An ASR-based Hybrid Approach for Auditory Attention Decoding

Alessandro Celoria

Valentín López

LUND UNIVERSITY

Department of Automatic Control

MSc Thesis
TFRT-6233
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 2024

Abstract

Auditory Attention Decoding (AAD) aims to determine the focus of a listener's attention in environments with multiple overlapping speakers, a challenging situation for hearing impaired patients known as the *Cocktail Party Problem*. This thesis investigates AAD using Whisper, a transformer-based Automatic Speech Recognition (ASR) system that performs a graded transformation from speech to text while encoding linguistic and semantic information in its latent encoder layers. Two approaches to AAD are explored: first, a forward pipeline that utilizes Whisper for preprocessing audio stimuli in conjunction with a Temporal Response Function (TRF) model for predicting Electroencephalography (EEG) responses. Second, a hybrid approach aims to enhance the classification performance by applying Canonical Correlation Analysis (CCA) and its neural network variant, Deep Canonical Correlation Analysis (DCCA), to Whisper's latent encoder layers and EEG signals. The performance of these models is compared across fixed decision window lengths, assessing their attention decoding capabilities when presented with limited information, to highlight Whisper's enhanced performance when combined with CCA. Additionally, we test Whisper's AAD performance when only a restricted number of electrodes limited to the temporal regions is available, as a step towards the development of wearable neurosteered hearing aid devices.

Acknowledgments

I would like to thank our thesis supervisors Bo Bernhardsson, Emina Alickovic and Martin Skoglund for following us through the long journey of our Master Thesis, for encouraging us when we got stuck and providing us invaluable advice and guidance when we most needed it, as well as helping us move our first steps in the interesting world of neurosteered hearing aids.

I would also like to thank the researchers and scientists at Eriksholm Research Centre, who we had the pleasure to meet with during our thesis and who provided us with much insight into this field and many ideas that helped us raise the standard of our work. Thank you to all the lovely people at the Department of Automatic Control of Lund University, for their warmness, for making us feel at home while working on our thesis, and for the interesting chats that entertained us during fika breaks. And above all, thank you Valentín for being a formidable colleague and for your invaluable contribution to this thesis, which would not have been possible without your hard work. All the best for your future!

I would also like to thank all of my friends, who kept me company as we went through this journey together, who listened to my problems, and supported me through the difficult moments.

And last, but definitely not least, I would like to thank my family, who supported me not only through my academic efforts, but also through my entire life, and without whom I would not be where I am today. Thank you to my younger sisters, who encourage me to become a better person every day, and to whom I hope to be as bright of a light as the people that illuminated my path.

Finally, I would like to dedicate this Master Thesis, and the journey it concludes, to *Giuse Merlo*, who I hope to have made proud with how far I've come.

Alessandro Celoria

I thank my thesis supervisors, Bo Bernhardsson, Emina Alickovic, and Martin Skoglund. Your guidance and willingness to help have been crucial to the success of this thesis. I also thank the Department of Automatic Control at Lund University and all of its members for making me feel at home and introducing me to a high-level research environment for the first time.

Thank you to all my university peers, who have become family and have been ready to help me in every possible way. I especially want to thank my teammate, Alessandro Celoria; you are a genuinely high-level programmer and I wish you the best.

To my family in the Dominican Republic, you have been a constant source of support despite the distance. Thank you for helping my dreams come true.

To Jacqueline Franco and her family in Alingsås, Sweden, for taking me in as a son. Jackie, you are a mother to me.

Lastly, I dedicate this thesis to my scholarship sponsors in my home country, especially Ezequiel Díaz and his wife, Carmen Rodríguez, who opened their arms and helped me before coming to Sweden. After my parents, you are the people who have done the most for me.

Valentín López

We are both deeply thankful to the NAISS project for making the computational resources we required to complete our thesis available to us under the project NAISS 2024/22-283.

Alessandro and Valentín

Contents

List of Figures

List of Tables

List of Abbreviations

AAD Auditory Attention Decoding
ASR Automatic Speech Recognition
CCA Canonical Correlation Analysis
CNN Convolutional Neural Network
DCCA Deep Canonical Correlation Analysis
DNN Deep Neural Network
DTW Dynamic Time-Warping
EEG Electroencephalography
ERPs Event-Related Potentials
EPs Evoked Potentials
FDR False Discovery Rate
LLM Large Language Model
MESD Minimal Expected Switch Duration
MCCA Multiway Canonical Correlation Analysis
NN Neural Networks
PCs Principal Components
PCA Principal Component Analysis
RNNs Recurrent Neural Network
SEM Standard Error of the Mean
SCs Summary Components
SVM Support Vector Machines
SPL Sound Pressure Level
TRF Temporal Response Function

1

Introduction

Hearing aids have long enhanced auditory experiences by filtering and amplifying speech information. However, challenges remain in multiple-speaker scenarios, particularly the difficulty of focusing on one speaker in a noisy environment with multiple conversations happening at once (known as the cocktail party problem [Cherry, 1953]). This chapter discusses the *cocktail party problem* and its specific impact on individuals with hearing loss. We propose a methodology that uses electroencephalography (EEG) signals and audio inputs based on state-of-the-art auditory attention decoding (AAD) algorithms, which has the potential to partially solve auditory attention challenges. This novel machine learning architecture, leveraging automatic speech recognition (ASR) systems, could significantly improve the listening experiences of individuals with hearing difficulties.

This section outlines the basis and importance of AAD. It also explains the role of ASR systems in these algorithms, and describes the thesis's two primary goals, providing the reader with a comprehensive overview of the subsequent chapters.

1.1 The Cocktail Party Problem

Imagine entering a crowded room where multiple conversations occur simultaneously. You recognize someone and approach him, joining the conversation. Despite the background noise from other speakers, your brain filters out the unwanted sounds, allowing you to focus on the desired audio source. This phenomenon, known as the *cocktail party problem* [Cherry, 1953], is an inherent brain capability that enables selective auditory attention, as illustrated in Figure 1.1.

Unfortunately, auditory selective attention tasks are challenging for individuals with hearing impairment. Previous research indicates that current hearing aids and cochlear implants do not fully address cocktail party situations. According to [Salorio-Corbetto and Moore, 2023], while hearing aids compensate for threshold elevation and loudness recruitment, they do not mitigate other perceptual effects of

hearing loss. Additionally, [Marrone et al., 2008] studied the effects of hearing aids in multi-talker environments with reverberation, concluding that hearing-impaired listeners showed less spatial release from the masker compared to those with normal hearing.

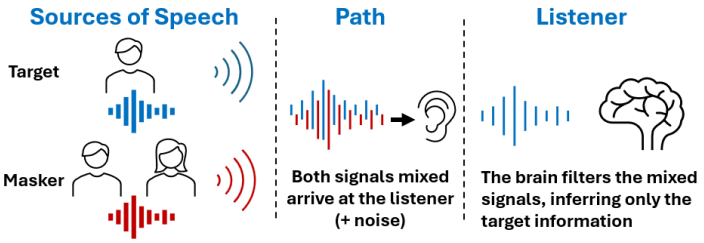


Figure 1.1 The cocktail party problem illustrates the human brain's ability to understand a speaker in a crowded place by filtering out undesired sounds (e.g., background noise, other speakers)

Moreover, hearing aids are currently not a feasible solution when the attended speaker is unknown. Various techniques have been proposed using EEG for speaker separation into **target** and **masker** sources, known as **AAD algorithms** [Alickovic et al., 2019; Geirnaert et al., 2021].

1.2 Automatic Speech Recognition (ASR) Systems

Automatic Speech Recognition (ASR) systems are computational technologies that convert spoken language into text. These systems have advanced significantly and are now integral to applications such as voice-activated assistants and transcription services, with their functionality largely based on Deep Neural Networks (DNNs) [Yu and Deng, 2016]. This work aims to integrate speech-to-text processing into AAD algorithm architectures, not merely as a translator, but to harness the potential of DNNs. We are particularly interested in the linguistic encoding power of the ASR system Whisper, a transformer deep learning architecture composed of an encoder and decoder.

Why Whisper? Several ASR transformers have been used in AAD architectures, such as Hubert and Wav2vec. These transformers employ self-supervised training approaches, pre-trained with unclassified audio data, learning to infer artificially masked speech sounds. This results in discrepancies in how these models represent speech and language across their layers, with inner encoder layers showing lexical and semantic comprehension, and later layers decoding back to speech. Whisper distinguishes itself through its training approach; it is trained in a weakly supervised manner on speech-to-language tasks, improving semantics and linguistics

with layer depth [Anderson et al., 2023]. Additionally, Whisper is trained for speech recognition, translation, and language identification, capable of transcribing 99 languages. This makes the model inherently suited for transcribing Danish audio into text [Radford et al., 2022].

1.3 Scope

The scope of this thesis encompasses two primary goals involving Whisper's implementation. First, for TRF generation to test the model's capabilities for EEG prediction in an attention-selective scheme (target and masker). Second, as an enhancement to an existing AAD architecture. The general details of each proposal are described below:

Whisper for TRF Generation: An AAD pipeline is developed using Whisper's encoder as an audio feature extractor, replacing traditional models (e.g., envelopes, onsets, or surprisal). The objective is to replicate findings in [Anderson et al., 2023] to demonstrate Whisper's capacity to infer the target of attention with audio and EEG data. As illustrated in Figure 1.2, the pipeline can be followed in steps:

1. **Feature Extraction:** Whisper's audio inputs are processed in a sliding window approach in 1/8 s steps. The Whisper base model encoder produces 512 features per sample. The window size ranges from 0.5 s to 30 s (fixed for each experiment, e.g., 10 s). For each window step of 1/8 s, the last six outputs of the encoder are saved, producing a $[512 \times 6]$ matrix for each window step, and the data is up-sampled to 48 Hz.
2. **EEG Generation:** A $[512 \times 6]$ matrix makes the computation of the experiments not possible due to its size. Using PCA, matrices are reduced to $[10 \times 6]$ each step and down-sampled to 32 Hz to match the EEG. The TRF receives the data to predict the brain stimuli response of 64 electrodes.
3. **Correlation Analysis:** Predicted and real EEG signals are compared using Pearson Correlation, producing a correlation coefficient as output. This is done twice for **target** and **masker** labeled audios; the one with the highest correlation with EEG is classified as the attended speaker (target).

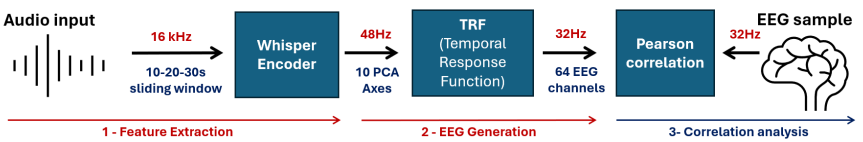


Figure 1.2 Simplified overview of the forward AAD pipeline for EEG prediction using Whisper, inspired by [Anderson et al., 2023]

Whisper + CCA - Hybrid AAD Architecture: Various AAD architectures exist; a forward architecture uses audio features to predict EEG, while a backward architecture uses EEG to reconstruct audio signals. Hybrid architectures utilize both signals to compute a score value. The key difference is that neither EEG nor audio are inferred by the model (e.g., removing the TRF), resulting in faster responses. The aim is to test how Whisper combined with CCA can more accurately predict the target of attention with a smaller window time, as shown in Figure 1.3 [Alickovic et al., 2019; Geirnaert et al., 2021].

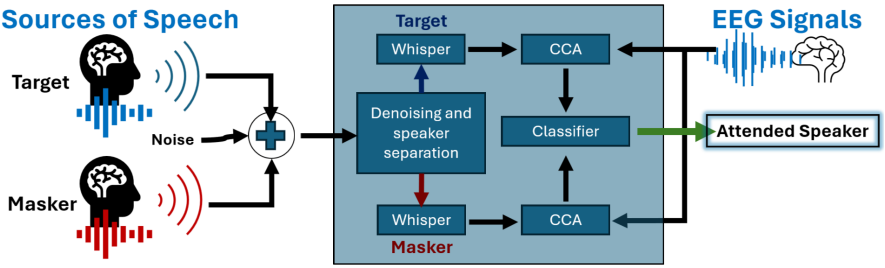


Figure 1.3 Hybrid architecture for AAD. In our experiments, the denoise and speaker separation blocks are discarded, and original audio files are fed to the CCA algorithm without noise. Inspired by [Geirnaert et al., 2021]

Typically, an AAD architecture includes a denoiser and speaker separation block at the beginning; however, this block is excluded from the scope of this thesis for simplification purposes. It is assumed that target and masker audios have already been split and processed.

1.4 Related Work

Our study relies on the results of many previous research projects, both inside and outside the field of cognitive neurosciences. While each of these contributions is fundamental to the existence of this project, some papers have been of particular inspiration to both our research question and methodology:

- **Context and Attention Shape Electrophysiological Correlates of Speech-to-Language Transformation [Anderson et al., 2023]**

  The authors of this paper are the first to attempt forward modeling and EEG prediction using the hidden states of Whisper's encoder. Some of the hyper-parameters used in our models, as well as the method used to extract information from Whisper's encoder, directly stem from this research.

- **A Tutorial on Auditory Attention Identification Methods [Alickovic et al., 2019]**

  This paper helped us formulate our research question. It also introduced us to hybrid modeling and CCA, together with the benefits that their application to complex predictors. The authors touch upon important topics, such as different methods to train CCA models based on multiple trials and the performance benefit of patient-specific models.

- **Electroencephalography-Based Auditory Attention Decoding: Toward Neurosteered Hearing Devices [Geirnaert et al., 2021]**

  This work gave us an invaluable overview of the field, helping us understand where research efforts have been focused in the past and where we could attempt improvements. It also introduced us to the usage of Minimal Expected Switch Duration (MESD) as a unified performance metric for AAD tasks.

2

Background

2.1 Cortical Responses

Humans are endowed with unique capabilities to perceive the world, constantly encoding and inferring sensory inputs to understand their causes, a process known as *perceptual inference*. Learning new causes of these inputs is referred to as *perceptual learning*. Cortical responses represent the brain's encoding of the most likely cause of stimuli [Friston, 2005; Mesgarani and Chang, 2012]. This section provides the foundational understanding necessary for brain activity modeling, beginning with the theory behind stimuli reconstruction, measurement, and modeling as TRFs.

2.1.1 Event-Related Potentials and Evoked Potentials

Event-Related Potentials (ERPs) are transient neural responses characterized by small voltage fluctuations in response to specific events triggered by motor, sensory, or cognitive activities. ERPs are derived from time-aligned EEG signals averaged over multiple trials of the same stimuli, such as a specific sound repeated numerous times. Evoked Potentials (EPs) are a subclass of ERPs, associated with rapid responses under 100 milliseconds, and are typically linked to external stimuli. In contrast, ERPs encompass higher-order mental activities such as language, attention, and memory [Zani, 2013; Sur and Sinha, 2009].

Despite advancements in neuro-imaging techniques, ERPs remain a vital method for studying and replicating cortical response information. ERPs offer a temporal resolution of 1 millisecond, which is crucial for tracking attention and perception activities occurring at a slower pace (around 10 milliseconds). Additionally, potential measurements are obtained directly from scalp-positioned electrodes, eliminating delays associated with the brain's electrical nature [Woodman, 2010].

ERPs are classified by latency and amplitude into various waveforms. The most relevant peaks for attention and perception tasks, known as endogenous responses, typically begin around 300 milliseconds of latency. Key ERPs related to these tasks include [Sur and Sinha, 2009]:

- **P300:** Latency 250 ms to 400 ms. Shorter latencies indicate better cognitive performance, and amplitude levels reflect attention effort.
- **N400:** Latency 300 ms to 600 ms. Negative waveform with an amplitude inversely related to the likelihood of words and sentences.
- **P600:** Occurs following N400, often related to semantic and syntactic errors in the stimulus, individual preferences, or complexity.

2.1.2 Electroencephalography

EEG measures brain activity from the scalp resulting from stimuli responses of millions of cortical neurons, creating an electrical field in the cerebral cortex, which comprises over 10 billion neurons. The cerebral cortex is divided into two hemispheres, each consisting of four lobes based on neuronal functionality: the frontal, temporal, parietal, and occipital lobes.

The cerebral cortex exhibits characteristic rhythmic electrical activity at specific frequencies based on an individual's state, with slower activity during sleep or calmness. Frequency changes correlate with the amplitude of EEG waveforms; calm states produce high-amplitude signals, while alertness results in high-frequency, low-amplitude waveforms. This variation arises from the number of activated neurons in a particular zone at low frequencies versus the chaotic neural information exchange in an alert state.

EEG signals typically range from 0.5 Hz to 40 Hz in frequency and from a few μV to 10 μV in amplitude. A suitable sampling frequency is at least 200 Hz, though higher frequencies are needed for ERP analysis due to their low amplitude (0.1 μV to 10 μV).

Electrode placement follows the International 10-20 system, which assigns letters based on electrode location on the scalp and numbers based on the hemisphere. Electrodes in the central lobe start with C, parietal with P, temporal with T, frontal with F, occipital with O, and auricular with A. Numbers denote the hemisphere, with even numbers on the right, odd numbers on the left, and zero (Z) at the central fissure. Figure 2.1 illustrates this placement. Choosing the appropriate number of electrodes is crucial to avoid spatial aliasing, especially for mapping purposes where 64-electrode configurations are recommended [Sörnmo and Laguna, 2006]. In this thesis, the dataset was recorded using a BioSemi ActiveTwo amplifier with 64 electrodes following the 10-20 system[1].

2.1.3 Temporal Response Functions

TRFs are a valuable tool in signal processing and neuroscience, modeling how systems respond over time to external stimuli. TRFs have been shown to correspond

[1] Refer to Chapter 3 for more details on EEG placement and sampling frequencies.

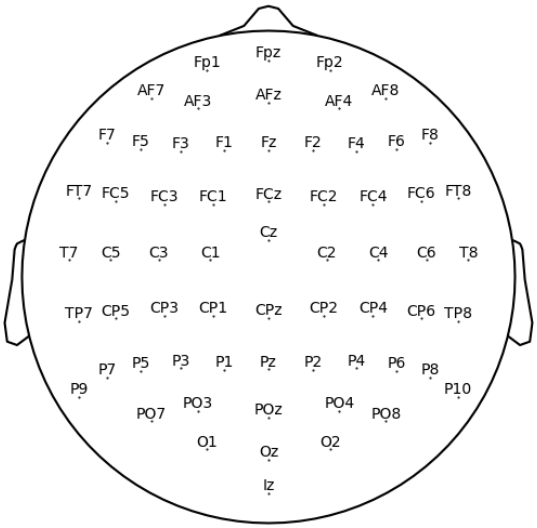


Figure 2.1 BioSemi cap with 64 electrodes placement. The labels are named according to the cortex location and enumerated with even or odd numbers according to their hemisphere. The central fissures are enumerated with Z (zero). Picture generated in Eelbrain [Brodbeck et al., 2023]

to ERPs, serving as impulse response models of the sensory system. System identification creates a mathematical model of how stimuli map to neural responses, typically treating the brain as a linear time-invariant (LTI) system, despite its inherent nonlinearity and time-variance.

