://doi.org/10.1016/B0-12-227410-5/00837-1.

Wang, C.-F. (2022). *A basic introduction to separable convolutions*. URL: https://towardsdatascience.com/a-basic-introduction-to-separable-convolutions-b99ec3102728 (visited on 2022-06-06).

Waytowich, N., V. J. Lawhern, J. O. Garcia, J. Cummings, J. Faller, P. Sajda, and J. M. Vettel (2018). "Compact convolutional neural networks for classification of asynchronous steady-state visual evoked potentials". *Journal of Neural Engineering* **15**:6, p. 066031. URL: http://stacks.iop.org/1741-2552/15/i=6/a=066031.

Weinberger, K. (2018). *Machine learning lecture 14 "(linear) support vector machines"*. URL: https://www.cs.cornell.edu/courses/cs4780/2018fa/lectures/lecturenote09.html.

Wilimitis, D. (2018). *The kernel trick in support vector classification*. URL: https://towardsdatascience.com/the-kernel-trick-c98cdbcaeb3f.

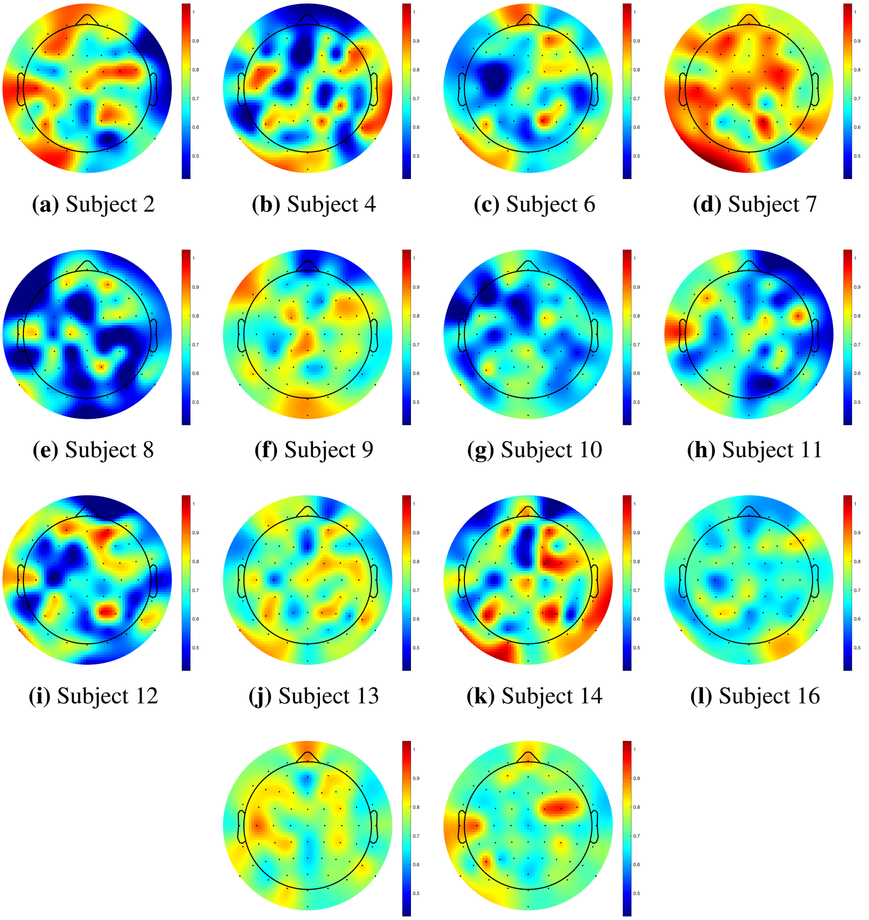
A

Appendix

A.1 Prediction accuracy from each electrode

Table A.1 Resulting mean scores for all subjects when each electrode was used separately to predict between a monologue or a dialogue for the SVM model. The last column presents the results when eye gaze data is used as an input for the subjects containing enough eye gaze data.

