Title
Forecasting the Neural Time Series: Deep Neural Networks for Predicting Event-Related EEG Responses

Author
Ibagon, Gabriel

Publication Date
2018

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA SAN DIEGO

Forecasting the Neural Time Series: Deep Neural Networks for Predicting Event-Related EEG Responses

A Thesis submitted in partial satisfaction of the requirements

for the degree Master of Science

in

Computer Science

by

Gabriel Ibagon

Committee in charge:

Virginia de Sa, Chair

Garrison W. Cottrell

Tim Mullen

Lawrence Saul

2018

The Thesis of Gabriel Ibagon is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

Chair

University of California San Diego

2018

DEDICATION

To my family.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to thank Dr. Tim Mullen, Director of Intheon Labs, for his guidance and mentorship throughout the development of my thesis. I'd also like to thank Nima Bigdely-Shamlo, Christian Kothe and the rest of team at Intheon for their instructive inputs and support in the development of this work. Working with this group has consistently extended the limits of my knowledge and ability, and I look forward to delving deeper into machine learning research in the coming years.

I would also like to thank my graduate adviser, Dr. Virginia de Sa, for her encouragement and guidance in furthering my interest and knowledge of brain-computer interfaces during my time at the University of California San Diego. I'm grateful for stimulating conversations with her and other members of the de Sa Lab.

Chapter 1-5 include material published in the proceedings of the 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC), "Deep Neural Networks for Forecasting Single-Trial Event-Related Neural Activity", Ibagon, Gabriel; Kothe, Christian; Bidgdely-Shamlo, Nima; Mullen, Tim. The thesis author was the primary investigator and author of this material.

This work was supported in part by the Army Research Laboratories and was accomplished under Cooperative Agreement Number W911NF-10-2-0022. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

ABSTRACT OF THE THESIS

Forecasting the Neural Time Series: Deep Neural Networks for Predicting Event-Related EEG Responses

by

Gabriel Ibagon

Master of Science in Computer Science

University of California San Diego, 2018

Virginia de Sa, Chair

In this work, we explore the topic of forecasting the neural time series using machine-learning based techniques on electroencephalography (EEG) data. Forecasting EEG has a number of potential applications in brain-computer interfaces (BCI), such as ahead-of-time event classification, cognitive response prediction, and preemptive intervention therapy. However, previous work in EEG forecasting has failed to accurately predict the time series more than a few steps into the future. Simple linear models lack the capacity to model the high-dimensional dynamics of EEG activity, while complex nonlinear models are difficult to specify and implement. However, recent deep neural networks have effectively modeled high-dimensional systems in a variety of domains. In this work, we hope to bridge the gap between previous work in EEG forecasting and current techniques in deep learning. In particular, we explore forecasting the EEG patterns that occur after the presentation of a time-locked visual stimulus. We implement a deep neural network that extracts features from pre-event data in order to predict single-trial event-locked EEG data. To capture the variation of a single trial, the network constructs the post-event waveform in two parts: 1) generating ongoing neural activity and 2) generating evoked event-related responses. We evaluate our model by forecasting 500 milliseconds of single channel post-event data from a Rapid Serial Visual Presentation (RSVP) task. Our results indicate a significant increase in forecasting performance compared to baseline methods, suggesting that deep neural networks can extract informative features from EEG data in order to generate a prediction of the post-event waveform.

Chapter 1

Introduction

Brain activity underlies the thoughts, behaviors, and experience that define human existence. Neural phenomena is driven by the passage of time as the brain transitions from one state to the next. Thus, the field of time series analysis applied to neural dynamics is an inherent part of understanding the nature of brain activity.

Electrodes placed on the scalp can record the summed electrical activity of various neural processes in the brain. The study of electroencephalography (EEG) involves analyzing these electrophysiological signals as they propagate onto the scalp. The analysis of these recordings can reveal information about an individual's health, behavior, and cognitive state.

Brain-computer interfaces (BCI) introduce an intimate paradigm of human-computer interaction, where computers access information directly from a user's neural signals. Computer systems can utilize this information to perform tasks based on the user's state, behavior, or intention. Many BCIs are EEG-based systems, due to their low cost, non-invasiveness, and mobility. However, decoding user state with EEG suffers from a low signal-to-noise ratio, since scalp potentials have low spatial resolution and are often contaminated with extraneous electromagnetic artifacts. EEG-based BCI performance can be boosted by machine learning algorithms that learn to detect relevant patterns within the EEG signal. Current trends in BCI research involve developing more accurate and sophisticated algorithms for extracting meaningful information about user states and intentions.

One class of machine learning algorithms in BCI research involves forecasting — generating estimates of future data given observations of past data. Forecasting EEG requires analyzing observed data to describe changes in brain dynamics over time. Using machine learning-based forecasting techniques, we can develop a forecasting model by optimizing its performance on a corpus of recorded data, with the goal of generating accurate forecasts on new examples of data. Developing a strong forecasting model gives us access to estimates of future data, as well as valuable insights about the nature of EEG dynamics. The information gained from the forecasting model can then be used to deliver key gains in BCI performance.

1.1 Objective

This work will explore the applications, techniques, and challenges of forecasting EEG data. Then, we will propose a method to address the problem of forecasting EEG data after a time-locked visual stimulus. The conclusion of this work demonstrates that deep neural networks can effectively learn to extract information from observed data in order to accurately forecast EEG data several steps into the future.

EEG data poses a series of challenges to many traditional techniques in time-series forecasting. EEG recordings are known to be high-dimensional, chaotic, and noisy, rendering simple linear models ineffective at forecasting more than a few time steps into the future. On the other hand, complex nonlinear models are often difficult to develop and optimize in practice. However, recent research in deep learning has demonstrated that deep neural networks perform well on a wide variety of tasks, including domains with high-dimensional and complex data.

For this reason, we aim to show that deep learning can be used to train an adequately complex forecasting model of the EEG time series. In particular, we construct a predictive model of single-trial event-related neural responses, in which we use pre-event EEG data to forecast the signal after a time-locked event. The algorithm learns to extract information about ongoing brain dynamics from the pre-event data, in order to estimate how these dynamics will affect the brain's response to an event. By structuring our model around known theories of event-related brain dynamics, the algorithm learns to construct a causal relationship between the observed brain data and a set of characteristics of the single-trial event-related neural response.

1.2 Motivation

Forecasting of EEG has a variety of potential applications in ERP analysis and brain-computer interfaces. In this section, we will outline a variety of applications of forecasting techniques, as well as justifying the benefits of using forecasting to achieve these goals.

1.2.1 Applications

Ahead-of-Time Classification

EEG event-type classification can be achieved using a variety of machine learning models, such as support vector machines, linear discriminant analysis, and deep neural networks [LCL+02][LSW+16]. In these techniques, the model incurs an inherent delay while awaiting the necessary amount of event-related data to be collected before making a classification. This delay can cause issues in time-critical BCI applications. An EEG forecasting model can reduce the amount of data needed to make an event classification by forecasting an estimate of the future waveform. Given a limited amount of post-event EEG data, a forecasting model could generate the completion of the waveform, giving a classification model the sufficient amount of data needed to make a decision.

Denoising

Forecasts can also be used to improve the signal-to-noise ratio of EEG recordings. Regression-based approaches to artifact rejection use autoregressive forecasts to detect artifacts in EEG data based on deviations from the predicted dynamics [NDR09]. This concept can further be applied to ERP analysis, where the evoked portion of the post-event signal can be isolated by subtracting forecasts of the ongoing activity from the post-event neural response [NNS91].

Cognitive/Behavioral Response Prediction

Individual trials of an ERP contain variability in peak latency, amplitude, and morphology. The characteristic of the individual waveform can be used as a predictor for cognitive and behavior factors, such as fatigue, reaction time, and recall. Analyzing the estimated waveform characteristics of the post-event forecast could be used to infer the user's cognitive state. In this sense, a generative model of ERPs could be used to create a regressive system to generate cognitive state predictions based on forecasts of the EEG.