There are two primary methods for response mapping: forward and backward. Forward approaches estimate neural responses from the stimuli, providing insights into neural-information encoding, while backward methods focus on reconstructing stimuli features from EEG. This work focuses on forward approaches, demonstrating Whisper's ability to support brain response prediction[2] [Crosse et al., 2016].

Linear Regression A forward LTI TRF response can be modeled as a convolution (2.1), where n represents the channel number (e.g., electrode), $y(t,n)$ is the predicted neural activity over time, and $h(\tau,n)$ is a filter representing the coefficients, weighting the influence of features over the time window $t-\tau$. In what follows $x(t-\tau)$ is the time-lagged input stimulus, $\varepsilon(t,n)$ is the error term, and τ is the time lag, indicating how past values influence the present.

[2] Whisper's encoder extracts audio features in its latent layers.

Symbol	Definition
n	Channel number (electrode) $n = 0, 1, 2, \ldots, N$
N	Number of EEG channels
t	Current sample $t = 0, 1, 2, \ldots, T_{\text{time}}$
T_{time}	Length of the time vector
$y(t,n)$	Predicted neural activity over time for channel n
$h(\tau,n)$	Filter representing the coefficients
$x(t-\tau)$	Time-lagged input stimulus
$\varepsilon(t,n)$	Channel error term
τ	Time lag, $\tau = \tau_{\text{min}}, \ldots, \tau_{\text{max}}$
τ_{min}	Minimum time lag
τ_{max}	Maximum time lag
τ_{window}	Lags window length, defined as $\tau_{\text{max}} - \tau_{\text{min}}$
$\mathbf{H} \in \mathbb{R}^{\tau_{\text{window}} \times N}$	Matrix of filter coefficients
$\mathbf{X} \in \mathbb{R}^{T_{\text{time}} \times \tau_{\text{window}}}$	Stimuli matrix
$\mathbf{Y} \in \mathbb{R}^{T_{\text{time}} \times N}$	Neural response matrix

Table 2.1 Mathematical definitions used in Section 2.1.3.

$$y(t,n) = \sum_{\tau=\tau_{min}}^{\tau_{max}} h(\tau,n)x(t-\tau) + \varepsilon(t,n) \tag{2.1}$$

To estimate an optimal $h(\tau,n)$, the error between measured and predicted EEG signals $\varepsilon(t,n)$ must be minimized, as shown in (2.2), which is solved by (2.3)

$$\text{minimize } \varepsilon(t,n) = \sum_{t}^{T} [y(t,n) - \hat{y}(t,n)]^2 \tag{2.2}$$

$$\mathbf{H} = [\mathbf{X}^\mathsf{T}\mathbf{X}]^{-1}\mathbf{X}^\mathsf{T}\mathbf{Y}. \tag{2.3}$$

Computing neural responses as a product $\mathbf{Y} = \mathbf{XH}$, (2.6) becomes $T_{\text{time}} \times \tau_{\text{window}}$, with T_{time} the time vector length, and the lags window $\tau_{\text{window}} = \tau_{\text{max}} - \tau_{\text{min}}$ the time interval of TRF prediction. The matrix defined in (2.5), contains $\tau_{\text{window}} \times N$ weights, with N representing the number of EEG channels. Finally, the predictions defined in (2.4), have dimensions $T_{\text{time}} \times N$ [Crosse et al., 2016]. In experiments, the TRF stimuli window interval is defined between -100 ms and 700 ms because the goal is to predict semantic and linguistic information, targeting ERPs such as the P300 and N400[3].

[3] Refer to Chapter 4 for more details on the TRF implementation.

The matrix $\mathbf{Y}$ represents the neural response matrix and is defined as follows:

$$\mathbf{Y} = \begin{bmatrix} y(1,1) & y(1,2) & \cdots & y(1,N) \\ y(2,1) & y(2,2) & \cdots & y(2,N) \\ \vdots & \vdots & \cdots & \vdots \\ y(T_{\text{time}},1) & y(T_{\text{time}},2) & \cdots & y(T_{\text{time}},N) \end{bmatrix} \tag{2.4}$$

where each element $y(t,n)$ represents the neural activity at time t for channel n. The rows correspond to different time points t, while the columns correspond to different EEG channels n.

The matrix $\mathbf{H}$ represents the filter coefficients and is defined as follows:

$$\mathbf{H} = \begin{bmatrix} h(\tau_{\min},1) & h(\tau_{\min},2) & \cdots & h(\tau_{\min},N) \\ h(\tau_{\min}+1,1) & h(\tau_{\min}+1,2) & \cdots & h(\tau_{\min}+1,N) \\ \vdots & \vdots & \cdots & \vdots \\ h(\tau_{\max},1) & h(\tau_{\max},2) & \cdots & h(\tau_{\max},N) \end{bmatrix} \tag{2.5}$$

where each column $\mathbf{h}_n$ is a vector of coefficients for the n-th EEG channel, and each row corresponds to a specific time lag τ.

$$\mathbf{X} = \begin{bmatrix} x(1-\tau_{\min}) & x(-\tau_{\min}) & \cdots & x(1) & 0 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots & x(1) & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots & \vdots & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \cdots & x(1) \\ x(T_{\text{time}}) & \vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\ 0 & x(T_{\text{time}}) & \cdots & \vdots & \vdots & \cdots & \vdots \\ \vdots & 0 & \cdots & \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & x(T_{\text{time}}) & x(T_{\text{time}}-1) & \cdots & x(T_{\text{time}}-\tau_{\max}) \end{bmatrix} \tag{2.6}$$

The matrix $\mathbf{X}$ defined in (2.6) represents a lagged time series of the input stimuli. In this matrix, the rows correspond to time samples, while the columns contain the lagged values of the stimuli, effectively shifting the lags over the time dimension. In this configuration, samples to the left represent future predictions, and those to the right represent past information.

To illustrate, consider the first row, where the sample $x(1)$ is positioned. To preserve causality, a series of zeros is padded to the right of the current sample, effectively padding the negative indices. As the window moves further from the edges of the

signal, the padding on the right gradually diminishes. Conversely, as the window approaches the end of the sequence and data availability decreases, the padding reappears from the left, maintaining the structural integrity of the lagged series throughout the entire duration of the analysis [Crosse et al., 2016].

Regularization A sufficiently large lag window does not guarantee optimal signal reconstruction; the TRF estimation can become numerically unstable when dealing with large time intervals [Alickovic et al., 2023]. Additionally, EEG data can be particularly noisy, introducing small random patterns in the training data. This noise complicates the estimation of coefficients and the prediction of a generalized brain response from stimuli input. When a model erroneously learns these noise patterns (variance), it results in overfitting. The solution is to have a better noise model or introduce a penalty in the Least Squares formulation (2.3), known as regularization.

Regularization helps ensure that coefficients have similar neighboring weights (Ridge Regression) or pushes them toward zero for sparse estimation (LASSO), aiding in feature selection. This penalty, represented by λ, increases the degree of regularization applied to the model. The value of λ must be carefully selected, as excessive regularization can hinder the model's ability to generalize effectively [Alickovic et al., 2019; Crosse et al., 2016; Alickovic et al., 2023].

2.2 Models of Auditory Attention Decoding

As discussed in Chapter 1, AAD addresses the cocktail party problem by aiming to extract the source of attention directly from the brain. There are three primary architectures, each employing a different prediction methodology but sharing common reconstruction algorithm principles to some extent:

- **Forward Modeling (Encoding):** These models predict EEG signals using stimuli input (e.g., envelopes, onsets, word surprisals). They are termed forward because they preserve causality, predicting EEG based on prior data.
- **Backward Modeling (Decoding):** These models utilize future EEG information to reconstruct the previous input stimuli.
- **Bidirectional Modeling (Hybrid):** These models use both EEG and sound stimuli to generate a score similarity value between the sources, combining forward and backward architectures, e.g. CCA.

Figure 2.2 presents an overview of the AAD algorithms in a two-part illustration. Part 1 (upper) outlines the general architecture of AAD algorithms, beginning with the denoising and reconstruction of split speaker audio sources. The AAD block can employ any forward, backward, or hybrid algorithm, with the classifier ranging from

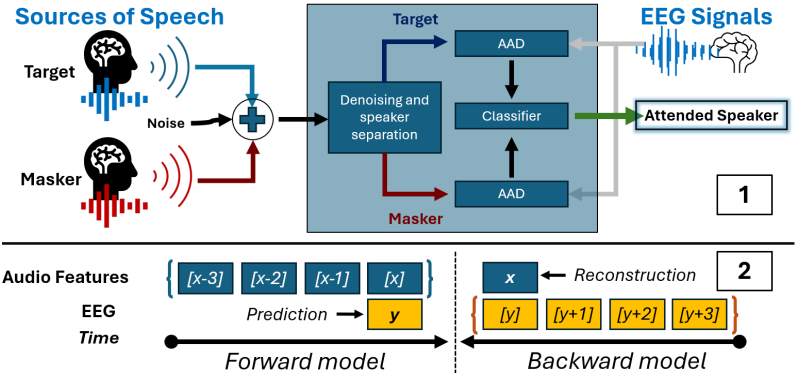


Figure 2.2 AAD explained in a two-part figure. Part 1 (upper) illustrates the general architecture of AAD algorithms. Part 2 (lower) compares the differences between forward and backward architectures.

a simple maximization procedure to a more complex machine learning task. Part 2 (lower) highlights the differences between forward and backward architectures. The backward architecture suggests the model's output is the reconstructed audio input, though it could represent any of its features, such as the envelope or onsets [Alickovic et al., 2019; Geirnaert et al., 2021]. [4]

2.2.1 Minimal Expected Switch Duration (MESD)

Typically, AAD algorithms are compared using a *correlation performance curve*, which visualizes accuracy over different audio time spans for the auditory selective task through various *decision window lengths*. This provides insight into the time required for a real-time algorithm to switch to the newest source of attention. However, there is a trade-off in finding an optimal point on the curve, as accuracy improves at the cost of a longer decision window, see Figure 2.3.

The MESD was proposed to address this trade-off by formulating an optimization problem that seeks the shortest switching duration within a predefined stable working region. Essentially, it identifies the optimal point on the curve and returns a metric that quantifies the performance of the AAD algorithm. A lower value indicates better performance. However, as noted in [Geirnaert et al., 2021], MESD is a theoretical metric for comparative purposes and does not necessarily reflect actual switching operation time [Geirnaert et al., 2020; Alickovic et al., 2019].

MESD Estimation The MESD can be estimated following the approach described in [Geirnaert et al., 2020] by modeling an adaptive gain control system as a

[4] The design of the denoiser and speaker separation block is beyond the scope of this thesis.

Markov chain with N states. These states correspond to different levels of gain applied to the target of attention relative to the background. Each state has a transition probability p towards the next higher amplification level equal to the algorithm's accuracy for a given decision window length τ, while the complementary probability $q = 1 - p$ results in a transition to a lower state.

In this framework, the operation of an intelligent hearing aid is modeled as a random walk on this Markov chain, where each step takes τ seconds. The number of states is optimized to meet specific user-oriented operational parameters, namely the *comfort level* c and the *confidence level* P_0. The comfort level c specifies the minimum amplification level necessary for comfortable hearing in a cocktail party environment, and the confidence level P_0 denotes the desired probability that the hearing aid remains within the *comfort region* $[c, 1]$ at any given time [Geirnaert et al., 2020].

Once the optimal number of states is determined[5], the Expected Switch Duration (ESD) can be calculated as the expected hitting time for the first state corresponding to a gain level $k_c \geq c$ starting from a state $i < k_c$, where each step takes τ seconds to complete. This process is repeated for each recorded (p, τ) pair to derive the *Minimal Expected Switch Duration* or MESD.

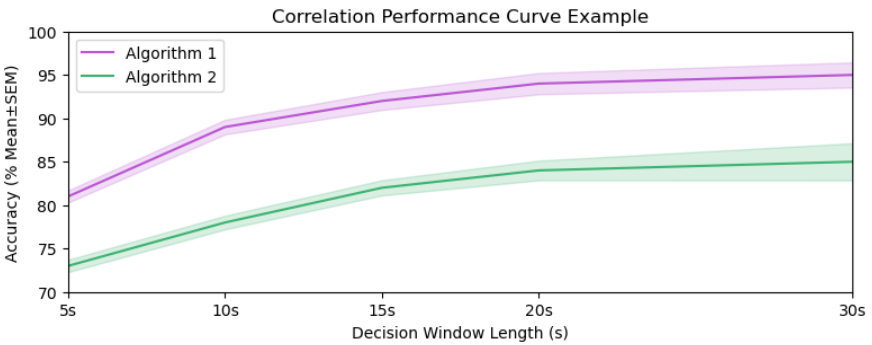


Figure 2.3 Example of correlation performance curve between two AAD algorithms: accuracy improves at the cost of the longest decision window length. This figure is only for illustrative purposes and does not correspond to the thesis results.

[5] The number of states N is typically lower-bounded by a chosen minimal number of states $N_{\min}$ to ensure sufficiently smooth gain transitions [Geirnaert et al., 2020].

2.3 Machine Learning

Machine Learning can be described as statistical methods that enable the learning of patterns without explicit instructions, thereby creating algorithms capable of generalizing information from data. This section is dedicated to machine learning techniques of significant importance to the methodologies discussed. It covers Neural Networks (NN)s initially described for the MCCA algorithm, as well as PCA and Support Vector Machines (SVM) used as tools for achieving the primary objectives of this work.

2.3.1 Neural Networks

NNs are a class of machine learning algorithms composed of interconnected neurons that emulate the behavior of the human brain. The Perceptron is the most basic building block, consisting of weighted inputs that are summed and passed through an activation function. Perceptrons can be combined into multiple layers, creating a NN model known as a Multilayer Perceptron (MLP), which typically comprises three layers: an input layer to receive data, hidden layers for computation and feature extraction, and an output layer for producing classification probabilities or regression values. Figure 2.4 illustrates a typical Multilayer Perceptron NN. Other architectures include Convolutional Neural Networks (CNNs) and autoencoders, which are used in image recognition and data compression, respectively.

Training Before performing inference, a neural network must be trained to achieve the desired objective. This is an iterative procedure that can be summarized in five steps:

1. **Data Preprocessing:** This step involves scaling the data to normalize the input features, resulting in faster convergence. During this step, the dataset is also split into validation, training, and testing subsets to evaluate the model's performance.
2. **Forward Propagation:** The data is passed through the NN in the forward direction from input to output.
3. **Loss Calculation:** This metric compares the performance of the network's output against the true target. Common loss functions include Mean Squared Error (MSE) for regression tasks and Cross-Entropy Loss for classification tasks.
4. **Backpropagation:** The loss is propagated backward through the network from the output to the input. Optimizers, such as Stochastic Gradient Descent (SGD), are employed to achieve faster convergence.
5. **Weight Updates:** The network's weights are updated based on the current loss to improve performance in the next iteration.

Hyperparameters Hyperparameters are configurations established prior to training that should not be confused with the NNs parameters learned from the data, such as weights. Hyperparameters can encompass the size of the neural network and training conditions. The most common hyperparameters include:

- **Learning Rate:** This defines the rate at which the NN weights are adjusted towards minimizing the loss function with each iteration.
- **Number of Epochs:** This indicates the number of times the NN passes through the entire dataset.
- **Batch Size:** This refers to the number of training samples processed in one iteration. Smaller batch sizes help avoid overfitting but may prolong the training process.
- **Number of Hidden Layers:** This pertains to the NN architecture and its complexity. Larger models may perform better on complex tasks but also increase the risk of overfitting.
- **Activation Functions:** These introduce nonlinearities into the model, enhancing the NN ability to learn from the dataset. Common activation functions include sigmoid, tanh, and ReLU.
- **Regularization:** These are techniques to prevent overfitting, such as early stopping, dropout, or L2-regularization.

Effective Deep Neural Network (DNN) implementation involves careful consideration of hyperparameters in accordance with the problem requirements and managing performance metrics such as confusion matrices and loss curves [Goodfellow et al., 2016; LeCun et al., 2015].

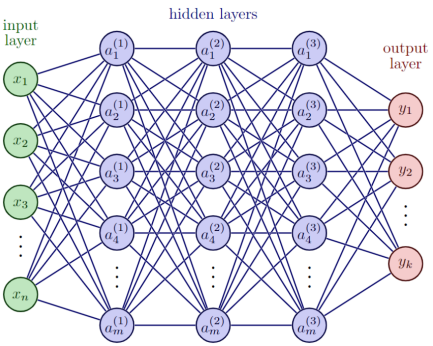


Figure 2.4 Multilayer Perceptron (MLP) architecture, consisting of an input layer for receiving data, hidden layers for computations and feature extraction, and an output layer for classification probabilities or regression values.

2.3.2 Transformers

Transformers are a type of DNN architecture comprising an encoder and a decoder, providing a novel approach to solving sequence prediction and transduction tasks. Historically, these tasks were exclusively addressed by Recurrent Neural Network (RNNs), which incurred long computational times due to their sequential processing nature. RNNs inherently preclude parallelism as perceptrons are sequentially connected in the hidden layer, requiring the synchronized processing of temporal data. Transformers address this issue by introducing the concept of *attention*, which allows the model to focus on different parts of the same sequence and compute multiple features simultaneously [Vaswani et al., 2017].

2.3.3 Whisper

Whisper [Radford et al., 2022] is a transformer-based encoder-decoder ASR system. The encoder transforms the input speech into a latent representation, referred to as *linguistic embedding*, while the decoder converts this series of feature vectors into a word sequence. This process is based in part on the selected (or inferred) audio language and task (*transcription* or *translation*). Whisper's encoder is of particular interest for this research, specifically the hidden states of the attention layers it comprises. Given that Whisper is capable of identifying, understanding, and transcribing 99 languages, as well as translating them into English, it is reasonable to infer that the encoder's representation of speech generalizes effectively across languages. Additionally, due to the robustness of the model and its weakly-supervised and translation-focused training, it is likely that it also generalizes well across accents, conditions, and noise levels [Anderson et al., 2023].