Subject	EEG	Eye gaze
2	0.77	1.0
4	0.77	-
6	0.76	1.0
7	0.87	0.96
8	0.73	0.96
9	0.74	-
10	0.74	1.0
11	0.79	1.0
12	0.79	-
13	0.81	-
14	0.82	1.0
16	0.76	1.0
17	0.80	1.0
18	0.81	-
Mean	0.78	0.99



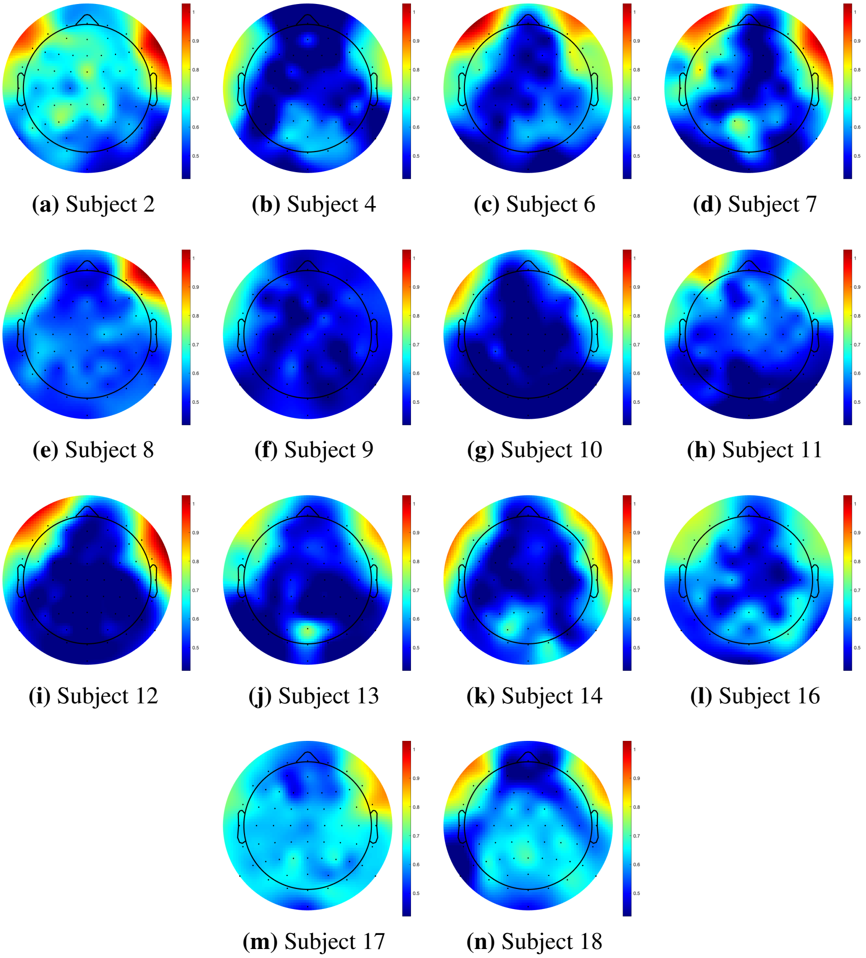
(a) Subject 2 **(b)** Subject 4 **(c)** Subject 6 **(d)** Subject 7

(e) Subject 8 **(f)** Subject 9 **(g)** Subject 10 **(h)** Subject 11

(i) Subject 12 **(j)** Subject 13 **(k)** Subject 14 **(l)** Subject 16

(m) Subject 17 **(n)** Subject 18

Figure A.1 Topographic plots for each subject's electrodes accuracy of predicting between a monologue or a dialogue using MLP, using the three input features percentile 25, percentile 75 and standard deviation.