Closed-Loop Intervention

Given the ability to perform ahead-of-time ERP event classification and cognitive response prediction, we can use forecasts to make informed decisions in closed-loop brain computer interfaces. For example, trans-cranial alternating current stimulation (tACS) therapies often require electrical stimulation to be applied in phase with certain oscillatory processes [MDG$^+$17]. If we wish to avoid certain types of cognitive responses, we can use a generative model to predict the anticipated ERP morphology or class to determine if preemptive intervention is necessary.

Medical Diagnostics

The characteristics of the forecast residuals can also be used to classify several cognitive states that relate to medical conditions, such as depression and dementia [ALK^{+}97]. Furthermore, statistical properties of trained nonlinear models, such as the correlation dimension, have been found to be predictors of medical conditions such as epilepsy [SvWK97][BD86].

Neuro-Cognitive Modelling

The statistical properties that can be learned from developing a forecasting model on EEG data can give us valuable insight onto the underlying mechanisms of brain dynamics. Many theoretical models of the brain have emerged from the development of forecasting, attempting to characterize the generators of physiological activity and temporal brain dynamics [BdP04][AAC^{+}18].

1.2.2 Motivation for Forecasting Models

Several of the above applications involve using a forecasting system as an intermediate step in performing a desired task:

Observed Data → Generated Forecast → Task

For example, in ahead-of-time classification, a forecast is generated from the observed data, and then a classifier is subsequently applied to the generated forecast. Hypothetically, this application could be solved by developing a single model that uses the observed data to learn a direct mapping to the task, bypassing the forecasting step:

Observed Data → Task

The benefit of this direct mapping is that it may involve a simpler system setup, where only one model needs to be developed.

However, in practice, generating an intermediate forecast may have a number of benefits in achieving the goal of the system. Supervised machine learning algorithms that perform classification tasks require labeled data, which, in the context of EEG, provides very sparse training samples. The training corpus is often limited to task-specific data, although some transfer learning approaches have addressed this constraint [WLW+15]. The collection of additional training samples to strengthen the model may be expensive or impossible to achieve. On the other hand, a forecasting algorithm can be optimized using unlabelled data, and thus, we can take advantage of a large corpus of EEG data to develop a strong forecasting model. Even while forecasting event-related potentials, the model can still benefit from the analysis of unlabelled, continuous data, as will be shown in our methods.

Additionally, a forecasting model can be developed independently from the target task. A single trained forecasting model can be re-used for a variety of different tasks, where improvements in the forecasting accuracy can simultaneously boost the performance of all tasks.

1.3 Content

Chapter 1 introduces the problem of forecasting the neural time series using EEG data, framing the development and use cases of forecasting techniques. **Chapter 2** frames the theoretical background of the problem, exploring forecasting, deep learning, and the application of forecasting techniques to event-locked EEG data. **Chapters 3** and **4** explain our proposed solution to forecasting single-trial event-related potentials. **Chapter 3** explains the theory and architecture of our method. **Chapter 4** describes an experiment demonstrating the application of this method to a data set of EEG recordings. Finally, **Chapter 5** concludes the work by evaluating our results in the wider context of forecasting neural time series, suggesting key lessons and directions for future work.

Chapter 2

Background

2.1 Forecasting

2.1.1 Introduction

Forecasting is a body of analytic techniques used to make predictions of data from the future based on observations of data from the past. Machine learning-based forecasting techniques extract informative statistics from the observed data to develop a model that minimizes the error of these predictions.

Formally, given observation(s) of a variable (or set of variables) X that occur up to time t, we would like to develop some model $f(X)$ that describes the value of a variable (or set of variables) Y at some time greater than t. The variables X and Y can either represent separate variables, or instances of the same variable at different points in time. In the first case, we are using information from a separate process to estimate the future value of the variable of interest. In the latter case, we can use historic observations of a variable ($X_{0...t}$) in order to estimate the future values of that variable ($X_{t'>t}$).

A machine learning approach to forecasting involves developing a model by updating its parameters based on the model's performance on a training set. Formally, given input data X, target values Y, and initial model parameters θ, we learn a model $f(X;\hat{\theta})$ that minimize an error measure L between our estimates $\hat{Y}$ and the true target values Y. Ultimately, our goal is to develop a generalizable model that minimizes $L(Y,\hat{Y})$ on previously unobserved inputs from a domain of interest.

2.1.2 Design Decisions

Selecting the modelling approach involves a series of decisions informed by the task, the domain, and the statistical properties of the observed data set. This section will discuss how several of these properties affect the development of a forecasting model, which will later inform the development of an EEG forecasting model.

System Definition

As demonstrated at the beginning of this chapter, defining the **system** of variables involves selecting input variables that contain sufficient information to model the dynamics of the target variable. A simple choice would be to use historical observations of a variable to predict its future values, such as in the case of autoregressive models that compute forecasts based on weighted combinations of past observations. Alternatively, one can increase the complexity of the model through the inclusion of multiple variables, which may provide relevant information about the system to inform our forecasts. One can also consider including exogenous variables as inputs to the forecaster, which are variables whose values are conditionally independent from the other variables in the system. These values affect the other variables in the system, but are not conversely affected by them; thus, including exogenous variables can provide important information about the system that otherwise cannot be extracted by the other variables.

A a **model-based** approach entails incorporating prior knowledge about the rules that govern the system [Bis10]. Prior knowledge uses known properties of the system to reduce the hypothesis space of the model we are trying to learn. In the context of this work, we rely on current understandings of neuroscience and EEG dynamics to create a domain-specific approach to aid the development of a forecasting algorithm.

Statistical Properties: Stationarity, Seasonality, and Stochastisticity

There exists a large corpus of techniques used to extract descriptive statistics about data to guide model development. One can either remove these properties from the data, or adapt the modelling approach to accommodate the data. For example, removing the trend or seasonality of the data stabilizes the long-term mean and variance of the data, allowing the model to more easily learn statistical features that are consistent over different stretches of data. One can alternatively use a forecasting technique that adapts to non-stationary statistics over time, or set other hyperparameters appropriately such as the time delay and forecasting horizon. Additionally, the presence of large artifacts caused by exogenous factors can interfere with learning the temporal dynamics of the data. Removing these artifacts can simplify the process of learning a model that generalizes to unseen data.

Time Delay and Forecasting Horizon

Another key set of hyperparameters may include the **time delay** and **forecasting horizon**, which describe the time scale of our model. The time delay, also called the time lag, refers to the number of time steps into the past fed into the model in order to make a single estimate of future value(s). The time delay contributes to the size of the input of the forecasting model and often require an increase in the number of parameters to be learned. A model with too much capacity may overfit to the training set, since much more data is needed to effectively train the model. On the other hand, given too few time lags, the model may not have enough temporal context to learn long-term temporal dynamics.

The choice in forecasting horizon often depends on the end-goal of the forecasting system. Developing a model that creates forecasts of k consecutive time steps ahead into the future is referred to as a direct forecasting model. The size of k may be dependent on how much future data the task requires. Like the choice in time delay, optimizing a system to estimate a larger forecasting horizon often requires an increase in model capacity, which has an adverse affect on the model's generalization in practice. Alternatively, an iterative forecasting model is optimized using a forecasting horizon of $k = 1$. During test time, a trained iterative model can be used to accumulate a sequence of one-step-ahead predictions in order to create an output of the desired length. The correct choice between a direct and iterative model depends on factors such as the statistical characteristics of the data distribution, the chosen time delay, and computational constraints [MSW10][Che07][FL10].

Like the time delay length, too high of a forecasting horizon using a direct forecasting model increases the number of output parameters to be fit, while also increasing the capacity and temporal context of the model. Selecting these hyperparameters requires a consideration of the a bias-variance trade-off, and many forecasting algorithms use measures such as the Akaike Information Criterion (AIC), Bayesian Information Criterion (BIC), or other goodness-of-fit tests to estimate the optimal model capacity needed to minimize performance error.