The model is available in several variants, ranging from `whisper-tiny` with 39M parameters to `whisper-large-v3` with 1550M parameters. Whisper distinguishes itself from previous ASR systems not only by its capability to operate across a variety of languages but also by its notable accuracy and robustness, which approach human levels [Radford et al., 2022]. This high level of performance has been achieved through *weakly-supervised training*, a method that contrasts with traditional techniques used to train advanced ASR models. On one hand, models like *Wav2Vec* [Baevski et al., 2020] exploit unsupervised pre-training followed by supervised fine-tuning, allowing training on enormous amounts of unvetted data. This results in a strong encoder at the expense of a weaker decoder which is more challenging to fine-tune. On the other hand, models trained entirely in a supervised fashion on curated, yet small, datasets achieve a symmetrical tradeoff [Radford et al., 2022]. Whisper, however, employs *weak supervision*, which involves using large amounts of unvetted data with automatically generated, best-effort labels, with only a subset of the dataset being manually labeled. This approach strikes a balance between quality and quantity, enabling the model to learn from a large amount of noisier data [Radford et al., 2022]. This methodology has resulted in a model that

is both robust and capable, while minimizing the effort required in the creation of suitable training data.

2.3.4 Principal Component Analysis

PCA is a statistical linear dimensionality reduction technique that transforms data into a new coordinate space that maximizes the preserved variance. The principal components are a set of eigenvectors that provide the transformation coordinates, which are uncorrelated with each other and ordered in descending order based on their corresponding eigenvalues, with the first eigenvectors explaining the most variance [Shlens, 2014].

2.3.5 Support Vector Machines

SVM is a supervised machine learning algorithm that determines the optimal hyperplane to separate different classes in the feature space, maximizing the margin between the closest points of the classes. These points are known as support vectors. The mathematical foundations of SVMs are based on linear classifiers, where the hyperplane can be defined by a simple linear equation, with weights to be determined. However, linear formulations are not always feasible, and nonlinear transformations, known as *kernel tricks*, are applied to map the data into a higher-dimensional space where the classes become linearly separable [Hastie et al., 2009]. For the scope of this thesis, we will focus on linear SVMs used as classifiers at the end of the pipelines for the target of attention.

2.3.6 Cross-validation

K-fold cross-validation is a method used to estimate model generalization by randomly splitting the dataset into K parts of equal size, using one part for validation and the remaining parts for training. This procedure is repeated K times, with the validation results averaged. This method is useful for generating confident generalizations from insufficient data, though it is time-consuming as the model is repetitively trained over the folds. Figure 2.5 illustrates the data partitions in a K-fold cross-validation implementation [Hastie et al., 2009].

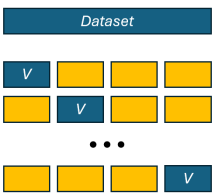


Figure 2.5 Example of K-fold cross-validation. Data is randomly split into K parts of equal size.

2.4 Canonical Correlation Analysis

CCA is a statistical method designed to identify correlations between two multivariate datasets. Although it is often regarded as a dimensionality reduction technique, in the context of AAD, it is employed as a forward and backward approach (hybrid), obviating the need for a predictive model such as the TRF. The fundamental principle of CCA is to find linear combinations that maximize the correlation between the two datasets.

Symbol	Definition
i	Sample $i = 0, 1, 2, \ldots, N$
N	Number of samples
p	Number of features in dataset $\mathbf{X}$
q	Number of features in dataset $\mathbf{Y}$
$\mathbf{X} \in \mathbb{R}^{N \times p}$	Multivariate dataset of N samples and p features
$\mathbf{Y} \in \mathbb{R}^{N \times q}$	Multivariate dataset of N samples and q features
$x_i \in \mathbb{R}^p$	Sample vector from dataset $\mathbf{X}$ at time instance i
$y_i \in \mathbb{R}^q$	Sample vector from dataset $\mathbf{Y}$ at time instance i
$\bar{x} \in \mathbb{R}^p$	Mean vector from dataset $\mathbf{X}$, $\bar{x} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{X}_{i\cdot}$
$\bar{y} \in \mathbb{R}^q$	Mean vector from dataset $\mathbf{Y}$, $\bar{y} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{Y}_{i\cdot}$
$\mathbf{w}_x \in \mathbb{R}^p$	Linear combination weights vector for dataset $\mathbf{X}$
$\mathbf{w}_y \in \mathbb{R}^q$	Linear combination weights vector for dataset $\mathbf{Y}$
$\mathbf{W}_x \in \mathbb{R}^{p \times K}$	Matrix of linear combination vectors for dataset $\mathbf{X}$
$\mathbf{W}_y \in \mathbb{R}^{q \times K}$	Matrix of linear combination vectors for dataset $\mathbf{Y}$
k	Current linear combination vector $k = 0, 1, 2, \ldots, \min(p, q)$
K	Number of linear combinations $\min(p, q)$
$\mathbf{R}_{xx} \in \mathbb{R}^{p \times p}$	Covariance matrix within dataset $\mathbf{X}$
$\mathbf{R}_{yy} \in \mathbb{R}^{q \times q}$	Covariance matrix within dataset $\mathbf{Y}$
$\mathbf{R}_{xy} \in \mathbb{R}^{p \times q}$	Covariance matrix between datasets $\mathbf{X}$ and $\mathbf{Y}$
$\overline{\mathbf{X}} \in \mathbb{R}^{N \times p}$	Mean matrix of dataset $\mathbf{X}$, where $\overline{\mathbf{X}} = \mathbf{X} - 1_N \bar{x}^T$
$\overline{\mathbf{Y}} \in \mathbb{R}^{N \times q}$	Mean matrix of dataset $\mathbf{Y}$, where $\overline{\mathbf{Y}} = \mathbf{Y} - 1_N \bar{y}^T$

Table 2.2 Mathematical definitions for CCA, Section 2.4.

In CCA, the two multivariate datasets are represented as matrices $\mathbf{X} \in \mathbb{R}^{N \times p}$ and $\mathbf{Y} \in \mathbb{R}^{N \times q}$, defined in (2.7) and (2.8), where N denotes the number of samples, and p and q denote the number of features in each dataset, respectively.

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{N1} & x_{N2} & \cdots & x_{Np} \end{bmatrix} \tag{2.7}$$

$$\mathbf{Y} = \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1q} \\ y_{21} & y_{22} & \cdots & y_{2q} \\ \vdots & \vdots & \ddots & \vdots \\ y_{N1} & y_{N2} & \cdots & y_{Nq} \end{bmatrix} \tag{2.8}$$

The linear combination weights are denoted $\mathbf{w}_x \in \mathbb{R}^p$ and $\mathbf{w}_y \in \mathbb{R}^q$, and are extended into the weight matrices $\mathbf{W}_x \in \mathbb{R}^{p\times K}$ and $\mathbf{W}_y \in \mathbb{R}^{q\times K}$, defined in (2.9) and (2.10), where K is the number of linear combinations, equal to the minimum of p and q.

$$\mathbf{W}_x = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1K} \\ w_{21} & w_{22} & \cdots & w_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ w_{p1} & w_{p2} & \cdots & w_{pK} \end{bmatrix} \quad (2.9) \qquad \mathbf{W}_y = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1K} \\ w_{21} & w_{22} & \cdots & w_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ w_{q1} & w_{q2} & \cdots & w_{qK} \end{bmatrix} \quad (2.10)$$

The definition of Pearson correlation (2.11) provides the foundation for CCA, where x_i, y_i are vectors at a particular time instance $i = 0, 1, 2, \ldots, N$ and $\bar{x}$, $\bar{y}$ are mean vectors for (2.7) and (2.8) respectively.

$$r = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{N}(y_i - \bar{y})^2}} \tag{2.11}$$

To compute the correlation between the datasets, we first calculate the covariance matrices. (2.12) represents the covariances within (2.7), (2.13) represents the covariances within (2.8), and (2.14) represents the covariances between (2.7) and (2.8).

$$\mathbf{R}_{xx} = \frac{1}{N-1}(\mathbf{X} - \overline{\mathbf{X}})^\mathsf{T}(\mathbf{X} - \overline{\mathbf{X}}) \tag{2.12}$$

$$\mathbf{R}_{yy} = \frac{1}{N-1}(\mathbf{Y} - \overline{\mathbf{Y}})^\mathsf{T}(\mathbf{Y} - \overline{\mathbf{Y}}) \tag{2.13}$$

$$\mathbf{R}_{xy} = \frac{1}{N-1}(\mathbf{X} - \overline{\mathbf{X}})^\mathsf{T}(\mathbf{Y} - \overline{\mathbf{Y}}) \tag{2.14}$$

The goal of CCA is to maximize the correlation between the datasets, formulated as an optimization problem (2.15)

$$\mathbf{w}_x, \mathbf{w}_y = \underset{\mathbf{w}_x, \mathbf{w}_y}{\operatorname{argmax}} \frac{\mathbf{w}_x^\mathsf{T}\mathbf{R}_{xy}\mathbf{w}_y}{\sqrt{\mathbf{w}_x^\mathsf{T}\mathbf{R}_{xx}\mathbf{w}_x}\sqrt{\mathbf{w}_y^\mathsf{T}\mathbf{R}_{yy}\mathbf{w}_y}}. \tag{2.15}$$

This optimization problem is solved through eigenvalue decomposition. Each eigenvector corresponds to a new linear combination, extending the weights as (2.9) and (2.10). The eigenvectors are mutually orthogonal within each weight matrix, and the pairs $(\mathbf{X}\mathbf{w}_x^{(k)}, \mathbf{Y}\mathbf{w}_y^{(k)})$ are maximally correlated and organized in descending order, with the first pair of canonical variates ($k = 1$) best explaining the multivariate sets with the highest possible correlation [Gundersen, 2018; Geirnaert et al., 2021].

Application in Auditory Attention Decoding As discussed in Section 2.1.3, feature vectors can significantly increase in dimensionality when time lags ($t - \tau$) are included. For CCA, this introduces a trade-off between flexibility and overfitting;

more features enhance the possibility of data fitting but also increase the risk of misleading results. Moreover, differences in temporal alignment between EEG and audio features present another challenge as auditory latencies between EEG and audio are not precisely known, and computational delays inherent to CCA must be managed. To mitigate this, a series of time shifts is introduced in conjunction with the time lags, selecting the shift that maximizes the correlation, which then becomes a hyper-parameter to be iteratively tested [Cheveigné et al., 2018].

2.4.1 Multiway Canonical Correlation Analysis

Developing a generalized model for multiple datasets is often challenging in the context of AAD. Specifically, constraints exist within CCA as it is traditionally limited to computing correlations between two datasets. MCCA extends CCA by allowing multiple data matrices to be concatenated, thereby maximizing the correlation across multidimensional datasets. Various formulations for MCCA exist; herein, a simplified approach is presented.

Symbol	Definition
N	Total number of samples
M	Total number of datasets
p_j	Total number of features in a dataset $\mathbf{X}_j$
P	Total number of features, $P = \sum_{j=1}^{M} p_j$
j	Index for the particular dataset, $j = 1, 2, \ldots, M$
$\mathbf{X}_j \in \mathbb{R}^{N \times p_j}$	Individual dataset j with N samples and p_j features
$\mathbb{X} \in \mathbb{R}^{N \times P}$	Superset matrix combining all individual datasets $\mathbf{X}_j$
$\widehat{\mathbf{X}}_j \in \mathbb{R}^{N \times p_j}$	Whitened matrix of $\mathbf{X}_j$ with normalized Principal Components
$\mathbf{Z} \in \mathbb{R}^{N \times P}$	Second PCA matrix with N samples and P Summary Components

Table 2.3 Mathematical definitions for MCCA, Section 2.4.1

An individual dataset $\mathbf{X}_j$ is defined in (2.16), and by combining all individual datasets, (2.17) is defined as a superset.

$$\mathbf{X}_j = \begin{bmatrix} x_{11}^j & x_{12}^j & \cdots & x_{1p_j}^j \\ x_{21}^j & x_{22}^j & \cdots & x_{2p_j}^j \\ \vdots & \vdots & \ddots & \vdots \\ x_{N1}^j & x_{N2}^j & \cdots & x_{Np_j}^j \end{bmatrix} \quad (2.16) \qquad \mathbb{X} = \begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 & \cdots & \mathbf{X}_M \end{bmatrix} \quad (2.17)$$

The whitened matrix $\widehat{\mathbf{X}}_j$ with horizontally stacked normalized Principal Components (PCs) $\widehat{\mathbf{PC}}_1, \widehat{\mathbf{PC}}_2, \ldots$ (each with norm 1) is defined in (2.18) and the Second PCA matrix $\mathbf{Z} \in \mathbb{R}^{N \times P}$ as (2.19) where $\mathbf{SC}_1, \mathbf{SC}_2, \ldots, \mathbf{SC}_P$ are the column vectors representing the Summary Components (SCs).

$$\widehat{\mathbf{X}}_j = \begin{bmatrix} | & | & & | \\ \widehat{\mathbf{PC}}_1 & \widehat{\mathbf{PC}}_2 & \cdots & \widehat{\mathbf{PC}}_{p_j} \\ | & | & & | \end{bmatrix} \quad (2.18) \qquad \mathbf{Z} = \begin{bmatrix} | & | & & | \\ \mathbf{SC}_1 & \mathbf{SC}_2 & \cdots & \mathbf{SC}_P \\ | & | & & | \end{bmatrix} \quad (2.19)$$

Following the notations in Table 2.3, Figure 2.6 formulates MCCA as a two steps PCA algorithm:

1. **First PCA:** After defining (2.17), PCA is applied on each (2.16). This yields a new transformation matrix where columns are PCs that contain specific variance information and are uncorrelated to each other within (2.16). However, as the purpose of MCCA is to explain relationships between datasets, a final step is to standardize the individual dataset by scaling the PCs to the unit norm. This overall process is known as *Whitening Transformation*, resulting in (2.18).

2. **Second PCA:** The set of (2.18) is concatenated and a final PCA is applied, to yield (2.19) where columns are SCs.

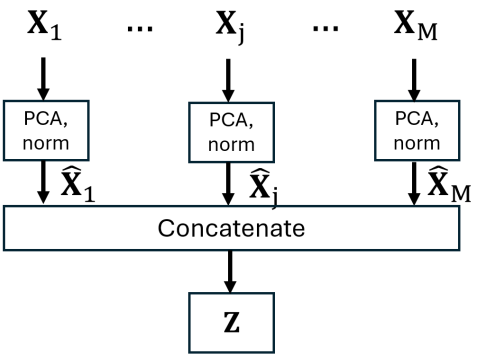


Figure 2.6 Block diagram of MCCA architecture, explaining the two PCA processes. Figure courtesy of [Cheveigné et al., 2019].

This MCCA approach is motivated by several factors. First, individual data matrices are spatially whitened, transforming their covariance to the identity matrix, which results in uncorrelated vectors that may have a denoising effect. Second, by applying PCA on the concatenated whitened data, SCs are obtained. The variance of each column signifies the presence of a specific component in the superset, organized in descending order, indicating that the most shared temporal patterns are found in the first columns [Cheveigné et al., 2019; Parra, 2018].

2.4.2 Deep Canonical Correlation Analysis

Deep Canonical Correlation Analysis (DCCA) is a nonlinear variant of CCA that employs two NNs, one for each dataset, with the goal of maximizing the correlation between their outputs. NNs are a robust parametric method for learning nonlinear representations, offering flexibility to control model capabilities. DCCA particularly allows the definition of each NN characteristics independently, tailoring the architecture and activation functions to the needs of each dataset.

$$(\theta_1^*, \theta_2^*) = \underset{(\theta_1^*, \theta_2^*)}{\operatorname{argmax}} \operatorname{corr}(f_1(\mathbf{X}, \theta_1), f_2(\mathbf{Y}, \theta_2)) \tag{2.20}$$

(2.20) formulates the optimization problem behind DCCA where the parameters θ_1^* and θ_2^* are the weights and biases for the respective NNs, to be determined by means of algorithms such as gradient descent that follow the gradient of the correlation objective [Andrew et al., 2013]. Matrices $\mathbf{X}$,$\mathbf{Y}$ are the datasets defined in (2.7) and (2.8) respectively.

3

Dataset

The dataset used in this study is presented in [Alickovic et al., 2023], which was collected at Oticon A/S. The study was conducted in accordance with the *Declaration of Helsinki* and approved by *Ethics committee for the capital region of Denmark* (reference number `H-21065001`). All participants provided informed consent prior to their participation [Alickovic et al., 2023].

3.1 Experimental Design

The dataset comprises recordings from 25 normal-hearing (20 dB HL threshold) native Danish speakers aged $18-40$ years (mean age 29 ± 6 years) who self-reported as free from any neurological diseases [Alickovic et al., 2023]. Due to issues with the measurements encountered during the acquisition process, performed in preparation of [Alickovic et al., 2023], data from 8 patients has been discarded, leaving only 17 out of the 25 subjects to take part in this experiment. Each participant engaged in 32 listening trials, approximately of one minute in length. During these trials, they were exposed to two concurrent auditory stimuli delivered by speakers positioned directly in front of them and at 30° to the left and right. Participants were instructed to focus on one specific audio stream (the *target of attention* or *target*) while ignoring the other (the *masker*). The concurrent stimuli consisted of excerpts from two audiobooks: a biography of Simon Spies narrated by a male speaker, and a story about traveling in the Himalayas narrated by a female speaker. Each trial contained segments from both audiobooks. To ensure a balanced dataset, the gender and spatial location of the target were systematically varied to form four balanced classes, each of 8 trials [Alickovic et al., 2023]:

- **Male target from the left**
- **Male target from the right**
- **Female target from the left**
- **Female target from the right**

For each trial, both the EEG data and the acoustic envelope of the auditory stimuli were recorded at a sampling frequency of 8 kHz and 44.1 kHz respectively, and later downsampled to 100 Hz during post-processing. The audio stimuli were delivered at 70 dB Sound Pressure Level (SPL) while the participants were seated in the center of a sound studio in the absence of meaningful background noise [Alickovic et al., 2023]. Following each trial, participants answered two yes/no questions related to the content of the attended speech to assess their comprehension during the task.

The dataset includes 330 audio segments, 66 of which have been used during trials and are associated with EEG recordings from one or more patients. The remaining 264 segments have instead been used to separately estimate the principal components of Whisper's hidden states in order to avoid introducing bias.

3.2 Data Preprocessing

As detailed in [Alickovic et al., 2023], the EEG data were collected using a BioSemi ActiveTwo amplification system equipped with a standard 64-electrode cap configured according to the international 10-20 system. Gel was applied to achieve low impedance and maintain each electrode's offset within ± 50 mV. The 64 EEG channels were re-referenced to the average of two additional channels placed on the mastoids. The recorded data were digitally band-pass filtered within the 1 Hz to 10 Hz range, resampled to 50 Hz, and synchronized with the envelope data. The final data segments used for analysis were trimmed to 59 s in length.