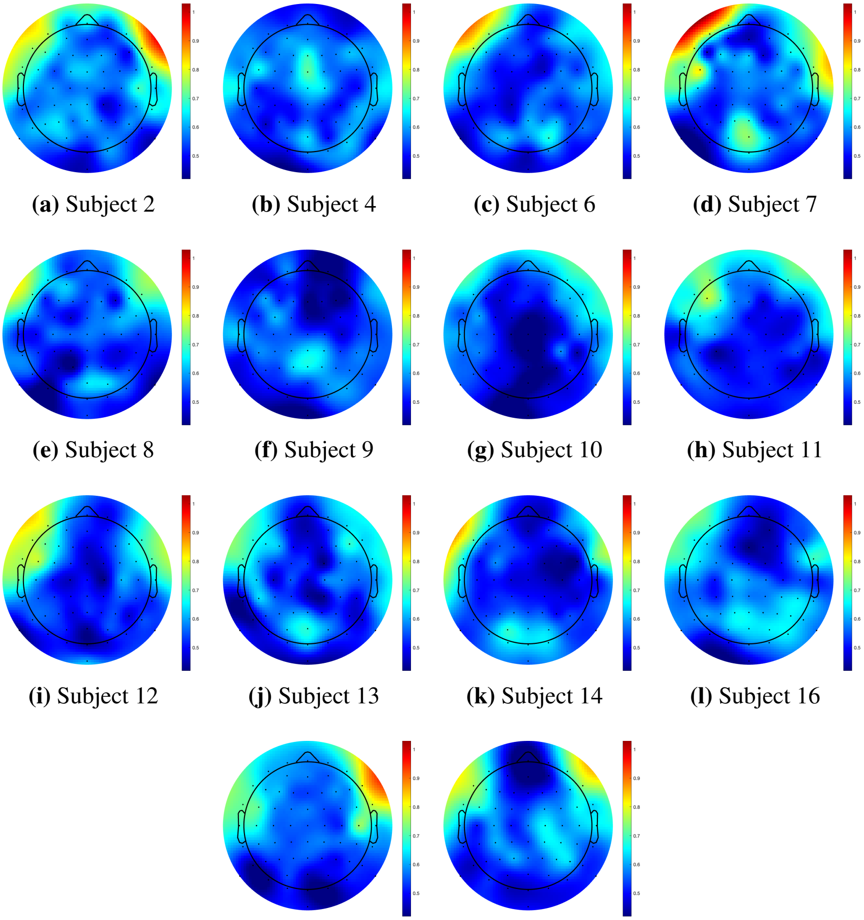
(a) Subject 2 **(b)** Subject 4 **(c)** Subject 6 **(d)** Subject 7

(e) Subject 8 **(f)** Subject 9 **(g)** Subject 10 **(h)** Subject 11

(i) Subject 12 **(j)** Subject 13 **(k)** Subject 14 **(l)** Subject 16

(m) Subject 17 **(n)** Subject 18

Figure A.2 Topographic plots for each subject's electrodes accuracy of predicting between a monologue or a dialogue using CNN1.



(a) Subject 2 **(b)** Subject 4 **(c)** Subject 6 **(d)** Subject 7

(e) Subject 8 **(f)** Subject 9 **(g)** Subject 10 **(h)** Subject 11

(i) Subject 12 **(j)** Subject 13 **(k)** Subject 14 **(l)** Subject 16

(m) Subject 17 **(n)** Subject 18

Figure A.3 Topographic plots for each subject's electrodes accuracy of predicting between a monologue or a dialogue using CNN5.

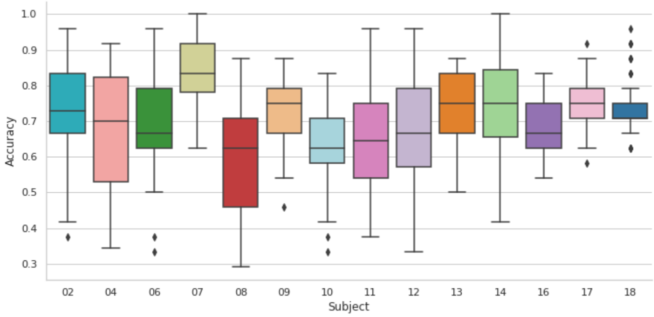


Figure A.4 Box plot of all subjects' electrodes' accuracies of predicting between a monologue or a dialogue for MLP (one layer, 80 nodes).