Linear versus Non-linear Systems

Another key factor in model selection involves the choice of developing a linear versus a nonlinear model of the system. Many classical forecasting techniques model time series data using linear processes. This involves learning a set of parameters that are transformed only through linear operations. Linear models have the advantages of being more simple to implement, interpretable, and computationally inexpensive [PRT96]. Additionally, many real-world systems that are nonlinear can still be adequately approximated using linear methods [BM91] . However, some systems are naturally too complex to be approximated by linear forecasters, including systems characterized by stochasticity, chaos, and many unknown factors [Str94]. The choice between linear and nonlinear modelling is dependent on the nature of the system and the threshold of tolerable approximation error for the task.

2.2 Forecasting Algorithms and Techniques

This section will explain the selection of forecasting algorithm used to model the system, which is often the most important decision during the development of the model. The choice of algorithm depends on a number of factors, which include the properties of the data distribution, computational constraints, and the nature of the task to be accomplished.

2.2.1 Autoregressive Models

An autoregressive (AR) model describes a linear relationship between a variable and its past values. Take a univariate time series of n points $\mathbf{x} \in \mathbb{R}^n = x_1, x_2, ..., x_n$. If s_t denotes the observation of s at time point t, given an vector $\mathbf{s} = \{s_i \in \mathbb{R} \mid t - p < i \leq t\}$ of p lagged observations and a single point x_{t+1}, we can describe the AR model as follows:

$$x_{t+1} = \sum_{i=0}^{p} w_i \cdot x_{t-i} + \varepsilon_{t+1} \tag{2.1}$$

where $\mathbf{w} \in \mathbb{R}^p$ denotes the parameters which apply a linear combination to the lagged input. The linear combination of the lagged values are summed with a noise process ε_{t+1} to create the unobserved future value x_{t+1}.

AR models can be extended to a multivariate case by using vector autoregressive (VAR) models [ZW06]. Take a multivariate times series of n points $\mathbf{X} \in \mathbb{R}^{n \times c} = \mathbf{X}_1, \mathbf{X}_2, ..., \mathbf{X}_n$, where each of the n observations $\mathbf{X}_t$ is composed of a vector of c variables, such that $\mathbf{x}_t \in \mathbb{R}^c = x_1, x_2, ..., x_c$. A VAR model computes a weighted sum of the input matrix $\mathbf{X} \in \mathbb{R}^{p \times c}$ of p lagged time points and c variables using a parameter matrix $\mathbf{W} \in \mathbb{R}^{p \times c}$ to model the each variable of the matrix $\mathbf{x}_{t+1}$. The model can be expressed as follows:

$$\mathbf{X}_{t+1} = \sum_{i=0}^{p} \sum_{j=0}^{c} x_{(t-i)j} \cdot w_{ij} \tag{2.2}$$

Another useful extension of the AR model is a mixture autoregressive (MAR) approach.

An MAR model asserts that the shape of the conditional distribution of a forecast depends on the recent history of the process. Accordingly, the MAR model allows one to learn a mixture distribution, such as through simultaneously learning several AR models as components of this mixture.

2.2.2 Deep Learning

Deep learning refers to a family of machine learning techniques that learn a hierarchical mapping between a source distribution and a target distribution. Deep neural networks, a class of architectures within deep learning, achieve this goal by chaining a series of transformations to extract relevant features from the input to estimate a target output.

Over the past decade, deep learning has demonstrated success in a variety of machine learning problems, such as in the domains of computer vision, audio, and natural language processing [KSH12][Hin12][BCB14]. This success is partly owed to the development of deep architectures that involve many layers of convolutions, recurrent connections, and non-linear activation functions. Given a sufficiently large corpus of training data, deep neural networks can extract features from that data to perform tasks such as classification, generation, and forecasting.

Referring to previous uses of deep neural networks in time series analysis suggests a body of techniques that can be applied to neural time series data. In particular, we will examine architectures for sequence-to-sequence learning that can be applied to EEG forecasting.

Recurrent Neural Networks

Recurrent neural networks (RNNs) are network architectures that involve recurrent layers, which consist of operations that are repeatedly re-applied to a sequence of inputs and their intermediate outputs. The recurrent layer can output a sequence of these intermediate outputs, or the single output that is the final result of all previous intermediate computations. Given that each step in the recurrent transformation depends on the outputs of the previous steps, an RNN is able to maintain information about its history via the current state of its intermediate value. This intermediate, or *hidden*, state h contains the ”memory” of past states of the system, and thus, can be used to describe the temporal dynamics of a system. Formally, given an input x_t at time step t, we can express the state h_t as:

$$h_t = Wx_t + Uh_{t-1} \tag{2.3}$$

where W and U are weight matrices applied to the input and hidden state, respectively. Through this expression, we see that h_t is described using the previous state h_{t-1}, which can further be unrolled to show a dependency between the hidden states of time steps from time steps $[0, t-1]$. Thus, recurrent operations maintain information about the past that provide each estimate with temporal context of the sequence.

The concept of learning long-term dependencies is further developed in extensions to the basic RNN architecture, such as the long short-term memory (LSTM) network [HS97]. LSTMs introduce gating structures to each recurrent unit, such that the model can adjust the amount of time that past hidden states can influence the current output. Attention mechanisms were later introduced to also adaptively control how each past time step contributes to the output of the current state [VSP+17].

Convolutional Neural Networks

Convolutional neural networks (CNNs) are neural networks that consist of layers that convolve a shared set of weights over multiple sections of the input. The convolution is useful for extracting features from patterns that appear across multiple parts of the input. While CNNs are most well-known for their effectiveness on image data, convolutions have also been effective in extracting features from time series data as well. In time-delay neural networks (TDNN), convolutions are applied across time, such that the same set of weights are applied to multiple time steps [WHH+89]. Furthermore, CNNs have played a key role in recent sequence modelling architectures, such as those used in speech synthesis [ODZ+16].

Deep Autoregressive Networks

We can take the techniques described in the previous two sections and explain how they can be applied in the autoregressive context described in **Section 2.2.1**. An autoregressive model describes a process in which the future data in the time series can be described using a linear combination of previous values, although some researchers have also explored nonlinear variants of AR modelling. Thus, both RNNs and CNNs can be similarly structured to intake a sequence of data from past time steps in order to output an estimate of future data. Chaining multiple layers of recurrent, convolutional, and/or nonlinear transformations in this way can extract features from past time steps to compute the forecast of a future time point. Autoregressive processes modelled using deep neural networks are a key component of the methods described in this work, and they connect previous research on neural time series forecasting and current advances in deep learning.

Applications

Previous applications of deep learning techniques to time series data suggests that these techniques may similarly be used for EEG analysis. Sequence-to-sequence learning using deep neural networks has been applied to domains such as speech synthesis, econometrics, and video prediction [ODZ+16][FSG17][MCL16]. Of particular interest is research in time series domains with similar emphasis on spatial, spectral, and temporal properties as EEG, such as audio and speech. Recent advances in audio synthesis have used RNNs and autoregressive CNNs to accurately generate audio conditioned on a sequence of observed data inputs [CBS+15][ODZ+16][AvdO17][MKG+16].

2.3 Forecasting the Neural Time Series

The primary focus of this work involves applying forecasting techniques to neural time series data. Many of the above techniques can be readily adapted to EEG data.

We can frame our understanding of time series forecasting now using the vocabulary of EEG sequence prediction. Given some EEG data that has occured up to some time t, the goal our our model is to predict the sequence of EEG data that occurs after time t.The multi-channel case of EEG sequence prediction is illustrated in **Figure 2.1**.