4

Methodology

This chapter details the methodologies we used to design novel solutions to AAD. We focused on *forward modeling* using TRFs and *hybrid modeling* with CCA-based methods to model EEG responses to speech stimuli, assessing their ability to distinguish the *target of attention* in a *cocktail party* scenario. Our methodological choices stem primarily from the findings in [Geirnaert et al., 2021; Anderson et al., 2023] and our own intuition on the potential benefits of integrating Whisper's promising capabilities with state-of-the-art AAD techniques such as CCA.

4.1 Feature Extraction

Forward and hybrid modeling approaches depend on the correlation of measured EEG data with representations derived from a set of reference (or *input*) signals. In this study, we explore three approaches to extract relevant information from input speech stimuli and their impact on the discriminatory power of existing modeling techniques (such as TRF and CCA):

- **Acoustic Features** (See Section 4.1.1)
- **Lexical Surprisal** (See Section 4.1.2)
- **Linguistic Embeddings** (See Section 4.1.3)

However, while acoustic features are already in a form suitable for direct use in TRF and CCA models, the same does not apply to linguistic embeddings and lexical surprisal (the latter of which even lacks a temporal dimension). Therefore, this data must be manipulated into appropriate time-resolved signals before it can be used to model brain responses.

Experiments Our research necessitates comparing the results of identical (or similar) procedures applied to different combinations of *features* and *sources*. Each such instance is considered an *experiment*, organized hierarchically according to

features such as the combination of stimuli used or the specific Whisper layer used to generate the embeddings (if applicable). A particularly significant distinction arises from the *source* of the data, indicating which speech stream generated it. Throughout this work, we investigate three possible cases:

- **Target:** features from the *attended* speaker
- **Masker:** features from the *ignored* speaker
- **Foreground:** features from both the *attended* and *ignored* speakers

Note that in the *foreground* case, features from the attended and ignored speakers are both available and separately accessible.

4.1.1 Acoustic Features

Acoustic features have been extensively used in prior research on this and related topics. Their ease of acquisition and processing, computational efficiency, evident correlation with neurological markers of speech processing, and their longstanding availability have established them as a baseline for neurologically aware hearing applications. To evaluate Whisper's contribution to AAD, we employed two types of acoustic features: *envelope* and *onsets*.

Acoustic Envelope The acoustic envelope represents the variation in amplitude of an audio signal over time, providing a measure of *loudness* and its temporal changes. It is computed by dividing the audio into windows of t_e seconds and calculating the *root mean square* of the signal within each window. This process results in a new signal with a frequency of $f_e = 1/t_e$, which can optionally be down-sampled for smoother representation and to avoid issues when computing the *acoustic onsets*. Typically, the computed envelope is then *compressed* by raising it to a power $p \leq 1$ to better model human loudness perception [Alickovic et al., 2023]. In this work, we used $p = 1$ (no compression).

Acoustic Onsets The acoustic onsets are obtained through half-wave rectification of the first derivative of the acoustic envelope. Mathematically, this is expressed as:

$$o(n) = \max\left\{0, \left(\frac{d}{dt}e(t)\right)(n)\right\}, \quad \text{where} \begin{cases} o(n) \text{ is the } n\text{-th onset sample} \\ e(n) \text{ is the } n\text{-th envelope sample} \end{cases}$$

4.1.2 Lexical Surprisals

Surprisals at various levels of speech — ranging from *phonetic surprisals* to *lexical and semantic surprisals* [Heilbron et al., 2022] — have been demonstrated to enhance accuracy of many AAD algorithms [Heilbron et al., 2022; Anderson et al., 2023]. Particularly pertinent to our research are the findings of [Anderson et al., 2023], which reveal a complementary relationship between GPT-2 based

surprisals and earlier layers of Whisper, with the influence of the former on the overall prediction diminishing when paired with deeper layers of the ASR model. Due to the absence of pre-computed accurate transcriptions, combined with our aim to test solutions applicable to real-world implementations, we evaluated Whisper's capability to automatically extract word alignment information from the audio stream to dynamically generate a lexical surprisal signal. For this purpose, we employed *Whisper Timestamped* [Louradour, 2023], which applies Dynamic Time-Warping (DTW) [Giorgino, 2009] to the base Whisper model to produce word-level timestamps and confidence scores over the model's predictions. While the base Whisper model provided by OpenAI is capable of offering time-alignment information, the special tokens are injected into the output stream at unpredictable intervals that do not necessarily match word or period boundaries. To achieve reliable transcriptions, we used the large variant of the model, which has a size of 1.55B parameters and requires powerful hardware and significant computational time to process. Although this setup is not practical for use in hearing aid devices, exploring the potential benefits of including surprisal information can guide future research efforts towards developing miniaturized models capable of reliable, time-resolved generation of surprisals. The lack of suitable models for the automatic generation of period-based surprisals and the focus of this study on the benefits of high-order linguistic information for AAD led us to not implement *semantic* [Heilbron et al., 2022] and *phonetic* surprisals respectively. Following the methodology in [Anderson et al., 2023; Heilbron et al., 2022], we used a GPT-2 model fine-tuned on Danish material using CLP-Transfer [Ostendorff and Rehm, 2023] to generate surprisal values.

Definition of Surprisal Surprisal measures how unexpected the presence of an element in a sequence is, given the context provided by its preceding members. In the case of *lexical surprisal*, we model a sentence — or more generally a text — as a sequence $W = [w_1, w_2, \ldots, w_n]$ of words, with the surprisal associated with the n-th word w_n being its probability $p(w_n|w_1, w_2, \ldots, w_{n-1})$ conditioned on the previously encountered words. To automatically and systematically produce surprisal features, we leverage the output of a Large Language Model (LLM) like GPT-2 which, after the necessary post-processing on the tokens produced by the transformer, yields an array representing the likelihood of each word in the model's dictionary to be the next in the sequence. Similarly to [Anderson et al., 2023; Heilbron et al., 2022; Tezcan et al., 2023; Zhang et al., 2023], we use the *negative-log likelihood* as a measure of lexical surprisal $s(w)$:

$$s(w_n) = -\log\left(p(w_n|w_1, w_2, \ldots, w_{n-1})\right)$$

Figure 4.1 illustrates the functional relationship between $s(w)$ and $p(w)$. As shown, the surprisal value reaches 0 when the conditional probability of the word approaches 1, indicating certainty about w being the next word. Conversely, it ap-

proaches infinity as the probability approaches 0, indicating certainty about *w not* being the next word. To avoid numerical and computational issues, we cap the surprisal value at 100.

Composite Words LLMs do not directly predict a sequence of words but rather a sequence of *tokens*, which are then converted into words using a dictionary [Radford et al., 2019] during the output processing step. As a result, infrequent composite words tend to be split into multiple tokens. For example, the word *læsevane* (translating roughly to *reading habit*) is tokenized into *læse* (*read*) and *vane* (*habit*). In these cases, we consider the conditional probability of the composite word to be the probability of all its components appearing at their respective positions in the sequence. For a composite word $w_n = [w_{n,1}, w_{n,2}, \ldots, w_{n,m}]$, our calculations are as follows:

$$\begin{aligned} s(w_n) &= -\log(p(w_n)) = -\log(p(w_{n,1})p(w_{n,2})\ldots p(w_{n,m})) \\ &= -\left[\log(p(w_{n,1})) + \log(p(w_{n,2})) + \cdots + \log(p(w_{n,m}))\right] \\ &= s(w_{n,1}) + s(w_{n,2}) + \cdots + s(w_{n,m}) \end{aligned}$$

Therefore, for composite words, we sum the surprisals of their components to obtain a single value for our signal.

Signal Processing We generate a *train of Dirac's deltas*, each positioned at a word's onset and proportional to its surprisal, following a methodology similar to [Anderson et al., 2023; Heilbron et al., 2022; Tezcan et al., 2023]. This signal is then processed through the same pipeline used for other signals (see Section 4.2), converting the abrupt surprisal spikes into smooth responses. This process ensures effective correlation with other signals and aligns the signal more closely with the expected shape of the surprisal-related EEG response.

4.1.3 Linguistic Embeddings

Whisper performs a *graded transformation of speech into language* across its layers, with the resulting hidden states becoming increasingly contextualized and linguistic as depth increases [Anderson et al., 2023]. This characteristic has the potential to be highly advantageous for AAD applications, as it is generally accepted that speech processing in the brain occurs at a more superficial level for unattended speech, which should manifest as a growing disparity between the EEG-predictive power of Whisper for attended and ignored speech in progressively deeper layers. In line with [Anderson et al., 2023] and considering Whisper's training objectives, we hypothesize that the encoder produces a linguistic embedding, which the decoder subsequently transforms into words. Given that Whisper can translate 99 languages into English as well as transcribing them [Radford et al., 2022], these embeddings must, according to [Anderson et al., 2023], *suppress within and between speaker*

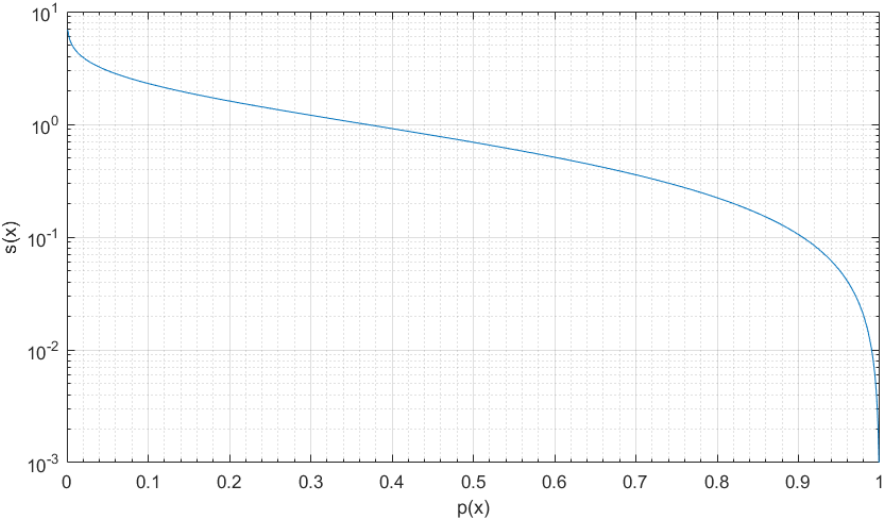


Figure 4.1 Plot illustrating the relationship between the conditional probability $p(w)$ and the surprisal $s(w)$ of a word.

variation in intonation, accents, volume, and intonation to encode word identity in a language-invariant and possibly semantic format. Due to our focus on language processing, this study concentrates on Whisper's encoder and omits its decoder.

Structure of Whisper As per [Radford et al., 2022], Whisper employs a multi-layer encoder-decoder transformer architecture that operates on 30 s long 16000 Hz audio segments. The model's input pre-processor computes an 80-channel log-magnitude Mel spectrogram (using 25 ms windows and 10 ms stride) of the audio, which is then further processed by a small Convolutional Neural Network (CNN) before being fed to the transformer. The output of each encoder layer, including the pre-processor, consists of a 1500-element long time series of 512-dimensional *embeddings*. Given the length of the input signal, these can be interpreted as a family of 30 s long, 50 Hz, 512-dimensional continuous[1] *linguistic embeddings*. Consistent with the notation used in [Anderson et al., 2023], we refer to these signals as Layer 0 through Layer 6 (or L0 through L6 for short), where Layer 0 is the output of the pre-processor (after the convolution), and the rest are the outputs of the corresponding transformer-encoder layers.

[1] The *continuous* nature of the signal is to be interpreted as the fact that these embeddings occur independently of word or phonetic cadence, are not event-driven, and represent the sampling of a *continuous linguistic signal*.

Sliding Window Approach Transformer architectures are popular for their computational efficiency, achieved by processing input data in parallel and using *attention* [Vaswani et al., 2017] to model the relationships between elements of a sequence. However, this parallelism poses a problem when modeling *causal* processes, as the transformer has access to future information. When the transformer computes the output corresponding to $t = 0$ s, it has access to information up to $t = 30$ s, which is unrealistic in a causal setting such as speech and natural language processing in the brain. To address this issue, we adopt the *sliding window* approach used in [Anderson et al., 2023], wherein Whisper's input window is moved forward through the audio in small increments of n_s samples. At each iteration, only the last n_s samples of the output are retained and appended to data from previous iterations, and this process is repeated until the entire waveform has been processed. Initially, only the last n_s samples contain data (the rest of the input is filled with zeros), and this portion expands by n_s samples at each iteration until it reaches a maximum size of $w_s \leq 30$ s. This ensures that at any point, the maximum foresight of the model is limited to $(n_s - 1)/f_s$ seconds, where $f_s = 16$ kHz is the audio sampling frequency. Setting $n_s = 1$ would make the model perfectly causal, but the computational cost would be prohibitive. Inspired by [Anderson et al., 2023], we used $n_s = 2000$ during our experiments, corresponding to a maximum of 0.125 s of foresight. The context window $W(n)$ available to Whisper when calculating the output for the n-th sample is represented mathematically as:

$$W(n) = [\max\{0, w_e - w_s\}, w_e] \quad \text{where} \quad w_e = \lceil n/n_s \rceil$$

Based on [Anderson et al., 2023], we initially set our maximum window size w_s to 10 s but also experimented with $w_s = 20$ s and $w_s = 30$ s. Figure 4.2 illustrates the complete audio feeding process end-to-end.

Dimensionality Reduction The linguistic embeddings are 512-dimensional, making computations based on Whisper's hidden states prohibitively expensive. Therefore, we implemented PCA to reduce the dimensionality of Whisper-based feature vectors while preserving as much information on the original data as possible. PCA identifies the *principal components* — that is, the *directions* in the original vector space along which the most variance is observed — and projects the data along those new axes (See Section 2.3.4).

To avoid introducing bias in our results, we performed principal component analysis on Whisper data generated from the 264 audio segments present in our dataset but not used in the available set of trials. Based on the results of [Anderson et al., 2023], we opted for a dimensionality reduction to 10 components. We also experimented with 64 components to match the number of available EEG channels and not limit the number of canonical correlates during the latter part of the experiment. However, preliminary results combined with the extreme computational requirements of 64-

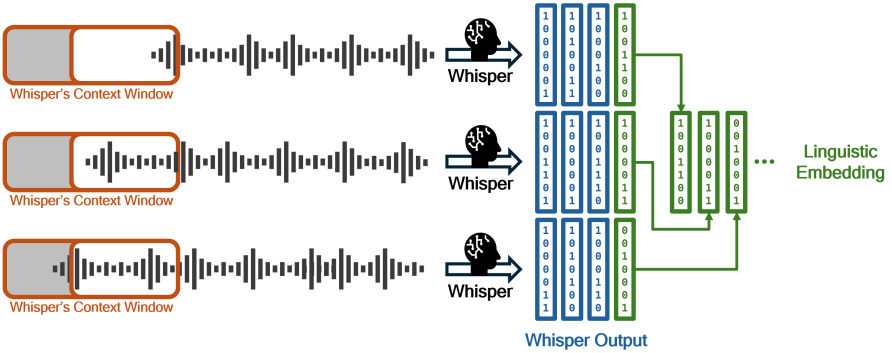


Figure 4.2 Illustration of the sliding window approach used to feed an unlimited length of data to Whisper while maintaining a reasonable level of causality in the predicted token sequence. At each step, the audio is slid in Whisper's context window (artificially limited to the desired length) in strides of fixed length, and a time-equivalent number of samples is selected at the end of Whisper's output. These output samples are then concatenated to form a continuous linguistic embedding.

dimensional stimuli led us to abandon the analysis of higher-dimensional linguistic embeddings. PCA was not performed on EEG data in order to preserve spatial information - and the often significant correlation between close EEG channels - as well as to avoid accidental degradation of the signal, as the typically high noise level of EEG recordings is responsible for a significant amount of the signal variance that PCA seeks to maximize, leading to the accidental amplification and extraction of noise and non-brain artifacts such as eye blinks [Artoni et al., 2018].

4.2 Processing Pipeline

All the aforementioned signals are collected and time-aligned among themselves and with the EEG channels. The signals are then resampled to 32 Hz (the highest power-of-two frequency[2] below the lowest sample rate of our data, which is 50 Hz for the linguistic embeddings), and a high-pass filter of 1 Hz is applied to remove any static components and slow drifts. The signals are then trimmed to the most conservative time range to ensure that we have valid data from all channels at any point during the experiment. Our processing is minimal as our work is based on the same data used in [Alickovic et al., 2023], which already performed part of this preprocessing: "*The available EEG data has already been digitally re-referenced to the average of the mastoid electrodes, digitally band-pass filtered between* 0.1 Hz *and* 10 Hz*, re-sampled to* 100 Hz*, and artifacts were removed based on an indepen-*

[2] Working with powers of two in the frequency domain is desirable as it simplifies FFT and the other arithmetical computations that depend on it.

dent component analysis. Not all data from all subjects was used in this work since some subjects and trials had to be excluded due to bad signal quality or incomplete processing. [...] *For our analysis, the EEG signal per trial and channel was digitally filtered between* 1 Hz *and* 10 Hz*, re-sampled to* 50 Hz*, and normalized to zero mean and unit variance.*"

4.3 TRF Generation

To generate a TRF from a set of source signals, we utilized the MNE software [Gramfort et al., 2014], specifically its `ReceptiveField` framework, which enables the fitting of encoding (source to brain) or decoding (brain to source) models using time-lagged input features. The TRF is calculated based on time delays sampled at a frequency of 32 Hz in the -100 ms to 750 ms range and fitted using delay-aware *ridge regression*. Regularization is incorporated into the model as an explicit L2 regularization factor α, optimally chosen from powers of ten in the 10^4 to 10^{10} range.

4.3.1 Performance Metric

Our primary performance metric for TRF fitness throughout the experiment is the Pearson correlation coefficient between the recorded EEG data and its reconstruction computed from a set of *source signals* (linguistic embeddings, semantic surprisal, acoustic features, etc.). This coefficient is computed for each *experiment* (i.e., combination of *source signals*) and for each of the 64 EEG channels. To gain insight into the individual contributions of different *source signals* to the prediction accuracy, multiple *experiments* have been conducted with different sets of sources to allow for comparative studies.