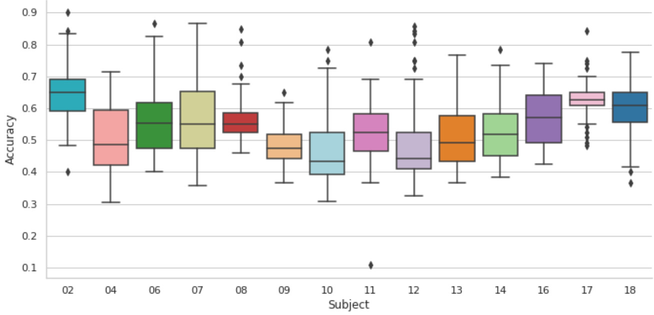


Figure A.5 Box plot of all subjects' electrodes' accuracies of predicting between a monologue or a dialogue for CNN1.

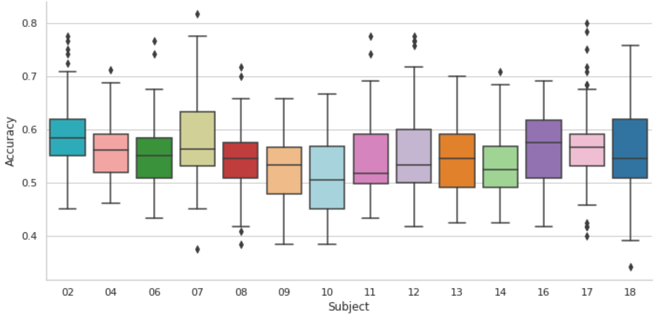


Figure A.6 Box plot of all subjects' electrodes' accuracies of predicting between a monologue or a dialogue for CNN5.

Table A.2 The five best electrodes of each subject when predicting between a monologue or a dialogue for CNN5.

Subject	Electrode 1	Electrode 2	Electrode 3	Electrode 4	Electrode 5
2	F8 : 0.78	AF7 : 0.77	F7 : 0.75	FT8 : 0.74	FT7 : 0.73
4	FCz : 0.71	Fz : 0.69	FC5 : 0.66	T7 : 0.65	Cz : 0.64
6	F7 : 0.77	AF7 : 0.74	Fp1 : 0.68	PO4 : 0.68	AF8 : 0.64
7	FC5 : 0.82	F7 : 0.78	AF7 : 0.77	FT8 : 0.75	POz : 0.74
8	F8 : 0.72	F7 : 0.70	AF8 : 0.66	FT7 : 0.65	PO4 : 0.64
9	Pz : 0.66	CPz : 0.64	P1 : 0.63	P2 : 0.63	T7 : 0.63
10	FT8 : 0.67	AF8 : 0.66	F8 : 0.63	F7 : 0.63	F6 : 0.62
11	F5 : 0.78	AF7 : 0.74	F3 : 0.69	F7 : 0.68	AF8 : 0.68
12	FC5 : 0.78	AF7 : 0.77	F7 : 0.77	FT7 : 0.76	T7 : T7
13	FT7 : 0.70	F6 : 0.68	POz : 0.68	F7 : 0.68	FT8 : 0.64
14	F7 : 0.71	PO3 : 0.68	FT8 : 0.68	FT7 : 0.66	O1 : 0.65
16	FT7 : 0.69	FT8 : 0.69	AF7 : 0.68	F7 : 0.66	PO4 : 0.65
17	F8 : 0.80	FT8 : 0.78	C6 : 0.75	FT7 : 0.72	FC5 : 0.71
18	F7 : 0.76	F8 : 0.75	FT7 : 0.70	AF8 : 0.69	FC5 : 0.68
All	F7	FT7	AF7	FT8	F8