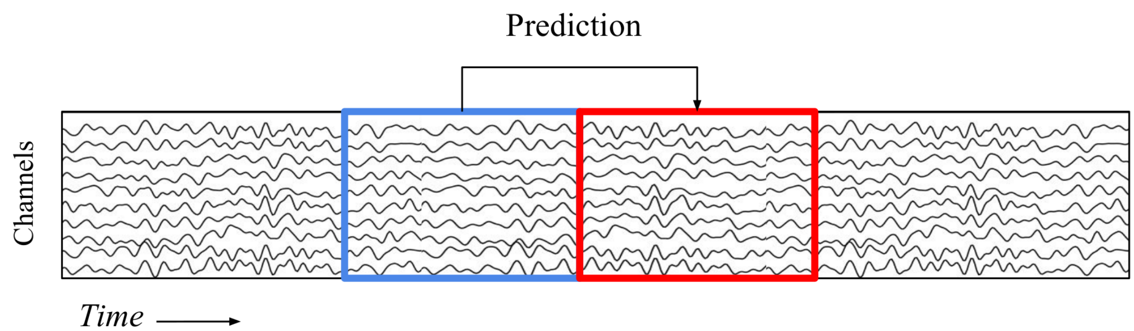


Figure 2.1: Predicting the EEG time series.

This section will explore the history of how forecasting has been applied to EEG, and then frame the problem of forecasting in the context of a time-locked event.

2.3.1 Prior Work

Forecasting the EEG time series has been a topic of interest throughout the history of computational EEG analysis. Much of the initial research on this problem focused on using linear autoregressive models [PRT96] [FB85] [BM91]. Researchers have also explored other types of linear modelling, such as wavelet transforms and related spectral methods [BBD92] [PRT96]. Although these prior works have been able to model a portion of the signal, research into the statistical properties of EEG questioned the adequacy of linear methods for modelling EEG [HVB+95]. Researchers have explored using nonlinear techniques to forecast EEG, since neural signals have been shown to contain statistical properties of nonlinear dynamical systems. For example, EEG is highly non-stationary, meaning that simple, non-adaptive linear models cannot model the signal through distribution shifts over long periods of time [MBFK87]. Researchers have also analyzed EEG using statistical tools from chaos theory such as correlation dimension estimation to show that the signal can be characterized by nonlinear dynamics [PD95]. EEG has also been shown to be successfully modelled by stochastic limit oscillators, further demonstrating nonlinear properties [BdP04]. Lastly, empirical comparison of linear and nonlinear techniques in a variety of experiments have shown that nonlinear methods can outperform linear methods on a variety of data sets [HVB+95] [BD86].

Deep learning architectures have begun to make an impact in several types of EEG analysis. Much previous work in recent years have focused on applying deep neural networks to event classification of time-locked signals. Several networks use spatio-temporal convolutions in order to extract features from the raw EEG data in order to achieve event classification [LSW+16] [SSF+17] [MG15][CG10] [TH16]. Other networks have explored the use of RNN for predicting EEG in both raw and spectrally pre-processed forms [SSOG15][BRYC15].

2.4 Forecasting Event-Related Potentials

2.4.1 Event-Related Potentials

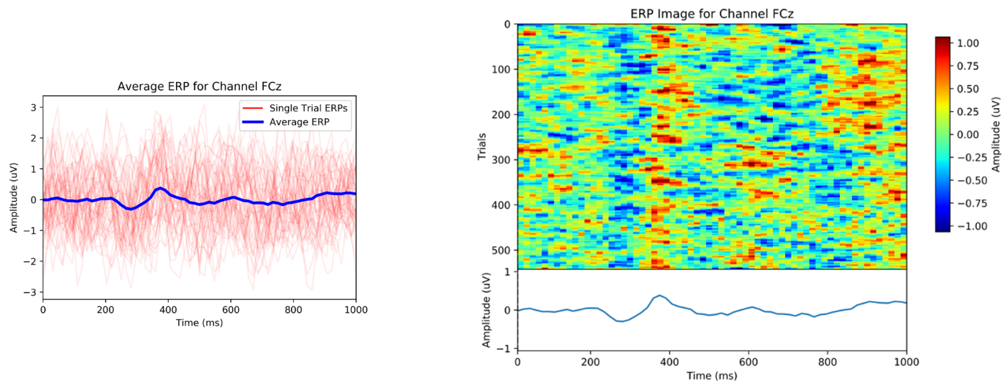
The study of event-related potentials (ERPs) involves characterizing the changes in EEG dynamics in response to a perceptual, behavioral, or cognitive event. Event-related brain dynamics are known to carry information about the cognitive state before and during an event [Luc12]. Unique ERPs have been identified for a variety of event types, such as motor activity [KD65] and sensory stimuli [Reg89].

One technique in ERP analysis involves computing the *average ERP* (AERP) over a ensemble of labeled instances of an event. The AERP reveals an underlying consistent morphology within the ERP waveform, which is a combination of several voltage peaks, or components, which are characterized by their latency, amplitude, and morphology. Several dimensionality reduction techniques can be used to extract these components from the original signal by transforming the raw data into a collection of maximally independent components. Previous research has demonstrated that these components can be associated with the activation of one or more specific brain areas during an event [JMM$^+$01].

The AERP reveals an underlying structure by masking the variation between single-trial neural responses. This variability stems from differences in internal state, stimulus irregularities, and phsyiology between individuals. Several factors pertaining to the cognitive state at the time of the event also affect how the components of the ERP are expressed on a given trial. Some of these indicators of cognitive state can be decoded in the pre-event EEG data, such as the relative power of various oscillatory bands found in the spectral decomposition.

This variation is demonstrated in **Figure 2.2**, which presents two views of the relationship between single-trial ERPs and their average underlying structure. A key indication of this variability is the differences between the main peaks of the ERP. As can be observed in **Figure 2.2b**, the peaks exhibit a range of latencies and amplitudes. Furthermore, several trials do not exhibit a clear first positive peak at 200 ms.

This variability holds importance in the analysis of single-trial ERPs. Characteristics such as peak amplitude and latency can be used as indicators of cognitive state or behavior before or during an event. For example, a delayed initial peak during a decision-making task may be an indication of the subject's current level of engagement [LS98]. This effect suggests a relationship between a subject's pre-event cognitive state and the subsequent ERP waveform.



(a) (b)

Figure 2.2: The inter-trial variability of ERPs. (a) An average ERP for a single event type for channel FCz from a single subject (blue) is overlayed on 30 single-trial events from the session (red), demonstrating a variability and noise that is attenuated during ERP averaging process. (b) An ERP Image (top) of 580 trials for a single event type for channel FCz from a single subject. Several common peaks and components that vary in latency and amplitude over each trial can be observed. The average ERP (bottom) is displayed for reference.

2.4.2 Forecasting the ERP

In the context of forecasting, an accurate model must be able to accommodate the occurrence of cognitive events. The presentation of a stimulus is an exogenous factor that cannot directly be inferred from pre-event EEG data. The information gained from ERP averaging is useful for understanding the general trend that may occur after the onset of an event; however, it masks the variation that occurs on a trial-to-trial basis. These characteristics must be captured by the model in order to accurately estimate the EEG sequence during the occurrence of an event. Therefore, a forecasting model can be used to generate a more accurate forecast than what is available through ERP averaging techniques. Given prior knowledge that an event occurs at a particular point in time, we can condition a forecaster to modify its prediction to take an event-related response into account. To simplify this idea, we can consider a forecast that occurs at the onset of an event at time $t = 0$. Using observed data from time $t - p + 1$ to t, where p is the time delay of the model, we construct a forecast of the data from $t+1$ to $t+k$, where k is the forecast horizon. In other words, the model accepts takes the pre-event data as input, in order to forecast a certain amount of post-event data.

We can re-frame the previous problem statement given in **Figure 2.1** with the illustration in **Figure 2.3**, which shows how pre-event EEG data is used to predict post-event EEG data.