4.3.2 Model Tuning and Evaluation

The TRFs have been fitted within a nested k-fold cross-validation scheme. The inner loop ($k = 4$) provides validation scores used to select the optimal L2 regularization factor α on a per-patient, per-experiment basis. The outer loop ($k = 8$) calculates a test score on left-out data using the optimal value of α found in the inner loop, providing a more rigorous performance evaluation. A final model is then trained on the whole data (once again per-patient and per-experiment) using the value of α most frequently selected as ideal. This training procedure has been devised to leverage the entire dataset for training, while preventing test scores and data from influencing hyperparameter selection and thereby avoiding data overfitting issues, resulting in strongly validated results for our forward models. To perform AAD, instead, we directly used the final models, as opposed to training and cross-validating new ones. This constitutes a compromise we had to accept due to time constraints, which prevented us from repeating the time consuming TRF training cycle.

4.4 Canonical Correlation Analysis and Classification

The classifier for the analysis has been independently trained for each patient within a 32-fold cross-validation framework, while the CCA transformation matrices $\mathbf{W}_x$ and $\mathbf{W}_y$ - defined in (2.9) and (2.10) respectively - have been estimated on the whole data of each patient to keep the training procedures of TRF and CCA models aligned. In each fold, one of the 32 trials was left out for validation, while the remaining 31 were used for training. This patient-specific training approach is justified by the statistically significant inter-individual variability in EEG responses, which leads to a decline in performance when using universal decoders compared to subject-specific decoders [Geirnaert et al., 2021; O'Sullivan et al., 2014]. Various sets of time delays were tested, characterized by different *start*,*step*,*stop* triplets governing the minimum and maximum time delays as well as the intervals between them. Additionally, an overall delay L was incorporated to model the average stimuli-to-EEG latency, allowing the bank of delays to be centered on the most active time region of the model, a technique also used in previous studies [Cheveigné et al., 2018]. The model's performance was evaluated based on the mean classification accuracy for different numbers of canonical correlates across all 17 patients in the dataset. Drawing inspiration from the discussion in [Alickovic et al., 2019], two methods for handling multiple trials were compared:

- **Multi-View CCA (MCCA)**: CCA filters are independently estimated for each of the 32 trials and then averaged to obtain a single estimator.
- **Trial Concatenation (CCA)**: Trials are concatenated along the time axis, with optional post-processing at the junctions to mitigate edge effects.

Unlike with TRF, we did not train the CCA models on *masker* or *foreground* (*masker* + *target*) data. This is because CCA, as a correlation-maximizing optimizer, would just amplify noise and spurious correlations when presented with ignored or partially ignored speech.

MCCA The first approach, MCCA, was selected based on the hypothesis that independently estimating filters would be beneficial given the varied stimuli across trials, and that averaging the filters would aid in regularization. This method was implemented and evaluated using Matlab and the NoiseTools toolbox [Cheveigné et al., 2018], and the results were compared to those obtained using trial concatenation. The rationale for employing MCCA for AAD is supported by discussions in [Cheveigné et al., 2019; Cheveigné et al., 2018].

Deep CCA Although we planned to include Deep CCA [Andrew et al., 2013] in our analysis since the inception of this study, its role was revised following preliminary results from AAD based on standard linear CCA. The decision was made

to use this advanced correlation technique to improve the generalization of CCA-based AAD implementations by training a generic algorithm to work effectively across patients. The Matlab implementation of Deep CCA provided by [Wang et al., 2016] was used to generalize the aforementioned experiments to a cross-patient model that does not require individual fine-tuning.

4.4.1 Classification and Parameter Selection

To classify audio segments as attended or not, we adopted a *match-mismatch* classification scheme similar to previous work in the field [Cheveigné et al., 2018; Alickovic et al., 2019]. The classifier is simultaneously presented with information from both audio streams and must determine which one is the *target of attention*. Classification is based on the Pearson correlation coefficients between the corresponding *canonical correlates* of each stimulus and the EEG data. Two methods for constructing the feature vector used for classification were tested:

- **Classification of the Union**: Features from both audio streams are presented side-by-side to the classifier.
- **Classification of the Difference**: The element-wise difference between the features of the two audio streams is presented to the classifier.

The first method results in $2n$ classification features, while the second uses n-dimensional vectors, where n is the selected number of *canonical correlates*. Although the CCA is trained on entire 59 s data segments, the correlation features presented to the classifier are computed over smaller windows of length w, a design parameter that affects the MESD of the solution. Various values of w were tested to compare against results in [Alickovic et al., 2019]. The classifier was chosen to be a SVM, based on results from [Alickovic et al., 2019], preliminary findings of our study, and the project's aim to optimize the reliability and notably the *efficiency* of AAD algorithms within real-world constraints.

Parameter Search To identify the optimal set of parameters for each task, a manual optimization process was conducted based on the model's validation scores across all available target data from all patients. Parameters were selected separately for different scenarios (acoustic features vs. Whisper, 64 electrodes vs. 6 electrodes, etc.), but the same parameters were used across all Whisper layers within each scenario for consistency.

4.5 Statistical Validation

To ensure the statistical significance of the performance figures obtained through experimentation, a range of tests were conducted. For validating the correlation figures derived from the fitted TRFs, Wilcoxon's signed-rank test [Wilcoxon, 1945]

was employed. This choice was influenced by the unsuitability of Student's t-test [Student, 1908] due to the non-normal distribution of our correlation metrics (see Figure 4.3) and unequal variances across the classes considered (both of which are prerequisites for the t-test), as well as the desire to compare our results with those of [Anderson et al., 2023].

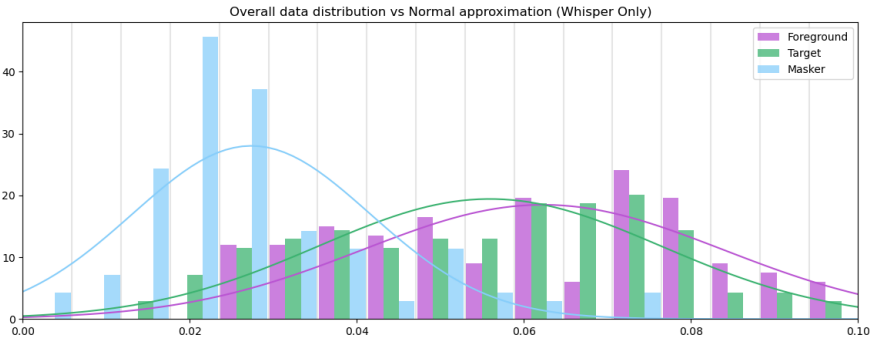


Figure 4.3 Example of the distribution of correlation metrics across the three experimental classes (target, masker, foreground) across all patients for a Whisper-only TRF predictor. Note how the values deviate from the normal distribution corresponding to the mean and variance of the data.

5

Results and Discussion

After generating linguistic embeddings and surprisal signals for all the 66 unique acoustic segments used in the experiment, we conducted an analysis to evaluate different combinations of stimuli. This analysis aimed to determine the ability of language-infused predictors to predict and match the measured EEG signals, understand each stimulus's individual contribution to AAD and how they complement each other - if at all - with a focus on low MESD and electrode count. Consequently, our analysis was divided into two main approaches:

- **Forward Modeling**, where we attempt to directly reconstruct the individual EEG channels using TRFs.
- **Hybrid Modeling**, where we utilize CCA to perform a more opaque classification based on a learned transformation of the original data.

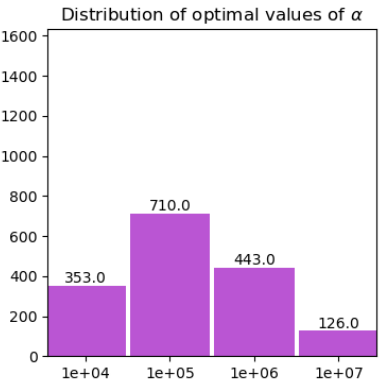
Forward modeling was primarily conducted to validate our results against [Anderson et al., 2023] and to ensure our model adequately captures language processing in the brain. Conversely, the hybrid approach aims to use the potential MESD advantage of CCA, as discussed in [Geirnaert et al., 2021]. To assess the contribution of individual stimuli and compare performance against appropriate baselines, we trained several *ensembles* containing multiple predictors. The performance of these ensembles was then compared with the scores of the individual members to reveal any performance delta, similarly to [Anderson et al., 2023].

5.1 EEG Prediction

In the initial phase of our work, we used the MNE library [Gramfort et al., 2014] to train TRF-based EEG prediction model using *Time-Delayed Ridge Regression* on a per-subject basis. The rationale behind training individual (subject-specific) classifiers is the significant variability in EEG responses to stimuli across subjects [Geirnaert et al., 2021; O'Sullivan et al., 2014], resulting in more accurate predictions when the TRF is trained on data from a single individual as opposed to the

usage of a generalized (subject-independent) model. Our study presents results derived from approximately 32 minutes of data per subject, which we consider both reasonable and practical for customizing a hearing aid solution to individual needs and, therefore, suitable for real-world scenarios.

5.1.1 Hyper-parameter Tuning



As detailed in Chapter 4, a nested cross-validation scheme was used to automatically fine-tune hyperparameters to meet the individual needs of each subject, while also allowing for the collection of unbiased performance metrics without necessitating a rigid data split into training and test subsets. This setup was used to determine the optimal value of α - the regularization parameter of the underlying ridge regression used by the MNE library — for each patient and each combination of inputs and types of stimuli (*target*, *masker*, or *foreground*). A predefined range of α values was tested for each experimental run, with all powers of ten within that range being used for the parameter search. Preliminary experimentation suggested that the 10^4-10^7 range offered an ideal compromise between coverage and computational complexity, which we consequently adopted for all experiments to ensure consistency. As illustrated in the figure above (demonstrating a TRF fitted with a 10 sec context window with 10 principal components and trained on all data), our window was slightly skewed towards higher regularization parameters, given that overfitting problems were found to be more common and impactful than underfitting issues. For a detailed breakdown of the α distribution, refer to Figure A.1.

5.1.2 Whisper and Acoustic Features

In Figure 5.1, we present the layer-wise breakdown of the Pearson correlation between the TRF-predicted EEG signals and actual measured EEG signals, averaged across the 64 electrodes, trials, and subjects. The correlation between predicted and actual measured EEG for the target speech is significantly higher than for the masker speech, and this gap widens as we move to deeper layers of the model. This general trend aligns with the findings reported in [Anderson et al., 2023]. However, our results exhibit one key difference: while [Anderson et al., 2023] observed roughly equal performance across all layers of Whisper for the *ignored* speech predictor and a significant increase in performance for the *attended* speech predictor with deeper layers, our results show a shallow upward slope in predictive power for the *target*

and a noticeable decrease in correlation for the *masker*. Various factors could account for this discrepancy, including differences in datasets and our increased focus on AAD relative to [Anderson et al., 2023]. Nonetheless, the overall performance pattern and relative performance of *target* and *masker* correlators align between our study and [Anderson et al., 2023]. Another interesting observation is the relatively small performance gap between *target* and *foreground* predictors compared to the larger gap between *target* and *masker*, suggesting that the majority of recorded brain activity is explained by the *target* predictor.

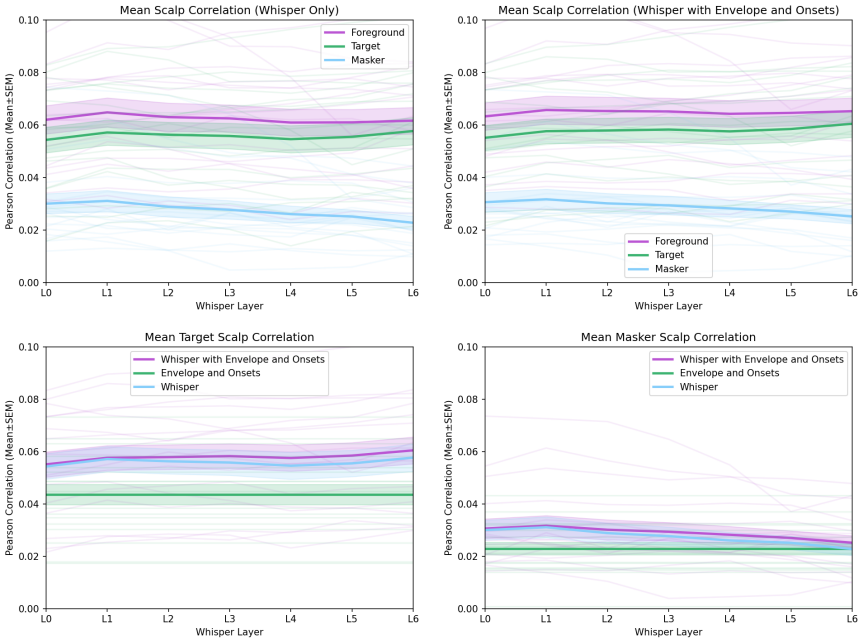


Figure 5.1 Layer-wise scalp-average correlation of the recorded EEG data with predictions computed by various TRF models (*10* s *context window, 10 principal components, trained on all data*), averaged across patients and trials. **Top Left:** Performance using Whisper only. **Top Right:** Performance using an ensemble of Whisper, Acoustic Envelope, and Onsets. **Bottom Left:** Comparison of *target*-based correlation of the ensemble and its components. **Bottom Right:** Comparison of *masker*-based correlation of the ensemble and its components.

In the lower half of Figure 5.1, we compare the performance of the ensemble (Whisper combined with the acoustic envelope and onsets) with its individual components. For the *target*-based predictor, we observe a positive correlation trend as we move through the deeper layers of the network, with the ensemble slightly outperforming the Whisper-only model in deeper layers. This observation aligns with the

reflections in [Anderson et al., 2023] regarding Whisper performing a *graded transformation of speech into language*, wherein deeper layers encode higher-level linguistic information and thus are better complemented by low-level acoustic features, while earlier layers, involved in lower-level processing, do not benefit from the addition of explicit acoustic information as much. For the *masker*-based predictors, no significant difference is noted between the ensemble's performance and that of the Whisper-only predictor. Additionally, the performance gap between Whisper-based and purely acoustic reconstructions diminishes further along the network, consistent with the conclusions in [Anderson et al., 2023] about the absence or reduction of linguistic processing for ignored speech. The higher performance of Whisper's earlier layers compared to the acoustic predictor suggests that the model manages to capture the brain's lower-level transformations of received acoustic signals.

5.1.3 Influence of Whisper's context Window

We tried three different values for the length of Whisper's context window: 30 s (the maximum allowed by the model itself), 20 s, and 10 s. These are some of the values tested in [Anderson et al., 2023], and based on its results, we expected to see a drop of about 20% in predictive performance for a 30 s window compared to a 10 s window. However, we were unable to reproduce such results, and as shown in Figure 5.2, the layer-wise correlation figures of the trained and cross-validated models were virtually identical for all three context windows. We did not test window lengths shorter than 10 s. Given the equivalence of the three window sizes, we opted to use a window length of 10 s for further discussion to keep maximal compatibility with [Anderson et al., 2023].

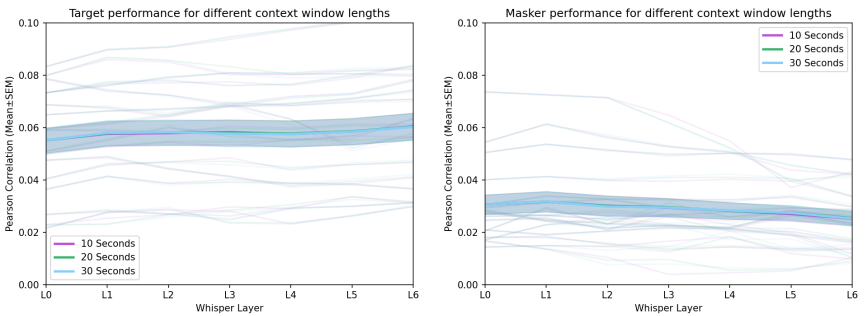


Figure 5.2 Comparison of the layer-wise scalp-average correlation of the recorded EEG data with predictions computed via TRF modeling (10 *principal components, trained on all data*) for different length of Whisper's context window for *target* (left) and *masker* (right) stimuli. The performance is virtually identical for all three tested window lengths.

5.1.4 Influence of Automatically Generated Surprisals

To assess the potential contribution of Whisper-generated syntactic surprisal signals, we compared the performance of a Whisper-only predictor and an ensemble of Whisper and acoustic features, with and without the addition of surprisals. The results are depicted in Figure 5.3. Unfortunately, the surprisal-only predictor signals did not show any meaningful correlation with the recorded EEG data, thus failing to enhance the performance of the Whisper-based predictors (further analysis based on the Whisper and acoustics ensemble can be found in the Appendix). A deeper analysis of the training logs for the surprisal-enabled models revealed that in many instances, surprisal-infused regressors achieved a high correlation with the training signals; in other cases, however, the correlation was either null or even strongly negative. This inconsistency suggested closer analysis of the logs that revealed the Whisper-enabled generation of surprisals using the Whisper Timestamped model [Louradour, 2023] and the current implementation to be too unreliable - both in timing and transcription accuracy - to serve as a robust EEG estimator for AAD. Our dataset lacked a properly formatted authoritative transcription, preventing further analysis using verified timing and surprisal information. A specific challenge encountered was the frequent incorrect grouping (or lack thereof) of composite words, a known issue with tokenized architectures like Whisper [Radford et al., 2022], as complex words often do not map one-to-one with tokens.

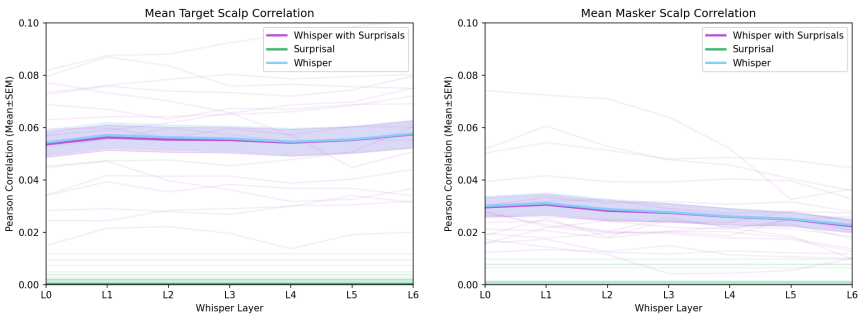


Figure 5.3 Layer-wise scalp-average correlation of the recorded EEG data with predictions computed via TRF modeling (**10** s **context window, 10 principal components, trained on all data**) using Whisper and automatically generated syntactic surprisals. The generated surprisals did not provide any discriminatory power and did not contribute to the accuracy of the model.

5.1.5 Electrode-wise Correlation Analysis

The Topomaps in Figure 5.4, all plotted on the same value scale, illustrate a clear distinction between the *target* (top row) and *masker* (bottom row) correlations. Most

of the correlation is observed in the *frontal* and *parietal* regions, with reduced but still significant correlation in the *temporal* regions, and substantially lower scores in the *occipital* regions. These results align with the findings of [Anderson et al., 2023], including the slightly better correlation scores in the *right temporal* region compared to the *left temporal* one. These encouraging results suggest a significant difference in correlation between *target* and *masker*, even in the temporal regions, which is crucial for the integration of AAD technology in wearable devices.