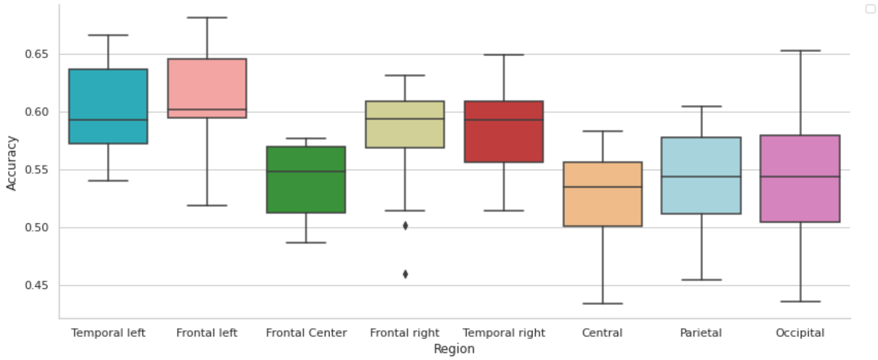


Figure A.7 Performance for electrodes in different regions for CNN5, over all subjects.

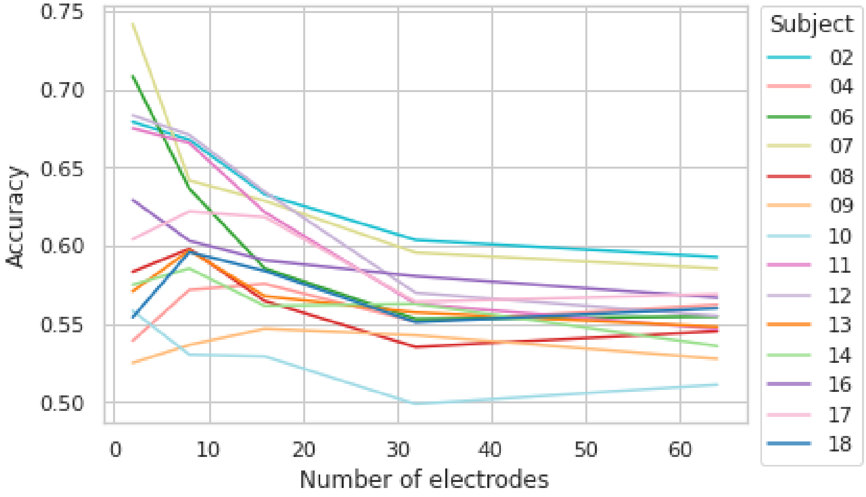


Figure A.8 Performance for each subject when decreasing number of electrodes using CNN5.

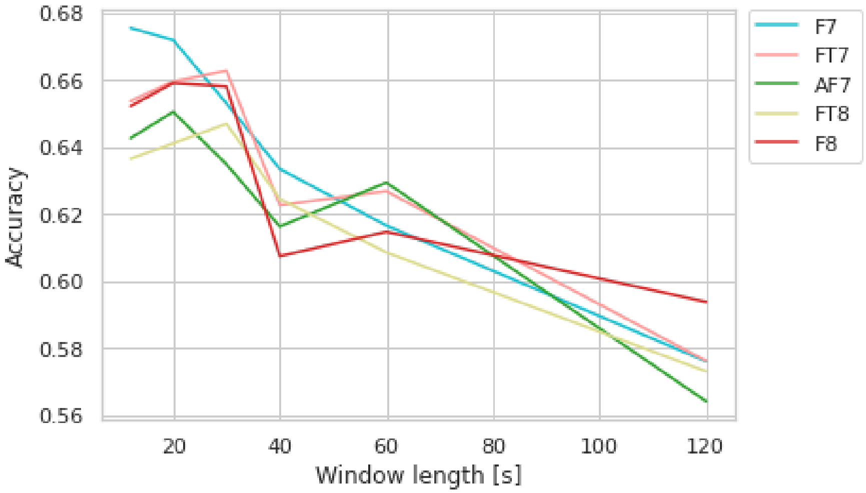


Figure A.9 The scores using EEG to predict a monologue and a dialogue for the five best electrodes of the CNN5 model when it is used different amount of seconds.