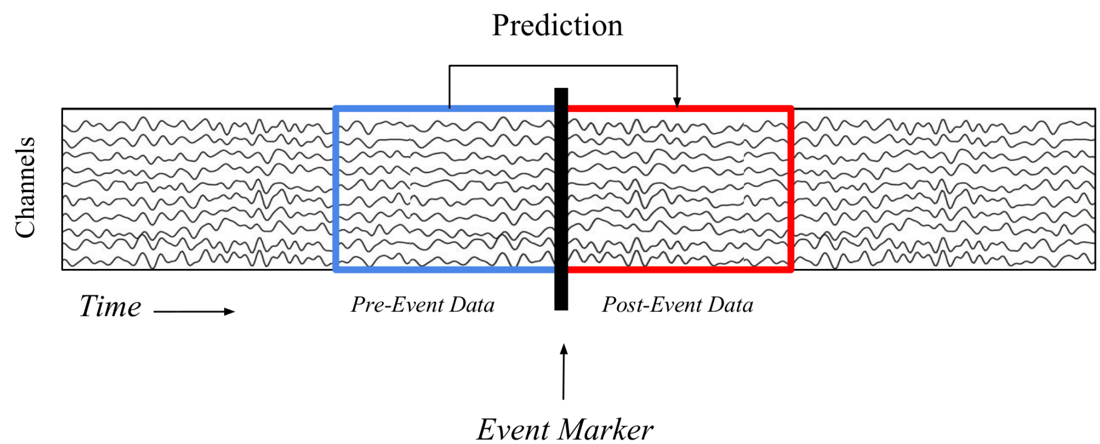


Figure 2.3: Forecasting the EEG time series in the context of an event. Given some pre-event EEG data and knowledge that an event has occurred, the model makes a forecast of the post-event EEG data.

To our knowledge, there has not been any published attempts at using deep neural networks for forecasting event-related potentials of EEG. In this thesis, we hope to bridge previous work on autoregressive forecasting of EEG signals with recent innovations in deep learning in order to develop methods that can accurately model nonlinear relationships of the EEG time series and generate accurate forecasts of ERPs.

Chapter 3

Methodologies

The previous chapters in this work have established the background of forecasting techniques for time series data. Many researchers have concluded that EEG dynamics are best described using nonlinear models as opposed to linear methods such as autoregression. However, there exists no research on using nonlinear deep neural networks to develop a forecasting model of EEG, despite the prevalence of deep learning in time series analysis and EEG event classification.

The methods in the work suggest an approach for implementing a deep neural network to forecast EEG data. In particular, we will investigate the extent to which the future time course of single-trial event-related brain dynamics can be predicted from pre-event EEG using this method. To achieve this, we will use a theoretical framework of event-related brain activity to develop a predictive model for single-trial EEG data.

3.1 Theory

There are a number of theories that attempt to characterize the relationship between evoked responses and ongoing brain dynamics. Certain models posit that ERPs arise due to phase resetting of ongoing activity following an event [MWJ$^+$02]. Other models place emphasis on the role of the stimulus-evoked response in the formation of the ERP [SSK+02].

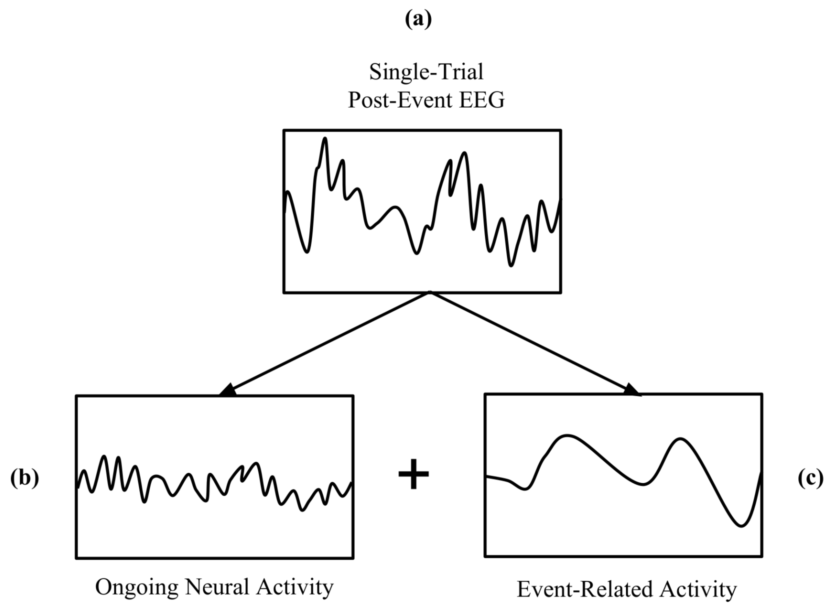


Figure 3.1: The variable signal plus ongoing activity (VSPOA) model of the ERP describes each single-trial post-event EEG waveform (a) into two constituent parts: ongoing neural activity (b) and event-related activity (c).

We consider the variable signal plus ongoing activity (VSPOA) model of the ERP [CBK+06], which characterizes an ERP as the linear sum of ongoing and evoked neural activity (see **Figure 3.1**). The ongoing activity may be considered any signal that is not directly connected to the event-related evoked activity. The VSPOA model treats the event-related portion of the post-event waveform as a sum of one or more event-related component ”templates”, whose amplitudes and latencies vary between individual trials. One must first extract a set of temporal components associated with an event. These components are common to the neural responses of all instances of this event. In the model's original formulation, these components were initialized as a set of Gaussian basis functions, with one component designed to accommodate for every major peak that can be observed in the average ERP.

Using this set of components, a single trial can be modelled by setting the amplitude and latency of each component. Then, the linear summation of these modified components will recreate the evoked response of the waveform. This process is illustrated in **Figure 3.2**.

The resultant post-event waveform is produced from a combination of these transformed components and the ongoing baseline activity, a relationship which can be described as follows:

$$x_r(t) = \sum_{n=1}^{N} a_{nr} s_n(t - \tau_{nr}) + \eta_r(t) \tag{3.1}$$

where x_r is the observed brain signal of the r^{th} trial. Each of the N event-related components s_n is modified by a_{nr} and τ_{nr}, the single-trial amplitude and latency of these components. The modified components are summed with the ongoing neural activity, η_r, which is modelled by a noise process in the original VSPOA model.

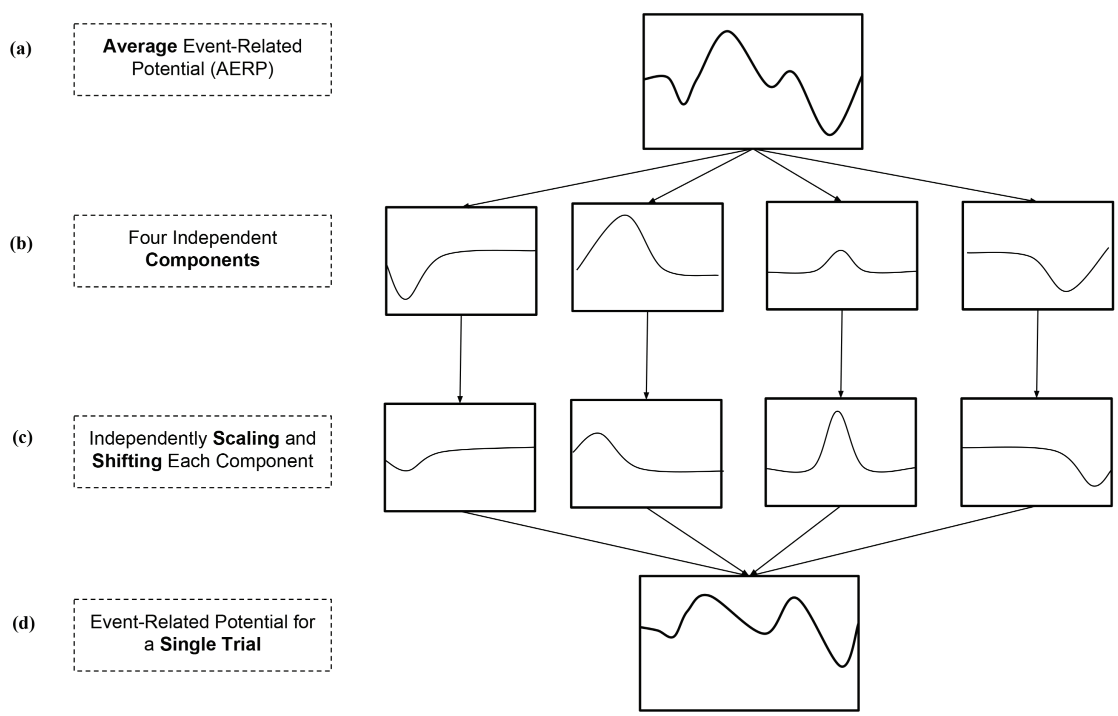


Figure 3.2: An illustrative example of the single-trial modification of event-related components in the VSPOA model. The AERP (a) can be broken down into four temporal components (b). Each of these components can be scaled and shifted independently in order to fit the evoked waveform of a given single-trial (d).