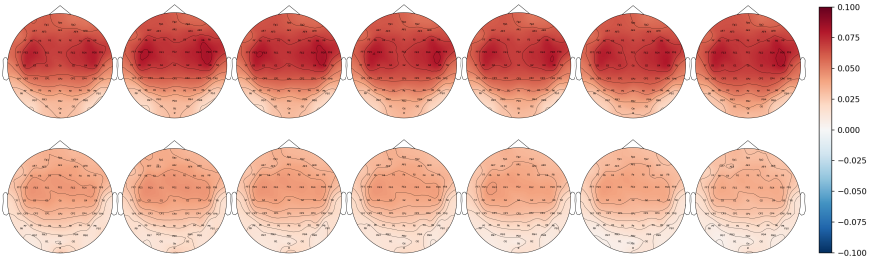


Figure 5.4 Topomaps of the Pearson correlation between the TRF-reconstructed and recorded EEG channels using Whisper with Acoustic Envelope and Onsets (*10s context window, 10 principal components, trained on all data*). **From left to right:** Whisper layers 0 through 6. **Top:** *target* prediction. **Bottom:** *masker* prediction. Whisper mainly predicts activity in the central-parietal region and the two temporal regions. We can also see a stark difference in correlation between the *target* and *masker* audio.

5.1.6 Statistical Analysis

In this section, we analyze the statistical properties of our model's performance across individual patients. Figure 5.1 shows the aggregated behavior of the model, which aligns with our expectations. However, individual datapoints for each patient, represented by faint lines in the background, do not always follow this trend. We observe several instances where the performance of the *foreground* and *target* predictors drops drastically across layers, and the expected downward trend for the *masker* predictor is sometimes absent. To determine whether these exceptions constitute outliers or indicate high instability in our predictions, we studied the correlation between the performance of the regressor (defined as the scalp-average Pearson correlation for each patient and stimulus type - *target*, *masker* or *foreground* - aggregated across the seven Whisper layers through either a minimum, mean or maximum map) and the slope of the first-order least-square estimator of the layer-wise progression of the scalp-average Pearson correlation on a patient-by-patient basis. Additionally, we employed Wilcoxon's signed-ranks test to evaluate the hypothesis that the *target* correlation distribution is significantly higher than the *masker* correlation distribution. The results of this analysis are presented in Figure 5.5. There is a

correlation between slope and performance, indicating a statistically significant difference between the predictive power of the *target* and *masker* EEG estimators. As shown, the variance of the three stimuli classes is significantly different, justifying the use of the signed-ranks test. Overall, the model's most accurate layer (Whisper Layer 6) shows scalp-average Pearson correlations of 0.060 ± 0.005 for the *target* and 0.025 ± 0.003 for the *masker* (Mean $\pm$ Standard Error of the Mean (SEM)).

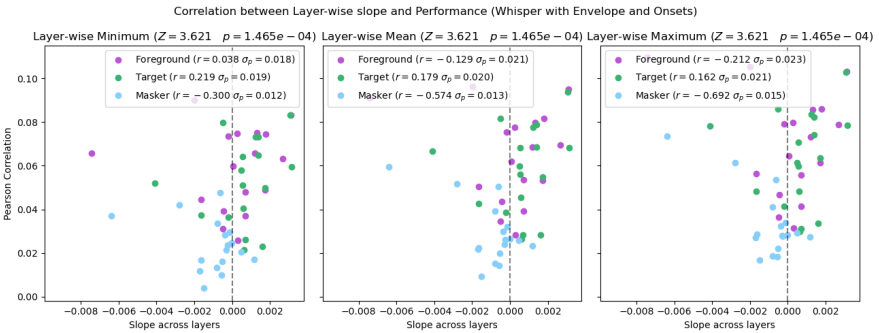


Figure 5.5 Correlation between performance (measured as the per-layer scalp-average between the recorded EEG data and matching TRF reconstruction using 10 s context windows and 10 principal components) and the slope of the first-order polynomial fitted to the layer-wise progression of the scalp-average Pearson correlation for each patient. Performance for each point (corresponding to a patient and stimulus type - *target*, *masker* or *foreground*) is calculated as the aggregated scalp-average Pearson correlation across Whisper's layers, through either a minimum, mean, or maximum map. The Z and p scores are the result of a Wilcoxon's signed-ranks test, where the alternative hypothesis is that the performance figures of the *target* predictor are significantly higher than those of the *masker* predictor. The legend boxes include the correlation r between slope and performance for each stimulus type and the variance σ_p of the performance within the type.

Furthermore, we analyzed the relevance of Whisper's contribution by performing layer-wise statistical tests on the performance difference between Whisper-based target predictions and Whisper-based masker predictions, as well as baseline acoustics. Figure 5.6 illustrates the results of this analysis before and after FDR correction. Both the acoustic and Whisper-based predictors show a significant difference in performance between the *target* and *masker* stimuli. The Whisper-only predictor shows a significant difference with respect to the acoustic predictor only in the first two layers when regressing *masker* stimuli, supporting the hypothesis that later layers of Whisper do not provide a predictive advantage over purely acoustic regression. Additionally, the Whisper with Acoustics ensemble shows a significant performance difference compared to the baseline acoustic predictor up to Whisper Layer 5. This indicates that the acoustic features complement the deeper layers of

Whisper and uniquely contribute to EEG estimation, as discussed in [Anderson et al., 2023].

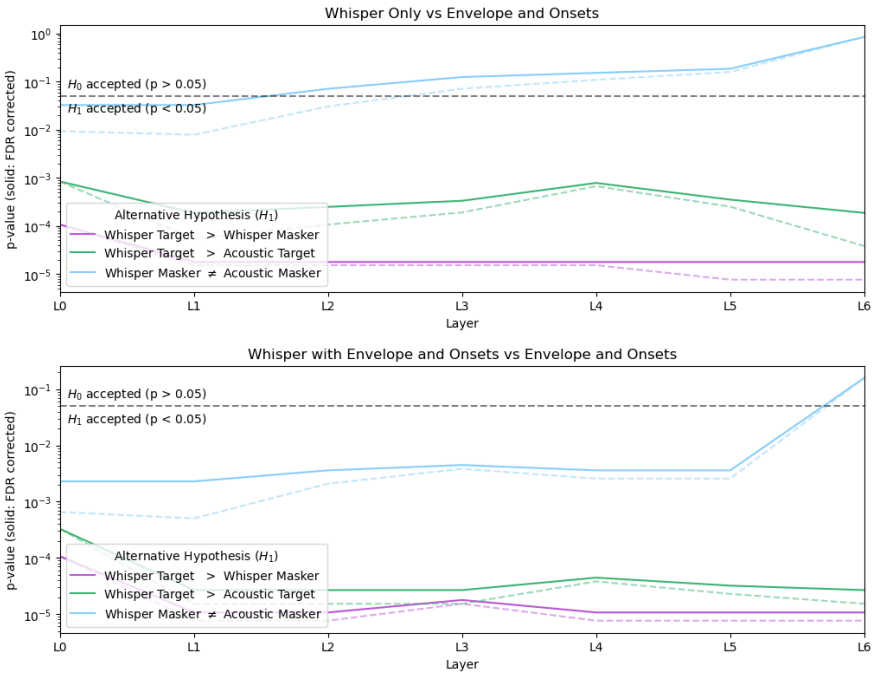


Figure 5.6 p-values of a series of signed-ranks tests performed to verify the significance of the contributions of each element of the Whisper + Acoustics ensemble. Dashed lines represent the raw p-values, whereas the solid lines represent the same values after FDR correction.

5.2 Auditory Attention Decoding

Having validated Whisper's contribution to EEG prediction, we now explore its potential application in near real-time Auditory Attention Decoding (AAD). At its core, AAD is a classification task that aims to determine which among a set of n audio streams is the current *target of attention*. For practical use, an AAD algorithm must be both accurate and fast. High classification accuracy is futile if paired with long response times, as potential users cannot wait extensively for the system to adjust. Conversely, a fast but inaccurate algorithm could be disruptive, as it might consistently amplify undesired sounds while muffling the target of attention.

Therefore, besides classification accuracy, the main performance metric for an AAD algorithm is the MESD, influenced by factors generally categorized into:

- **Operational Delays**, such as processing time for auditory stimuli or general computations.
- **Intrinsic Delays**, which are inherent delays caused by the amount of past data required by the algorithm to predict the target of attention.

5.2.1 TRF-Based Attention Decoding

Initially, we attempted attention decoding through a forward-modeling approach using the models obtained during the training phase. To evaluate the model's performance in AAD, we segmented the available data for each patient into small chunks of fixed length, corresponding to the classifier's *decision window* — the length of data on which the target detection is based. For each context window, we computed the EEG-TRF scalp-average correlation of both stimuli using both the *target* and *masker* classifiers, resulting in a total of four features. These features were used as input to a SVM classifier tasked with identifying whether the first or second stimulus is the target of attention. The classification dataset was carefully constructed to ensure an equal number of cases where the target of attention is the first or second stimulus, and 8-fold cross-validation was employed to ensure the validity of the results, as shown in the left side of Figure 5.7 and Table A.1.

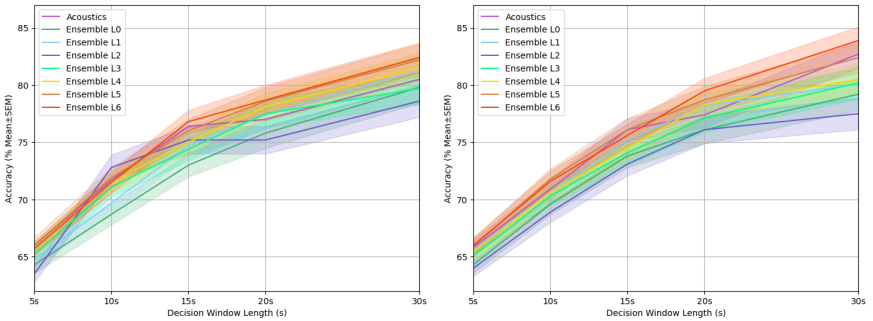


Figure 5.7 Classification accuracy for AAD based on TRF using 64 (left) or 6 (right) EEG channels.

The classifier demonstrates reasonable accuracy even at relatively short context lengths. However, Figure 5.7 shows that Whisper does not significantly enhance performance compared to plain acoustic features when used for attention decoding through TRF reconstruction, despite showing substantial improvements in correlation when fitted to the recorded EEG data using the same method. Furthermore, we notice very small changes in accuracy figures across the Whisper layers for

any given decision window length, perhaps with the exception of the longest 30 s window, seemingly contradicting the previously verified notion that the increased linguistic content of deep Whisper layers provides a significant performance increase over both earlier layers of the ASR model itself as well as acoustic-based predictors. This discrepancy from the favorable results presented in Section 5.1.2 may be attributed to the reduction in context length used to compute the correlation coefficients from the entire 59 seconds of audio previously used down to 30 or even 5 seconds. Whisper's increased language awareness makes its predictions rely on a longer context window compared to plain acoustic features, which primarily correlate with lower levels of acoustic processing requiring less context. This longer information period might make Whisper's internal state more stable and accurate over extended periods but less locally correlated to the EEG, causing the correlation coefficient to drop for short decision windows. This is reflected in Table A.1, where the gap between acoustic and Whisper-based predictors grows with the decision window length.

There are other possible explanations for the unexpected performance. TRF-based predictions use the scalp-average correlation of both audios, which compresses the information embedded in the correlation patterns across the scalp into a single number. However, using the entire set of electrodes for classification would drastically increase the dimensionality of the classification vector, challenging the capabilities of a relatively simple classifier like SVM. Additionally, during the linguistic embedding pipeline, Whisper's 512-dimensional hidden states were reduced down to 10 dimensions using PCA. PCA maximizes *cross-component variance* instead of *explanatory power*, which is usually but not always a good proxy. This is the reason why caution should be used when applying PCA to EEG data, and similar issues might arise when applying it to Whisper's latent space, likely subject to noise (patterns uncorrelated with EEG signals). In general, the relatively high-dimensional data we are working with (both on Whisper's and the EEG side) is subject to the *curse of dimensionality*, which complicates working with high-dimensional data in Machine Learning settings.

5.2.2 CCA-Based Attention Decoding

Previous research has highlighted the significant impact that CCA can have on the performance of AAD classifiers [Alickovic et al., 2019; Geirnaert et al., 2021]. Indeed, projecting the data into a latent space that maximizes the correlation between two sets of data is both an intuitive and powerful extension of a correlation-based classification task. In this study, we decoded the target of attention using subject-specific predictors based on three variants of CCA:

- **Multi-View CCA (MCCA)**: Trials are individually fitted and the filters are then combined together to obtain a single predictor.

- **Single-View CCA**: The 32 trials for each patient are concatenated along the time dimension, treating them as a single, long signal.
- **Deep CCA**: Similar to Single-View CCA but backed by Deep Neural Networks.

The results of the CCA-based attention decoding are presented in Figure 5.8 and Table A.2. The application of CCA not only enhances the overall accuracy of every tested model but also significantly increases the performance gap between traditional acoustic-powered predictors and Whisper for Auditory Attention Decoding. Whisper demonstrates impressive performance, even when paired with relatively simple classifiers using short decision windows. This attention decoding pipeline underscores Whisper's potential and highlights its advantage over acoustics-based AAD algorithms.

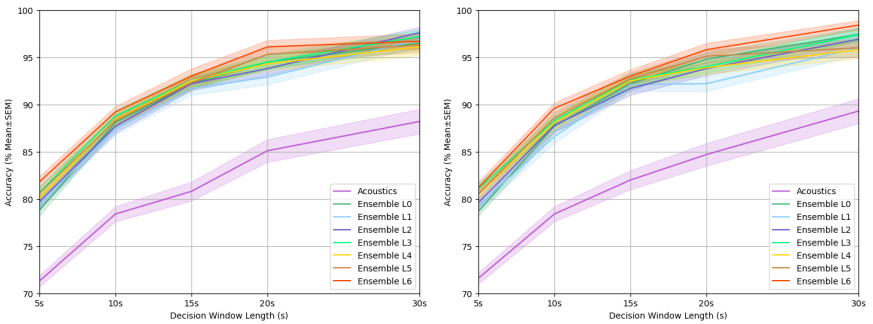


Figure 5.8 Classification accuracy for AAD based on CCA (left) and MCCA (right) using 64 EEG channels. The two approaches produce very similar results, within 0.2% of each other for any decision window length.

Although the performance improvement between classifications using acoustic data and Whisper data is notable, the difference between the different Whisper layers is less pronounced than expected based on the results discussed in Section 5.1. This is particularly true for longer context windows, where performance does not even follow a clear gradient across layers. This could suggest that the hybrid model's performance is being limited in some way, however, the excellent prediction accuracies imply that while increased linguistic content does contribute to prediction accuracy (as seen with shorter decision windows like 5 s, where the layer gradient is more noticeable), there are other intrinsic qualities of the linguistic embeddings driving the substantial performance difference between TRF-based and EEG-based AAD. These qualities remain to be fully explored. Another interesting aspect of these results is the increase in variance as the decision window lengthens, which is counterintuitive. This can be explained by the fact that a longer context window

means fewer windows fit within the same audio segment, leading to more diverse data compared to scenarios with more decisions made on the same file. Note that the results presented were obtained by presenting the classifier with separate features for the two audio streams, as opposed to only their difference (see Section 4.4.1). This classification vector style proved to be significantly more effective than its counterpart.

5.2.3 Reduced Electrode Analysis

In our investigation of the applicability of Whisper-based AAD techniques, we also examined the impact on classification performance when only a few electrodes in the two *temporal regions* are available. This scenario simulates the limited scalp coverage that a pair of smart glasses with embedded EEG probes might realistically achieve. We selected the electrode set `FT7`, `T7`, `TP7` and `FT8`, `T8`, `TP8` (see Figure 2.1). The methods for reducing the data differ between the TRF-based and CCA-based classifiers:

- **For TRF-based AAD**: We used the pre-trained 64-electrode model to compute the correlation coefficients of the reconstructed and recorded EEG signals for each channel. We then selected the coefficients corresponding to the channels of interest and used only those to compute the scalp-average correlation.
- **For CCA-based AAD**: We trained an entirely new model using only the 6 channels of interest on the EEG side.

Our rationale for reusing the more comprehensive TRF models for reduced-electrode analysis was that training would likely occur in a lab with a full electrode cap available. Therefore, it makes sense to train a more complex model using a comprehensive setup to enhance its understanding of the relationship between Whisper and the subset of channels available at runtime. For CCA, it is impossible to "cut out" the unavailable channels from a trained model, as it just specifies a linear combination of EEG channels to complete the transformation into latent space for each canonical correlate. While it is possible to zero out the lost channels or remove the corresponding columns from the EEG-side filters, this approach has experimentally proven to drastically reduce the performance of CCA-based classifiers. This is unsurprising, as the correlation-maximizing computations performed by CCA assume full access to the EEG data and may not transfer well to a reduced set of channels. Thus, for CCA, we opted to train the models from scratch. The results are presented in Figure 5.9 (for CCA and MCCA), the right side of Figure 5.7, and the second halves of Tables A.1 and A.2.

An interesting observation is that the TRF-based classifier does not suffer any noticeable performance hit from the reduction in electrode availability, while the

CCA-based classifier's predictive power drops significantly. This suggests that using the 64-electrode TRF for the 6-electrode classification task helps mitigate the performance penalty inflicted by the substantial reduction in available data. However, it also indicates that the TRF-based classifier (but not the TRF model itself) struggles to extract information from the available Whisper data, as looking at Figure 5.4 we can notice how most of the correlation from the TRF model originates from the central-parietal region, and losing access to that region and more than 90% of the available channels should impose a significant penalty on the results. Despite the reduced accuracy, CCA and Whisper-based AAD still maintain a significant performance gain over the acoustic predictor, indicating that even in constrained environments, Whisper can provide an advantage for AAD algorithms. As we will discuss later, this advantage becomes even greater in terms of MESD when faced with a reduced dataset.