A.2 Optimal number of electrodes

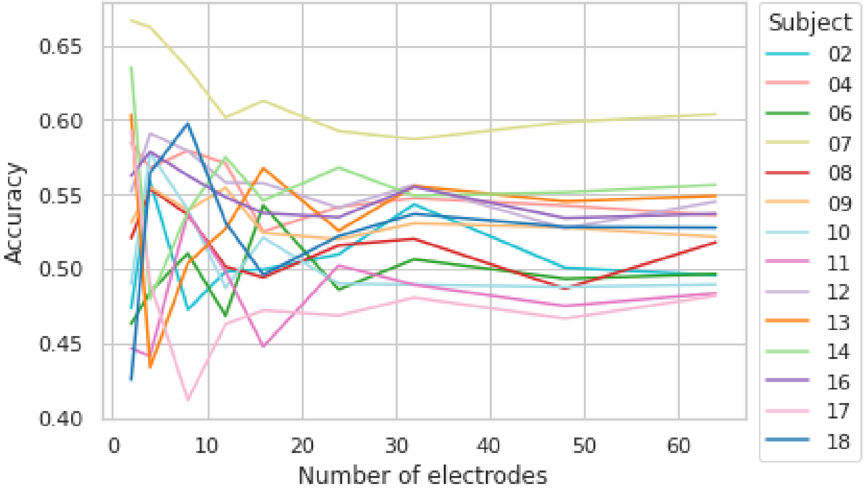


Figure A.10 Performance for each subject when decreasing number of electrodes for the MLP model.

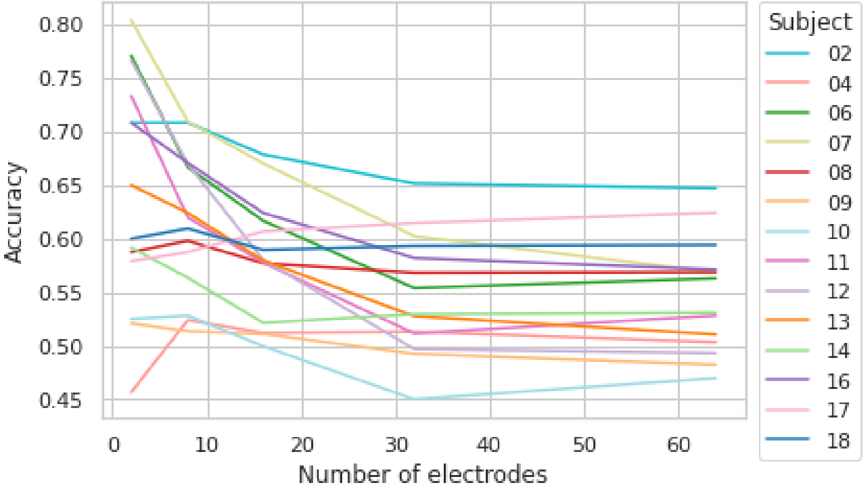


Figure A.11 Performance for each subject when decreasing number of electrodes for the CNN1 model.