A variant of the VSPOA approach, proposed in [TKS+03], characterizes the ongoing activity as an autoregressive (AR) process:

$$\eta_r(t) = \sum_{k=1}^{p} w_k \cdot \eta_r(t-k) + \varepsilon_r(t) \tag{3.2}$$

In this variant, the ongoing activity of the r^{th} trial at time t, $\eta_r(t)$, is represented as a weighted sum of p past observations of η_r with additive Gaussian noise, $\varepsilon_r(t)$.

3.2 Architecture

We propose a novel deep learning approach that constructs accurate forecasts of the post-event waveform using a VSPOA-like characterization of the ERP as described in **Formula 3.1**. The basis of our method involves forecasting single-trial event-related neural activity by jointly predicting ongoing activity and event-related components. The event-related portion of this prediction is further subdivided into a series of intermediate predictions of latency and amplitude parameters of these components. The goal of these divisions is to allow the network to independently modify only certain aspects of the signal, generating a prediction that captures the main sources of variability suggested by the VSPOA model of the ERP.

The overall structure of this architecture is illustrated in **Figure 3.3**.This architecture contains two main pathways, used to forecast the data in two parts: 1) ongoing neural activity and 2) event-related activity comprising the ERP. The event-related activity pathway is done in two steps, where the amplitude and latency parameters of each event-related component is separately modified in order to predict the trial's ERP.

3.2.1 Estimation of Single-Trial Parameters

Previous work using the VSPOA model have characterized single-trial ERP variability using amplitude and latency parameters for each component of the ERP [CBK+06][TKS+03].

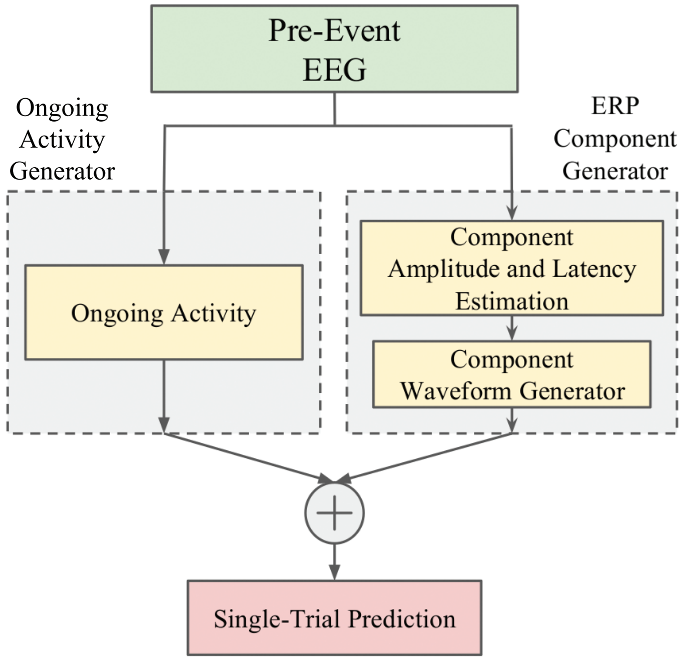


Figure 3.3: Overview of the deep neural network architecture.

Accordingly, we estimate the latency and amplitude parameters of each component using features extracted from the pre-event EEG data. We construct the parameter estimation algorithm as a nonlinear convolutional neural network, followed by a series of fully-connected layers, which learns a mapping between the pre-event input and single-trial component parameters, shown in **Figure 3.4**.

Recent successes in using deep learning for EEG event classification have demonstrated that convolutional filters are effective feature extractors for raw EEG data [LSW+16]. Specifically, recent studies have demonstrated that temporal convolutional filters can specifically be used to extract spectral features from raw EEG [HSB18].

The pre-event EEG data is passed through the convolutional network, which uses a series of convolutional blocks to extract relevant features from the data. These features are then passed through two pathways of fully connected (dense) layers to extract features specific to the estimation of the amplitude and latency of each component. The final fully connected layer contains $2 \times N$ nodes, each of which represents a linear regression operation

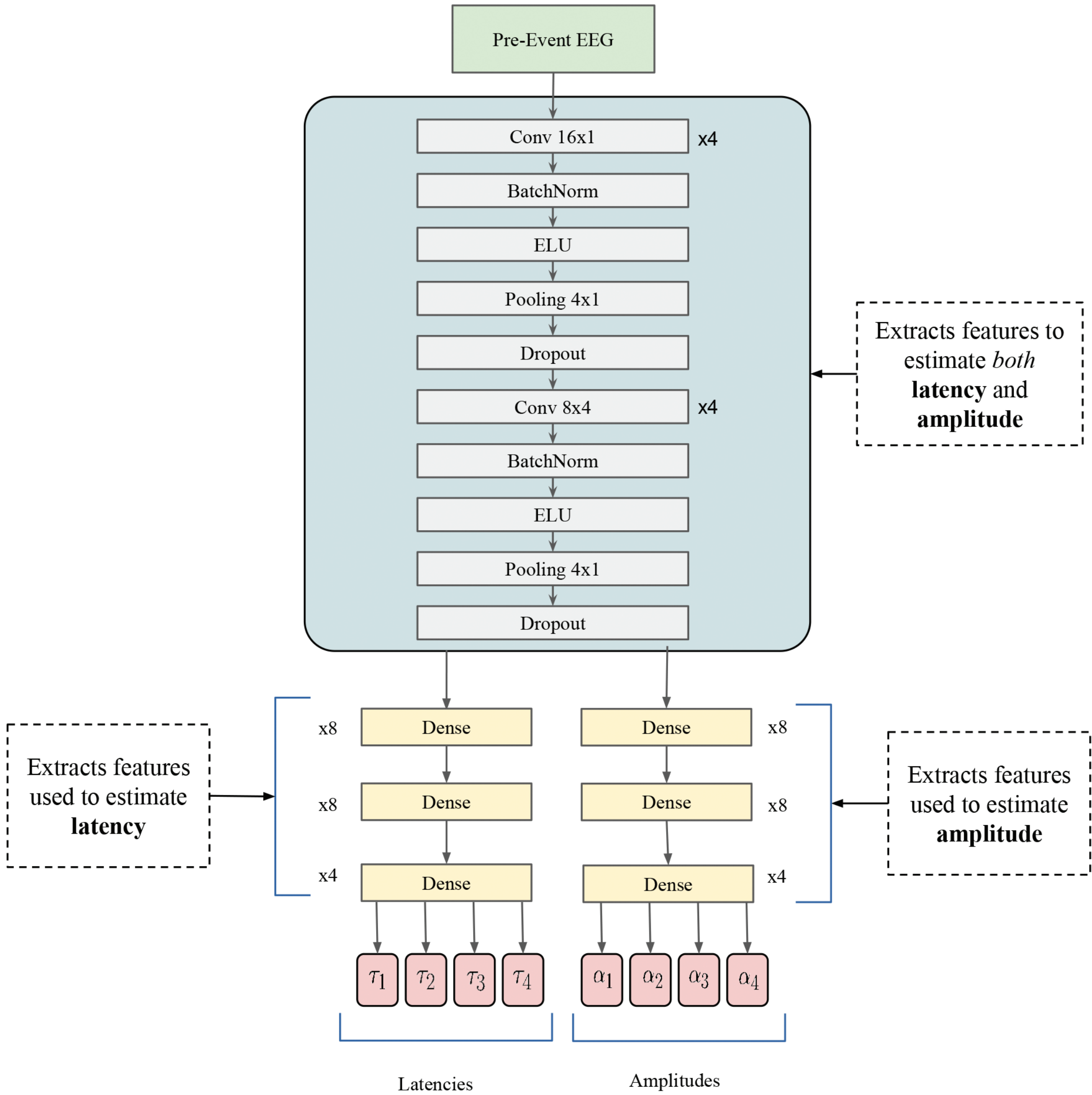


Figure 3.4: Architecture of the single-trial component amplitude (α) and latency (τ) estimator, shown here for N = 4 components.

to compute a prediction of the amplitude or latency value for each of the N components.