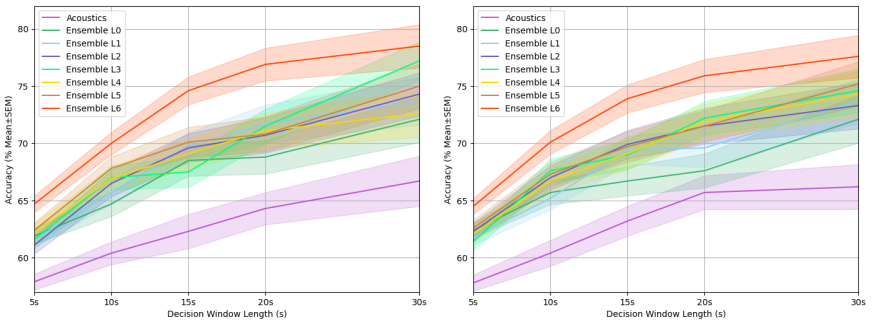


Figure 5.9 Classification accuracy for AAD based on CCA (left) and MCCA (right) using 6 EEG channels. While CCA and MCCA still produce quite similar results, the difference between the two approaches increases when only considering 6 electrodes.

It is important to note that limited electrode count is not the only challenge for AAD-capable wearable technology. Limited computational power, energy availability, and the use of dry electrodes all impact classifier performance, all factors the effect of which cannot be easily predicted without building a prototype and using it to acquire comparable data.

Parameter Selection A comprehensive manual parameter search was conducted to identify the optimal number of canonical correlates, the appropriate set of lags, and the ideal overall time shift L. All models discussed in this work favored the maximum available number of canonical correlates: specifically, 2 correlates for the classifier based on the acoustic envelope and onsets, and 12 correlates for the classifier based on the acoustic envelope, onsets, and Whisper. Interestingly, the

models also preferred an empty set of lags, resulting in a sample-by-sample correlation analysis between EEG and the input stimuli. The only adjustment made to the temporal relationship between the two data streams was through the overall time shift, which varied between -90 ms and -140 ms depending on the specific case. This finding was unexpected, given that previous work on CCA had relied on manipulating acoustic data through a filter bank or set of delays to achieve the reported performance levels [Alickovic et al., 2019; Geirnaert et al., 2021].

5.2.4 Deep CCA

Given the strong performance demonstrated by classical CCA algorithms, for our analysis of the contributions of Deep CCA (DCCA) when paired with Whisper for Auditory Attention Decoding (AAD) we decided to focus on the development of a classifier capable of generalizing well across different subjects. To this end, the 32 trials of all 17 subjects (for a total of 544 instances) were concatenated along the time dimension into a single, long of signal. The signal was then partitioned along the time dimension into 8 equally long contiguous segments, which were then used to perform cross-validation to produce validated performance figures without the necessity for a separate held-out data set.

As a consequence of this cross-validation scheme, each validation set contained data from 4 different trials, guaranteeing the presence in each of significantly different stimuli and, therefore, the correct representation of the expected performance of the model. The results, based on an ensemble of acoustic features and Whisper's Layer 6, are displayed in Table 5.1, while the parameters used for model training are listed in Table A.3. Due to time constraints, an extensive evaluation and tuning of the DNN architecture was not feasible. Consequently, the results presented here are intended to represent the *potential* contribution of DCCA to AAD when paired with Whisper, and to serve as a guide for further, more rigorous research. The collected data reveals that the deep classification pipeline performs reasonably well but still falls short compared to individual implementations - especially in terms of responsiveness - for the 64-electrode AAD. The performance penalty, however, becomes much less significant when considering the 6-electrode scenario. This suggests that the increased modeling capabilities of DCCA over classical CCA algorithms can provide a performance edge, potentially allowing for effective generalization of these classifiers when paired with a more thoroughly optimized architecture. Nevertheless, the inherent complexity of DCCA poses challenges for integration with compact, power-efficient, and resource-constrained devices, given the already substantial performance and training requirements of the relatively simple DCCA networks tested in this study.

Accuracy %	Decision Window Length (s)				
Mean±SD	5	10	15	20	30
64 Electrodes	67.9 ± 0.02	73.4 ± 0.02	77.9 ± 0.02	81.3 ± 0.02	85.7 ± 0.03
6 Electrodes	61.6 ± 0.01	66.7 ± 0.02	69.2 ± 0.03	72.4 ± 0.02	75.6 ± 0.03

Table 5.1 DCCA accuracy for cross-patient AAD with 64 and 6 electrodes.

5.3 MESD Performance

After gathering accuracy performance data at various time windows using different methods, we calculated the MESD for the tested approaches. The results, generated using a confidence level $P_0 = 0.8$, a confidence interval lower bound $c = 0.65$, a minimal number of states $N = 5$, a particle count $K = 1000$ (see Section 2.2.1), and the Matlab MESD Toolbox [Geirnaert et al., 2019a; Geirnaert et al., 2019b; Geirnaert et al., 2020], are reported in Table 5.2. Whisper-based predictions do not provide significant advantages over acoustic features when using TRF reconstruction. In fact, earlier layers produce significantly slower switch times than just using envelope and onsets. However, the combination of Whisper with CCA lowers the MESD by up to 4.5 seconds or 18.5% when using 64 electrodes, whereas the usage of MCCA does not seem to yield any further performance advantage. Deep CCA combined with the deepest Whisper layer successfully generalizes across the entire population of 17 test subjects, achieving a reduction of 3.1 seconds or 12.7% over the best result achieved with acoustic features while being 7.7 seconds or 38.9% slower than a patient-specific AAD pipeline relying on the same Whisper layer and classical CCA. The performance gains of Whisper over envelope and onsets become more apparent when lowering the electrode count from 64 to 6, yielding a reduction of up to 77.3 seconds or 64.9% when using MCCA. In this instance, a larger performance gap between CCA and MCCA is observed, as well as a clearer performance gradient across layers when using CCA-based methods. The generalized DCCA predictor also gains significance with this lower-dimensional EEG representation, with a MESD increase of "just" 8.9 seconds or 21.4% over the per-patient standard CCA equivalent. These results with reduced electrode count, however, are overshadowed by the results obtained with TRF modeling:the TRF approach is able to work with 6 electrodes causing only a marginal increase in MESD, although it exhibits the same issues previously described regarding a lack of performance difference with respect to purely acoustic features. Notably, the Whisper-based ensemble performs similarly, and in many cases *worse*, than the purely acoustic TRF with both electrode counts.

Stimuli Generators	64-Electrode MESD (s)				6-Electrode MESD (s)			
	TRF	CCA	MCCA	DCCA	TRF	CCA	MCCA	DCCA
Env + Ons	36.1	24.3	24.1	-	38.3	132.8	119.2	-
Whisper with Acoustic Envelope and Onsets								
Layer 0	42.2	20.9	20.9	-	42.2	50.0	44.8	-
Layer 1	41.5	20.6	20.7	-	39.6	51.8	50.5	-
Layer 2	39.2	20.5	20.6	-	42.5	54.3	44.9	-
Layer 3	39.4	20.2	20.0	-	40.1	50.2	50.9	-
Layer 4	37.3	20.4	20.2	-	36.2	45.1	46.2	-
Layer 5	35.8	20.2	20.2	-	36.4	44.8	44.5	-
Layer 6	37.3	19.8	20.0	27.5	39.6	41.6	41.9	50.5

Table 5.2 MESD figures for the different AAD pipelines tested in this study. These figures have been generated using the method described in Section 2.2.1, averaging the ESD over 1000 particles

6

Conclusion

In this thesis, we successfully replicated the findings presented in [Anderson et al., 2023], demonstrating a significant increase in correlation between EEG data and forward TRF reconstructions based on Whisper's hidden states. This was compared to predictions obtained through classical sound-based features such as Acoustic Envelope and Acoustic Onsets, using a Danish dataset and achieving results similar to the original paper. This confirms that Whisper-based predictions generalize well across languages, making Whisper an invaluable tool for language-agnostic analysis of EEG data.

Furthermore, we demonstrated that these advantages translate into tangible improvements in Auditory Attention Decoding tasks, with Whisper showing potential for lower MESD when coupled with Canonical Correlation Analysis to enhance the extraction of the underlying correlations between the ASR model and recorded brain data. While developing our AAD pipeline, we discovered that TRF-based attention decoders, unlike CCA-based ones, are unable to effectively capture the linguistic content embedded in Whisper's hidden states. However, they are significantly more resilient to a post-training reduction in electrode data availability than our CCA-based AAD implementation.

Aside from the expected drop in performance when switching from a full-scalp electrode cap to a smaller subset in the temporal regions, CCA failed to fully capture - or at least convert into a tangible accuracy increase - the enhanced linguistic content found in deeper, more linguistic Whisper layers. Indeed, the deep Whisper layers, while performing very well in AAD, showed no significant difference compared to all the layers before them. This may be due to the *curse of dimensionality*, affecting the tractability of our fairly high-dimensional data, especially for simpler mathematical models such as CCA and MCCA. This hypothesis is supported by the fact that Whisper-backed, TRF-based AAD did not significantly improve over acoustic predictors, even with the full 64 electrodes available (Table A.1), which definitely defies our initial expectations when taking on this project. To explore the potential future developments of Whisper and CCA combinations for attention

decoding, we conducted preliminary tests of a Deep CCA architecture. Even with a brief parameter search, our DCCA models achieved competitive performance with acoustic-based classifiers while generalizing across the entire population of 17 test subjects, suggesting potential for further rigorous analysis and not falling much behind other CCA variants in performance over 6-electrode data.

We also experimented with Whisper-generated syntactic surprisal signals [Heilbron et al., 2022] in our predictors, but this approach did not produce a meaningful correlation with EEG data. However, some results discussed in Chapter 5 pertaining to the generation of surprisal-backed TRFs suggest the potential of integrating such information in forward modeling approaches, and possibly hybrid and backward modeling, provided the availability of more reliable surprisal information.

6.1 Future Ramifications of This Work

Future research in Machine Learning-based AAD should further explore the application of Whisper-based forward modeling to AAD, aiming to translate the significant increase in correlation demonstrated in this work and [Anderson et al., 2023] into tangible performance improvements in attention decoding tasks. Tackling the performance of CCA-based AAD with lower electrode counts should also be considered a good candidate for future work on the subject, and further exploration of DCCA-based solutions could prove to be an effective fix for such shortcomings. Rigorous analysis of the contribution of surprisal to attention decoding through manually generated and aligned transcripts is crucial to better understand the potential further contributions of Machine Learning in this field, with the hope of future availability of robust, near-real-time transcription models. AAD has applications across various fields, but one of its most promising developments is in intelligent, neuro-steered hearing aids, which could significantly improve the lives of hearing-impaired patients. It is important to note that our dataset comprises normally hearing subjects, not the intended users of such technology. Research with hearing-impaired subjects has shown drastically reduced correlation, higher variability, and degraded improvements with advanced models, underscoring the need for detailed studies to assess this technology's impact on its intended audience.

6.2 Applicability in Real-World Scenarios

A key concern for implementing this technology is the performance requirements of a large model like Whisper. Wearable devices face constraints in processing power, memory availability, heat generation, volume, power consumption, and energy storage, which are challenging to meet with the high-performance computer components needed to run Whisper today. Although a far cry from the sort of hardware we expect to find in such embedded applications, we did manage audio to linguistic

embedding transformation in less than real-time on a laptop computer (Acer Nitro AN515-57, Intel Core i7-11800H 2.30 GHz, 16 GB DDR4 RAM, RTX 3070 with 8GB of video memory). Future development in hardware technology could also help accelerate the path forward to the integration of these advanced models in implants.

A

Additional Data

Accuracy % Mean±SD	Decision Window Length (s) 5	10	15	20	30
Env + Ons	66.0 ± 0.6	71.7 ± 0.9	76.4 ± 0.9	77.0 ± 1.2	80.5 ± 1.4
Whisper with Acoustic Envelopes and Onsets					
Layer 0	64.3 ± 8.2	68.7 ± 10.5	73.0 ± 11.7	75.8 ± 15.2	79.8 ± 16.3
Layer 1	64.8 ± 8.2	69.7 ± 10.5	74.7 ± 11.7	77.4 ± 14.0	81.3 ± 15.2
Layer 2	63.5 ± 8.2	72.8 ± 12.8	75.2 ± 14.0	75.2 ± 14.0	78.6 ± 16.3
Layer 3	65.2 ± 8.2	71.1 ± 10.5	74.4 ± 11.7	77.5 ± 14.0	79.7 ± 16.3
Layer 4	65.7 ± 7.0	71.5 ± 10.5	75.0 ± 11.7	78.1 ± 14.0	81.5 ± 15.2
Layer 5	66.0 ± 8.2	71.9 ± 10.5	76.0 ± 11.7	78.6 ± 14.0	82.2 ± 16.3
Layer 6	65.7 ± 7.0	71.5 ± 10.5	76.8 ± 11.7	78.7 ± 15.2	82.4 ± 15.2
Accuracy % Mean±SD	Decision Window Length (s) 5	10	15	20	30
Env + Ons	65.8 ± 8.2	70.9 ± 10.5	76.1 ± 11.7	77.4 ± 12.8	82.7 ± 14.0
Whisper with Acoustic Envelopes and Onsets					
Layer 0	64.3 ± 8.2	69.6 ± 10.5	73.8 ± 11.7	76.1 ± 14.0	79.2 ± 16.3
Layer 1	65.6 ± 7.0	70.6 ± 10.5	75.2 ± 11.7	78.2 ± 12.8	80.1 ± 15.2
Layer 2	64.0 ± 8.2	68.9 ± 10.5	73.1 ± 11.7	76.1 ± 14.0	77.5 ± 16.3
Layer 3	65.1 ± 8.2	70.3 ± 9.3	74.1 ± 12.8	77.1 ± 14.0	80.2 ± 16.3
Layer 4	65.4 ± 8.2	70.6 ± 11.7	74.5 ± 11.7	78.3 ± 14.0	80.5 ± 15.2
Layer 5	65.9 ± 8.2	71.8 ± 10.5	76.1 ± 11.7	78.7 ± 14.0	82.4 ± 15.2
Layer 6	66.0 ± 8.2	71.6 ± 10.5	75.6 ± 11.7	79.5 ± 12.8	83.9 ± 14.0

Table A.1 Classification accuracy across all patients for TRF-based AAD (*10s Whisper context length, 10 principal components, trained on all data*) for different combinations of input stimuli and correlation window lengths. **Top:** full electrode coverage. **Bottom:** reduced electrode count.

Accuracy % Mean±SD		Decision Window Length (s) 5	10	15	20	30
Env Ons	MCCA	71.6 ± 7.0	78.4 ± 9.3	82.0 ± 11.7	84.7 ± 14.0	89.3 ± 15.2
	CCA	71.3 ± 7.0	78.4 ± 9.3	80.8 ± 11.7	85.1 ± 14.0	88.2 ± 15.2
Whisper with Acoustic Envelops and Onsets						
L0	MCCA	78.7 ± 5.8	87.7 ± 8.2	92.2 ± 8.2	94.8 ± 8.2	97.4 ± 8.2
	CCA	78.8 ± 7.0	88.1 ± 7.0	92.2 ± 7.0	94.5 ± 8.2	96.5 ± 8.2
L1	MCCA	79.3 ± 7.0	86.6 ± 8.2	92.1 ± 8.2	92.2 ± 9.3	96.0 ± 9.3
	CCA	79.5 ± 7.0	87.4 ± 7.0	91.7 ± 8.2	92.9 ± 9.3	96.9 ± 9.3
L2	MCCA	79.6 ± 5.8	87.8 ± 8.2	91.7 ± 8.2	93.8 ± 8.2	96.9 ± 8.2
	CCA	79.7 ± 7.0	87.7 ± 8.2	92.2 ± 8.2	93.8 ± 8.2	97.6 ± 7.0
L3	MCCA	81.1 ± 7.0	88.3 ± 7.0	92.6 ± 8.2	94.0 ± 8.2	97.4 ± 8.2
	CCA	80.7 ± 7.0	88.8 ± 7.0	92.5 ± 8.2	94.5 ± 9.3	97.2 ± 8.2
L4	MCCA	80.5 ± 7.0	88.1 ± 8.2	92.5 ± 7.0	93.9 ± 8.2	95.8 ± 10.5
	CCA	80.1 ± 7.0	88.4 ± 7.0	92.5 ± 8.2	94.3 ± 8.2	96.1 ± 10.5
L5	MCCA	80.5 ± 5.8	88.6 ± 7.0	92.7 ± 8.2	95.1 ± 8.2	96.0 ± 10.5
	CCA	80.6 ± 7.0	88.3 ± 7.0	92.3 ± 8.2	95.3 ± 8.2	96.3 ± 9.3
L6	MCCA	81.2 ± 7.0	89.6 ± 7.0	93.0 ± 8.2	95.8 ± 8.2	98.4 ± 5.8
	CCA	81.8 ± 7.0	89.2 ± 7.0	93.0 ± 9.3	96.1 ± 8.2	96.7 ± 9.3
Accuracy % Mean±SD		Decision Window Length (s) 5	10	15	20	30
Env Ons	MCCA	57.8 ± 8.3	60.4 ± 13.2	63.2 ± 15.2	65.7 ± 17.2	66.2 ± 22.9
	CCA	57.9 ± 8.1	60.4 ± 11.6	62.3 ± 17.4	64.3 ± 16.4	66.7 ± 25.6
Whisper with Acoustic Envelops and Onsets						
L0	MCCA	62.4 ± 7.9	65.7 ± 12.1	66.7 ± 14.9	67.6 ± 17.4	72.1 ± 24.2
	CCA	61.9 ± 8.0	64.7 ± 12.6	68.5 ± 15.8	68.8 ± 17.2	72.1 ± 23.6
L1	MCCA	61.8 ± 7.9	65.2 ± 11.6	69.4 ± 14.4	69.6 ± 19.7	73.7 ± 22.4
	CCA	61.6 ± 7.8	65.7 ± 11.9	69.7 ± 14.4	71.7 ± 19.4	73.7 ± 24.2
L2	MCCA	62.3 ± 8.6	67.0 ± 12.2	69.9 ± 14.4	71.5 ± 18.3	73.3 ± 23.6
	CCA	61.1 ± 8.7	66.5 ± 13.2	69.6 ± 14.6	70.7 ± 18.0	74.3 ± 22.1
L3	MCCA	61.4 ± 8.5	67.6 ± 12.0	69.0 ± 15.1	72.2 ± 17.1	74.6 ± 22.1
	CCA	61.6 ± 8.9	67.0 ± 11.9	67.5 ± 15.7	71.5 ± 17.5	77.2 ± 19.6
L4	MCCA	62.1 ± 8.3	66.7 ± 12.4	69.1 ± 16.1	71.9 ± 17.7	74.3 ± 23.4
	CCA	62.2 ± 9.0	66.9 ± 12.6	69.1 ± 15.2	71.1 ± 18.0	72.6 ± 24.1
L5	MCCA	62.6 ± 8.4	67.3 ± 12.0	69.7 ± 15.8	71.5 ± 16.0	75.2 ± 22.7
	CCA	62.4 ± 7.8	67.8 ± 12.1	70.1 ± 15.2	70.8 ± 17.7	75.0 ± 22.8
L6	MCCA	64.5 ± 8.1	70.1 ± 12.0	73.9 ± 14.3	75.9 ± 16.9	77.6 ± 21.6
	CCA	64.7 ± 8.5	70.0 ± 11.5	74.6 ± 14.2	76.9 ± 16.7	78.5 ± 22.0

Table A.2 Classification accuracy across all patients for CCA-based AAD (*10s Whisper context length, 10 principal components, trained on all data*) for different combinations of input stimuli, correlation window lengths, and CCA methods. **Top:** full electrode coverage. **Bottom:** reduced electrode count.