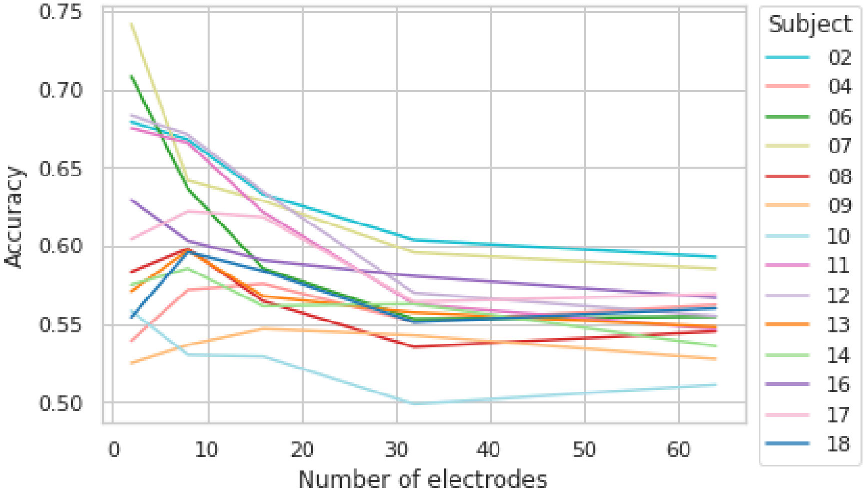


Figure A.12 Performance for each subject when decreasing number of electrodes for the CNN5 model.

A.3 Time dependency eye data

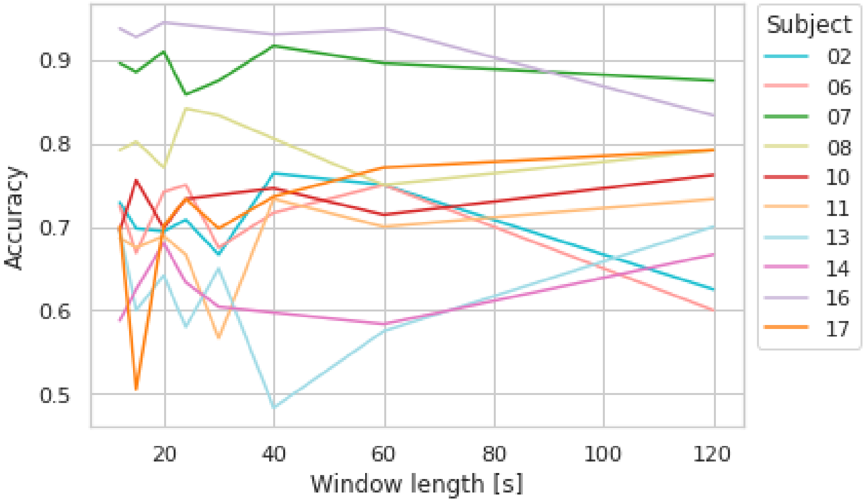


Figure A.13 The scores using eye data to predict a monologue and a dialogue for the MLP using trials of varying length.

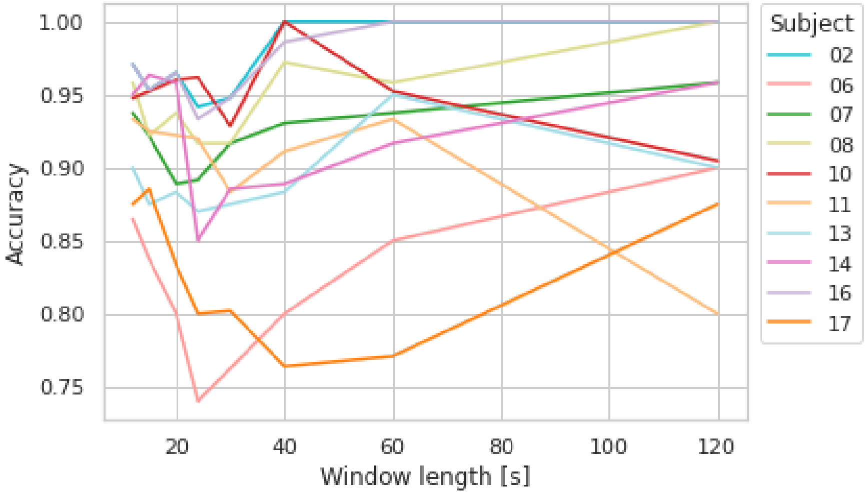


Figure A.14 The scores using eye data to predict a monologue and a dialogue for the CNN1 using trials of varying length.