3.2.2 Component Waveform Generator

A baseline approach to modelling the event-related component templates would be to use the average ERP (AERP), which reveals the underlying event-related structure common to all trials. However, this approach typically fails to adequately characterize single-trial variability in the ERP. Previous work on single-trial ERP estimation suggests that ERPs consist of multiple components that vary independently [CTB78].

In our model, we learn n components in parallel, such that the network can estimate a single-trial τ_n and α_n for each component s_n. An illustration of the component generator is shown in **Figure 3.5**. The network learns each component waveform as a set of weights, which represent the signal amplitude at each time step up to one second after the event onset. These initial set of weights act as the template component, which is modified by the amplitude and latency parameters estimated based on the single-trial pre-event EEG data. For each trial, to scale the amplitude of the component, these weights are multiplied by the estimated amplitude scale α_n of that trial. To temporally shift the component, we create a series of vector masks based on the estimated latency τ_n, such that when the weights of each component waveform is convolved over the masks, the output produced is a shifted version of the initial component.

Finally, we construct the single-trial ERP waveform c_{nr} by summing these components together. We can formalize this process as follows:

$$c_{nr}(t_i) = \sum_{n}^{N} \alpha_{nr} \cdot s_n(t_i + \tau_{nr}) \tag{3.3}$$

3.2.3 Ongoing Activity Generator

As part of the VSPOA model of the ERP, we must also generate the ongoing activity to be added to the prediction of the ERP components. Here we model the ongoing activity as an order-p autoregressive (AR) process.

The AR process performs the linear transformation described in **Equation 3.4**, generating a one-step ahead prediction. Given pre-event EEG data as the initial input, the AR process iteratively generates T samples of future data, where the output of a previous time window is used as input for the subsequent time window.

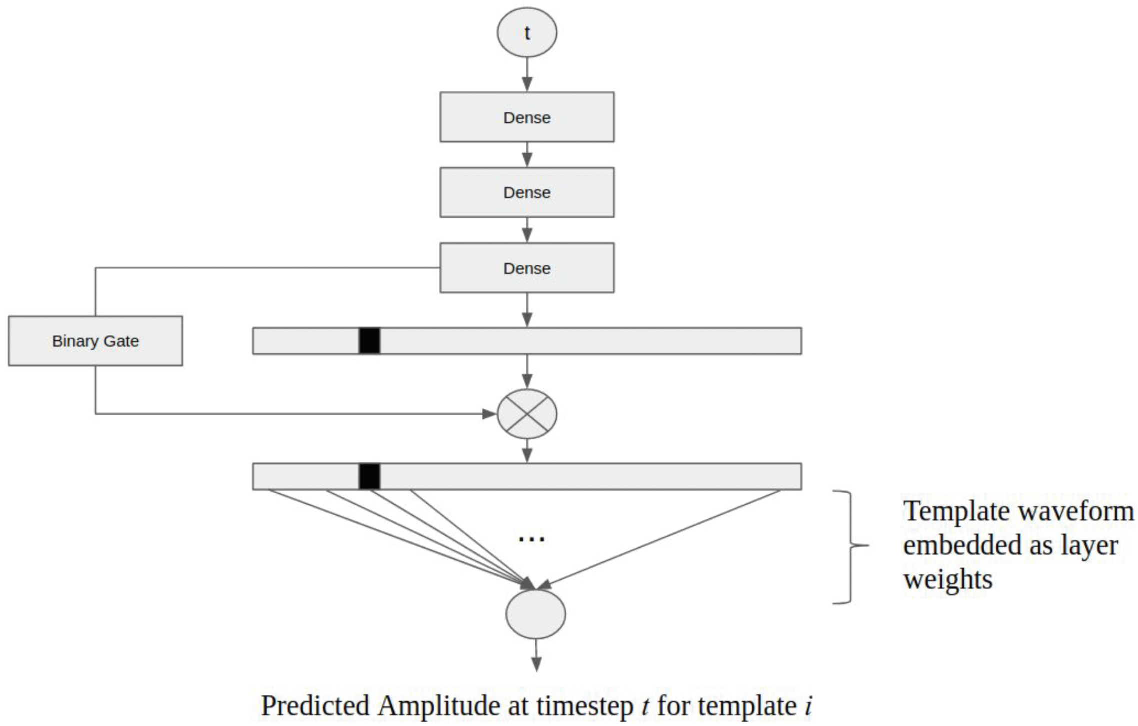


Figure 3.5: Architecture of the component generator, which learns a set of N functions to model N components, which are modified by single-trial latency (τ) and amplitude (α) parameters to generate the single trial event-related potential.

Autoregressive processes can be used to describe stationary data of a particular spectral distribution. However, given that EEG data is generally non-stationary, a single spectral distribution may not accurately describe the underlying activity at different points in time or across different trials. This problem can be rectified by using an adaptive AR model, or a mixture of several AR models, in which the mixture of the models is adjusted to fit a given time window or trial.

We implement the AR process using a fully-connected dense network, composed of p fully connected units in its input layer. To implement the mixture of AR models, we use $K = 4$ parallel AR processes which are combined through a weighted summation:

$$\eta_r = \sum_{k=1}^{K} w_{kr} \cdot \eta_{kr} \tag{3.4}$$

where the weights w_{kr} are estimated from the pre-event data for a given trial, using a 3-layer fully-connected network, with ReLU activation functions for each hidden layer node and a softmax activation function for each of the K output nodes.

Chapter 4

Experiment

In this section, we will apply our proposed method to a data set of experimental EEG data. By evaluating the model's performance on this data set, we can understand and assess the strength of the model's performance on predicting event-locked EEG data.

4.1 Setup

4.1.1 Dataset

We obtained a series of EEG recordings from a Rapid Serial Visual Presentation (RSVP) task [BSVRM08]. This task involved showing a human subject a sequences of images in rapid succession (12 images per second within a 4.1 second "burst"), where each image either did or did not include a target feature. The subject was instructed to press a button after a burst to indicate whether or not a target was present in one of the images of the sequence. A full experimental session involved multiple bursts presented to a subject in a single session, and each subject (except one) participated in two sessions. The event of interest in our experiments is the "Visual Target" event, in which an ERP is evoked upon a subject's detection of a target.

The study included 15 1-hour sessions from 8 different subjects. The data was originally recorded with 256 channels at a sampling rate of 256 Hz, which we downsampled to 64 Hz. For all experiments, we performed our analysis on data from channel FCz from the 10-20 electrode system. Furthermore, there were 3,562 "Visual Target" events in the total data set.

Preprocessing

In order to remove environmental and physiological artifacts, we applied a small amount of pre-processing using methods that preserved causality in our data. The filtering steps consisted of a minimum-phase causal 1-Hz high-pass FIR filer (filter order: 1690) and 30-Hz low-pass FIR filter (filter order: 384). Prominent ocular artifacts were removed using FastICA and EyeCatch [SKDKM13]. Heavily contaminated windows were removed using an online bad window removal method which discarded 1.5 second long windows of data wherein 15% or more of the data points within the window have an EEG voltage greater in magnitude than 4 standard deviations from the window's mean voltage. The recordings of each subject were processed separately, which ensured that no information was passed between the training and testing set during the cross-validation procedure.