Parameter	Value	
	64 Electrodes	6 Electrodes
Stimuli DNN Architecture	[128, 128, 128, 10]	[10, 10]
EEG DNN Architecture	[128, 128, 128, 10]	[64, 10]
Regularization Coefficient	10^{-4}	
Weight Decay	10^{-4}	
Batch Size	1000	
Initial Learning Rate	0.01	
Learning Rate Decay	No Decay	
Momentum	0.99	
Maximum number of epochs	25	

Table A.3 Parameters used to train the DCCA models. Despite the relatively low maximum epoch we haven't observed instances where that turned out to be the limiting factor to learning. Separate architectures have been used to train full-scalp and temporal-only deep correlators. The DNN architectures are represented here as a list of layer widths, excluding the input layers, ordered by increasing depth.

Layer	Scalp-Average Pearson Correlation (Mean±SD)		
	Masker	Target	Foreground
Acoustic Envelope			
-	0.0214 ± 0.0100	0.0411 ± 0.0165	0.0467 ± 0.0148
Acoustic Envelope and Acoustic Onsets			
-	0.0229 ± 0.0102	0.0435 ± 0.0164	0.0490 ± 0.0156
Whisper			
Layer 0	0.0300 ± 0.0161	0.0543 ± 0.0208	0.0620 ± 0.0229
Layer 1	0.0311 ± 0.0171	0.0571 ± 0.0205	0.0648 ± 0.0235
Layer 2	0.0288 ± 0.0165	0.0563 ± 0.0204	0.0630 ± 0.0229
Layer 3	0.0277 ± 0.0149	0.0558 ± 0.0207	0.0625 ± 0.0219
Layer 4	0.0260 ± 0.0134	0.0546 ± 0.0220	0.0609 ± 0.0218
Layer 5	0.0251 ± 0.0110	0.0555 ± 0.0215	0.0610 ± 0.0210
Layer 6	0.0227 ± 0.0104	0.0577 ± 0.0221	0.0616 ± 0.0215
Whisper with Envelope and Onsets			
Layer 0	0.0306 ± 0.0158	0.0551 ± 0.0203	0.0633 ± 0.0223
Layer 1	0.0317 ± 0.0165	0.0576 ± 0.0199	0.0657 ± 0.0228
Layer 2	0.0301 ± 0.0162	0.0579 ± 0.0201	0.0653 ± 0.0226
Layer 3	0.0294 ± 0.0150	0.0583 ± 0.0202	0.0651 ± 0.0217
Layer 4	0.0282 ± 0.0136	0.0576 ± 0.0212	0.0642 ± 0.0215
Layer 5	0.0270 ± 0.0116	0.0585 ± 0.0211	0.0646 ± 0.0208
Layer 6	0.0251 ± 0.0116	0.0605 ± 0.0219	0.0653 ± 0.0216

Table A.4 Scalp-average Pearson correlation with the recorded EEG data of TRF-based predictors fed with different sets of input stimuli.

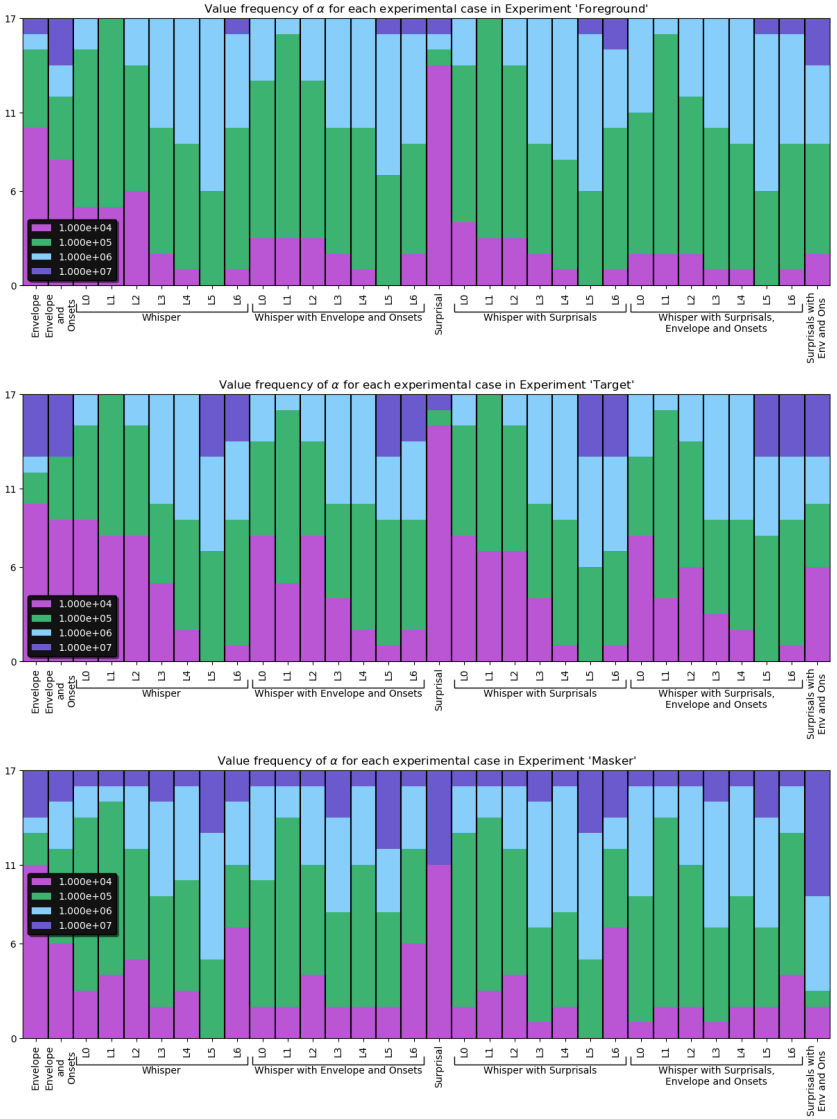


Figure A.1 Distribution of ideal values of α during TRF fitting (*context window of 10s, 10 principal components, all data included*). This range of values has been experimentally chosen as a compromise of good coverage and computational intensity as we were building our TRF pipeline. Ideal α values have been individually chosen for each patient, set of stimuli (and Whisper layer when it was involved), and type of stimuli (*target*, *masker* or *foreground*).

References

Alickovic, E., T. Dorszewski, T. U. Christiansen, K. Eskelund, L. Gizzi, M. A. Skoglund, and D. Wendt (2023). "Predicting eeg responses to attended speech via deep neural networks for speech". In: *2023 45th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pp. 1–4. DOI: 10.1109/EMBC40787.2023.10340027.

Alickovic, E., T. Lunner, F. Gustafsson, and L. Ljung (2019). "A tutorial on auditory attention identification methods". *Frontiers in Neuroscience* **13**. ISSN: 1662-453X. DOI: 10.3389/fnins.2019.00153.

Anderson, A. J., C. Davis, and E. C. Lalor (2023). "Context and attention shape electrophysiological correlates of speech-to-language transformation". *bioRxiv*. DOI: 10.1101/2023.09.24.559177.

Andrew, G., R. Arora, J. Bilmes, and K. Livescu (2013). "Deep canonical correlation analysis". In: *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*. ICML'13. JMLR.org, Atlanta, GA, USA, III–1247–III–1255.

Artoni, F., A. Delorme, and S. Makeig (2018). "Applying dimension reduction to eeg data by principal component analysis reduces the quality of its subsequent independent component decomposition". *NeuroImage* **175**, pp. 176–187. ISSN: 1053-8119. DOI: https://doi.org/10.1016/j.neuroimage.2018.03.016. URL: https://www.sciencedirect.com/science/article/pii/S1053811918302143.

Baevski, A., H. Zhou, A. Mohamed, and M. Auli (2020). *Wav2vec 2.0: a framework for self-supervised learning of speech representations*. arXiv: 2006.11477 [cs.CL].

Brodbeck, C., P. Das, T. L. Brooks, S. Reddigari, and jpkulasingham (2023). *Eelbrain*. Version 0.39. DOI: 10.5281/zenodo.7951251. URL: https://doi.org/10.5281/zenodo.7951251.

Cherry, E. C. (1953). "Some experiments on the recognition of speech, with one and with two ears". *The Journal of the Acoustical Society of America* **25**:5, pp. 975–979. DOI: 10.1121/1.1907229.

Cheveigné, A. D., D. Wong, G. D. Liberto, M. Slaney, and E. Lalor (2018). "Decoding the auditory brain with canonical component analysis". *NeuroImage*. Query date: 2024-05-17 14:15:42. URL: https://www.sciencedirect.com/science/article/pii/S1053811918300338.

Cheveigné, A. de, G. M. D. Liberto, D. Arzounian, D. D. Wong, J. Hjortkjær, S. Fuglsang, and L. C. Parra (2019). "Multiway canonical correlation analysis of brain data". *neuroimage*. Query date: 2024-05-17 14:14:51. URL: https://www.sciencedirect.com/science/article/pii/S1053811918321049.

Crosse, M. J., G. M. Di Liberto, A. Bednar, and E. C. Lalor (2016). "The multivariate temporal response function (mtrf) toolbox: a matlab toolbox for relating neural signals to continuous stimuli". *Frontiers in Human Neuroscience* **10**. ISSN: 1662-5161. DOI: 10.3389/fnhum.2016.00604.

Friston, K. (2005). "A theory of cortical responses". *Philosophical transactions of the Royal Society of London. Series B, Biological sciences* **360**, pp. 815–36. DOI: 10.1098/rstb.2005.1622.

Geirnaert, S., T. Francart, and A. Bertrand (2019a). *MESD toolbox*. Query date: 2024-05-17 21:02:17. GitHub.

Geirnaert, S., T. Francart, and A. Bertrand (2019b). "A new metric to evaluate auditory attention detection performance based on a markov chain". In: *2019 27th European Signal Processing Conference (EUSIPCO)*, pp. 1–5. DOI: 10.23919/EUSIPCO.2019.8903146.

Geirnaert, S., T. Francart, and A. Bertrand (2020). "An interpretable performance metric for auditory attention decoding algorithms in a context of neuro-steered gain control". *IEEE Transactions on Neural Systems and Rehabilitation Engineering* **28**:1, pp. 307–317. DOI: 10.1109/tnsre.2019.2952724.

Geirnaert, S., S. Vandecappelle, E. Alickovic, A. de Cheveigne, E. Lalor, B. T. Meyer, S. Miran, T. Francart, and A. Bertrand (2021). "Electroencephalography-based auditory attention decoding: toward neurosteered hearing devices". *IEEE Signal Processing Magazine* **38**:4, pp. 89–102. ISSN: 1558-0792. DOI: 10.1109/MSP.2021.3075932.

Giorgino, T. (2009). "Computing and visualizing dynamic time warping alignments in r: the dtw package". *Journal of Statistical Software* **31**:7. DOI: 10.18637/jss.v031.i07.

Goodfellow, I., Y. Bengio, and A. Courville (2016). *Deep Learning*. http://www.deeplearningbook.org. MIT Press.

Gramfort, A., M. Luessi, E. Larson, D. A. Engemann, D. Strohmeier, C. Brodbeck, L. Parkkonen, and M. S. Hämäläinen (2014). "Mne software for processing meg and eeg data". *NeuroImage* **86**, pp. 446–460. ISSN: 1053-8119. DOI: https://doi.org/10.1016/j.neuroimage.2013.10.027.

Gundersen, G. (2018). *Canonical correlation analysis in detail*. URL: https://gregorygundersen.com/blog/2018/07/17/cca/.

Hastie, T., R. Tibshirani, and J. Friedman (2009). *The elements of statistical learning: data mining, inference and prediction*. 2nd ed. Springer. URL: http://www-stat.stanford.edu/~tibs/ElemStatLearn/.

Heilbron, M., K. Armeni, J.-M. Schoffelen, P. Hagoort, and F. P. de Lange (2022). "A hierarchy of linguistic predictions during natural language comprehension". *Proceedings of the National Academy of Sciences* **119**:32, e2201968119. DOI: 10.1073/pnas.2201968119. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.2201968119. URL: https://www.pnas.org/doi/abs/10.1073/pnas.2201968119.

LeCun, Y., Y. Bengio, and G. Hinton (2015). "Deep learning". *Nature* **521**, pp. 436–44. DOI: 10.1038/nature14539.

Louradour, J. (2023). *Whisper-timestamped*. https://github.com/linto-ai/whisper-timestamped.

Marrone, N., C. R. Mason, and G. Kidd (2008). "Evaluating the benefit of hearing aids in solving the cocktail party problem". *Trends in Amplification* **12**:4, pp. 300–315. DOI: 10.1177/1084713808325880.

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

O'Sullivan, J. A., A. J. Power, N. Mesgarani, S. Rajaram, J. J. Foxe, B. G. Shinn-Cunningham, M. Slaney, S. A. Shamma, and E. C. Lalor (2014). "Attentional Selection in a Cocktail Party Environment Can Be Decoded from Single-Trial EEG". *Cerebral Cortex* **25**:7, pp. 1697–1706. ISSN: 1047-3211. DOI: 10.1093/cercor/bht355.

Ostendorff, M. and G. Rehm (2023). *Efficient language model training through cross-lingual and progressive transfer learning*. arXiv: 2301.09626 [cs.CL].

Parra, L. C. (2018). "Multi-set canonical correlation analysis simply explained." *arXiv: Machine Learning*. URL: https://api.semanticscholar.org/CorpusID:88517637.

Radford, A., J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever (2022). *Robust speech recognition via large-scale weak supervision*. arXiv: 2212.04356 [eess.AS].

Radford, A., J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever (2019). "Language models are unsupervised multitask learners". In: URL: https://api.semanticscholar.org/CorpusID:160025533.

Salorio-Corbetto, M. and B. Moore (2023). "Hearing aids can't solve the cocktail party problem — yet". *Acoustics Today* **19**, p. 45. DOI: 10.1121/AT.2023.19.2.45.

Shlens, J. (2014). *A tutorial on principal component analysis*. arXiv: 1404.1100 [cs.LG].

Sörnmo, L. and P. Laguna (2006). *Bioelectrical signal processing in cardiac and neurological applications*. Elsevier Academic Press.

Student (1908). "The probable error of a mean". *Biometrika*, pp. 1–25.

Sur, S. and V. Sinha (2009). "Event-related potential: an overview". *Industrial Psychiatry Journal* **18**:1, p. 70. DOI: 10.4103/0972-6748.57865.

Tezcan, F., H. Weissbart, and A. Martin (2023). "A tradeoff between acoustic and linguistic feature encoding in spoken language comprehension". *Elife*. Query date: 2024-05-11 17:18:56. URL: https://elifesciences.org/articles/82386.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin (2017). "Attention is all you need". In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. NIPS'17. Curran Associates Inc., Long Beach, California, USA, pp. 6000–6010. ISBN: 9781510860964.

Wang, W., R. Arora, K. Livescu, and J. A. Bilmes (2016). "On deep multi-view representation learning: objectives and optimization". *CoRR* **abs/1602.01024**. arXiv: 1602.01024.

Wilcoxon, F. (1945). "Individual comparisons by ranking methods". *Biometrics* **1**, pp. 196–202.

Woodman, G. F. (2010). "A brief introduction to the use of event-related potentials in studies of perception and attention". *Attention, Perception, & Psychophysics* **72**:8, pp. 2031–2046. DOI: 10.3758/bf03196680.

Yu, D. and L. Deng (2016). *Automatic speech recognition: A deep learning approach*. Springer.

Zani, A. (2013). "Evoked and event-related potentials". In: pp. 787–793. ISBN: 978-1-4020-8265-8. DOI: 10.1007/978-1-4020-8265-8.

Zhang, X., J. Li, Z. Li, B. Hong, T. Diao, X. Ma, G. Nolte, A. K. Engel, and D. Zhang (2023). "Leading and following: noise differently affects semantic and acoustic processing during naturalistic speech comprehension". *NeuroImage*. Query date: 2024-05-11 17:20:17. URL: https://www.sciencedirect.com/science/article/pii/S1053811923005554.

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

Document name
MASTER'S THESIS

Date of issue
June 2024

Document Number
TFRT-6233

Author(s)
Alessandro Celoria
Valentín López

Supervisor
Emina Alickovic, Eriksholm Research Centre, Sweden
Martin Skoglund, Eriksholm Research Centre, Sweden
Bo Bernhardsson, Dept. of Automatic Control, Lund University, Sweden
Pontus Giselsson, Dept. of Automatic Control, Lund University, Sweden (examiner)

Title and subtitle
An ASR-based Hybrid Approach for Auditory Attention Decoding

Abstract

Auditory Attention Decoding (AAD) aims to determine the focus of a listener's attention in environments with multiple overlapping speakers, a challenging situation for hearing impaired patients known as the Cocktail Party Problem. This thesis investigates AAD using Whisper, a transformer-based Automatic Speech Recognition (ASR) system that performs a graded transformation from speech to text while encoding linguistic and semantic information in its latent encoder layers. Two approaches to AAD are explored: first, a forward pipeline that utilizes Whisper for preprocessing audio stimuli in conjunction with a Temporal Response Function (TRF) model for predicting Electroencephalography (EEG) responses. Second, a hybrid approach aims to enhance the classification performance by applying Canonical Correlation Analysis (CCA) and its neural network variant, Deep Canonical Correlation Analysis (DCCA), to Whisper's latent encoder layers and EEG signals. The performance of these models is compared across fixed decision window lengths, assessing their attention decoding capabilities when presented with limited information, to highlight Whisper's enhanced performance when combined with CCA. Additionally, we test Whisper's AAD performance when only a restricted number of electrodes limited to the temporal regions is available, as a step towards the development of wearable neurosteered hearing aid devices.

Keywords

Classification system and/or index terms (if any)

Supplementary bibliographical information

ISSN and key title
0280-5316

ISBN

Language
English

Number of pages
1-62

Recipient's notes

Security classification