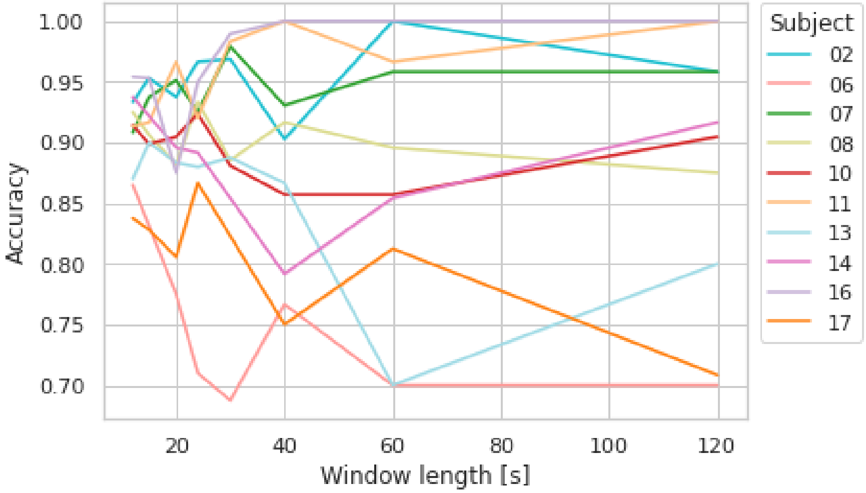


Figure A.15 The scores using eye data to predict a monologue and a dialogue for the CNN5 using trials of varying length.

Lund University **Department of Automatic Control** **Box 118** **SE-221 00 Lund Sweden**	*Document name* MASTER’S THESIS *Date of issue* June 2022 *Document Number* TFRT-6165

Author(s) Sara Enander Louise Karsten	*Supervisor* Emina Alickovic, Eriksholm Research Centre Martin Skoglund, Eriksholm Research Centre Johannes Zaar, Eriksholm Research Centre Bo Bernhardsson, Dept. of Automatic Control, Lund University, Sweden Kristian Soltesz, Dept. of Automatic Control, Lund University, Sweden (examiner)

Title and subtitle

Computation models for audiovisual attention decoding

Abstract

When being in a noisy environment, a normal hearing person can manage to sort out background noise and focus on the attended source. This is something that a person with impaired hearing will struggle with, even when wearing a hearing aid. Research for developing intelligent hearing aids has not yet come up with a solution for solving this problem and more research is needed. This thesis uses data from experiments where a noisy environment is simulated. The test subjects are exposed to a monologue and a dialogue at the same time but are told to only focus on one of them. Using EEG and eye gaze data collected from these experiments as an input, different machine learning models are implemented to solve a binary classification task to predict whether a subject is attending to a monologue or a dialogue. The investigated models are support vector machine, multilayer perceptron, and convolutional neural network. The input to the models is time series arrays from either EEG signals or eye gaze data. For the support vector machine and the multilayer perceptron models, more compact representations of the time series arrays are used as inputs. The convolutional neural network performs best overall and reaches an average prediction score of 87% for all subjects when using inputs from all electrodes at the same time. When using one electrode at the time as input, and then averaging over all electrodes, the support vector machine performs best with an average accuracy of 78%. There is however a clear pattern in what regions of electrodes that succeed best with the classification task for all models. These are the electrodes at the temporal lobe as well as the sides of the front of the frontal lobe. It varies how long the trials need to be to get a decent accuracy for each model when EEG data is used. The support vector machine and the multilayer perceptron performs best for longer trials while the convolutional neural network performs best for shorter trials. For the eye gaze data, the support vector machine reaches the highest average score of 99%. The accuracy for the eye gaze data is not affected remarkably by decreasing the length of trials.

Keywords

Classification system and/or index terms (if any)

Supplementary bibliographical information

ISSN and key title 0280-5316	*ISBN*

Language English	*Number of pages* 1-82	*Recipient’s notes*
Security classification		

http://www.control.lth.se/publications/