4.1.2 Training

For training and model evaluation, we used a leave-one-subject-out cross-validation procedure. The data set was separated into 8 folds where, for each fold, data from all subjects except one was combined into a single training set, while the eighth subject was held out as a test set. The training data was sliced into maximally-overlapping 1.5 second windows to create a set of training samples (see **Figure 4.1**). We used the first second of each slice as input to our model and the last 0.5 seconds of each slice as the prediction target (see **Figure 4.2**). For each training sample, both the model input data and target data were standardized by subtracting the mean and dividing by the standard deviation of the input portion of the sample. Each sample was associated with the time in seconds from the closest Visual Target event, which informed the model to activate the component generator when the event occurred within the prediction target window.

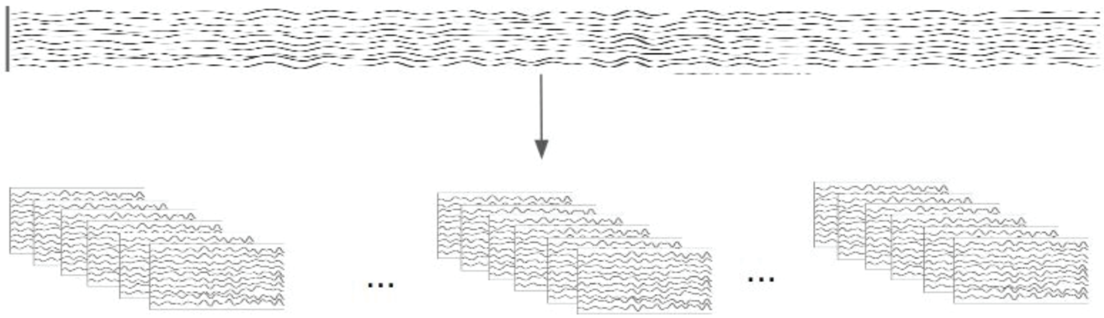


Figure 4.1: Maximally overlapping 1.5 second segments of the continuous training data. Each of these slices become a training example by further slicing each segment into an input segment and an output segment.

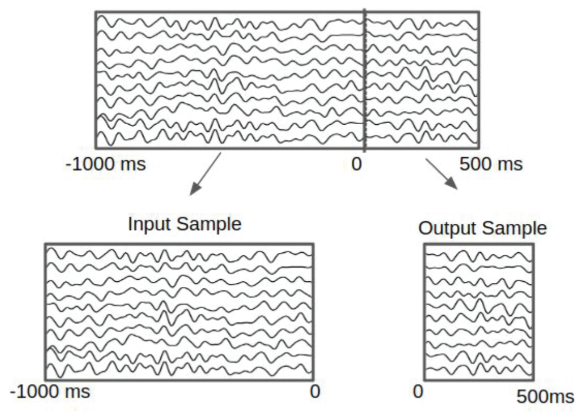


Figure 4.2: Illustration of dividing a segment into an input-output pair. A 1500 millisecond (1.5 second) slice is divided into an input segment (first 1000 ms) and an output segment (last 500 ms). The input segment is passed as input to the model, which attempts to estimate the values of the output segment.

During training, we fit the model on this collection of slices in order to minimize the mean-squared error (MSE) between the predicted waveform and the target waveform. The model was optimized using mini-batch stochastic gradient descent with an adaptive learning rate computed by the *RMSprop* algorithm. We iteratively trained the model until an early stopping criterion was reached, which was when the loss on a hold-out set (10% of samples randomly selected from the training set) failed to improve more than 0.001 for 5 iterations.

The weights during the iteration with the best performance on the hold-out set were saved for evaluation against the test subject's data. In order to evaluate model performance in predicting post-event data from the pre-event data, we only test using those slices where a Visual Target event appears $t = 1$ second into the slice.

4.2 Evaluation

We evaluated performance in predicting post-event waveforms for each test subject using the following models:

1. *Training Set Average ERP (AERP)* [Baseline]:

   We use the training set's AERP as a prediction for every sample of the testing set. We treat this as our baseline.

2. *Component Generator (CG)*

   We use the component generator without the addition of the predicted ongoing activity.

3. *Autoregressive Process (AR)*

   We use the autoregressive process without the addition of predicted variable signal.

4. *AR + AERP*

   We use the autoregressive process in combination with training set's AERP instead of the component generator.

5. *AR + CG*

   We use a linear combination of the predictions of the component generator (CG) and the autoregressive process (AR).

The performance of each model was evaluated using the following metrics:

Table 4.1: Experimental results for our baseline (AERP) and various combinations of the component generator (CG) and an Autoregressive process (AR). All results are the average of leave-one-subject-out cross validation procedure.

Approach	**MSE**	**Correlation**
AERP	1.189 +/- 0.017	0.115 +/- 0.021
CG	1.179 +/- 0.026	0.177 +/- 0.015
AR	1.151 +/- 0.031	0.232 +/- 0.014
AR+AERP	1.113 +/- 0.024	0.278 +/- 0.005
AR+CG	1.119 +/- 0.026	0.284 +/- 0.008

Table 4.2: P-values computed with two-tailed t-tests on differences in mean MSE and Correlation for all pairs of models shown in Table **4.1**. Non-significant differences ($p > 0.05$) are highlighted in gray.

MSE \ Corr	**AERP**	**CG**	**AR**	**AR+AERP**	**AR+CG**
AERP		0.003	0.028	0.001	0.001
CG	0.109		0.094	0.001	0.001
AR	0.027	0.051		0.003	0.004
AR+AERP	0.000	0.000	0.002		0.494
AR+CG	0.000	0.000	0.004	0.373	

1. *Mean Squared Error (MSE)*:

   The average mean squared error (MSE) between the true and predicted single-trial ERP waveforms.

2. *Pearson Correlation Coefficient (Corr)*:

   The average Pearson correlation coefficient between the true and predicted single-trial ERP waveforms.

4.3 Results

4.3.1 Forecasting Evaluation Metrics

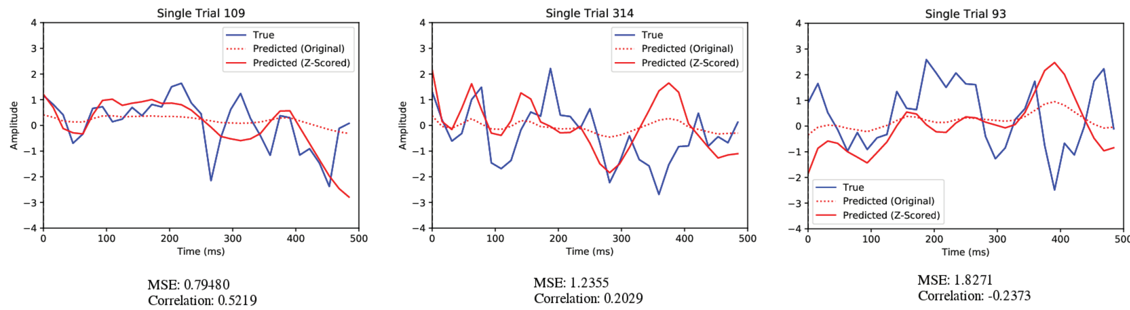
The results in **Table 4.1** demonstrate the performance of each of the models outlined in **Section 4.2**, with **Table 4.2** showing the associated p-values computed with two-tailed t-tests on the differences in mean MSE and Correlation metrics between models.

Both metrics demonstrate the effectiveness of using the combined AR+CG approach to predict the post-event EEG activity over the baseline AERP approach. We also see that the combined AR+CG approach outperforms the AR and CG models alone, The AR+CG model outperformed the AERP baseline suggesting that the prediction of post-event EEG can be improved when jointly modelled using the ongoing activity and variable signal process. However, we also point out that the AR+CG model did not significantly outperform the AR+AERP model, indicating that the majority of the improvement of the AR+CG model is due to the AR portion of the model.

4.3.2 Qualitative Analysis

Figure 4.3 shows several representative single-trial predictions for the AR+CG model randomly drawn from the top, middle, and bottom percentiles of the test set. The predicted waveforms are plotted along with the z-scored predictions, for a scale-invariant comparison of the waveforms.

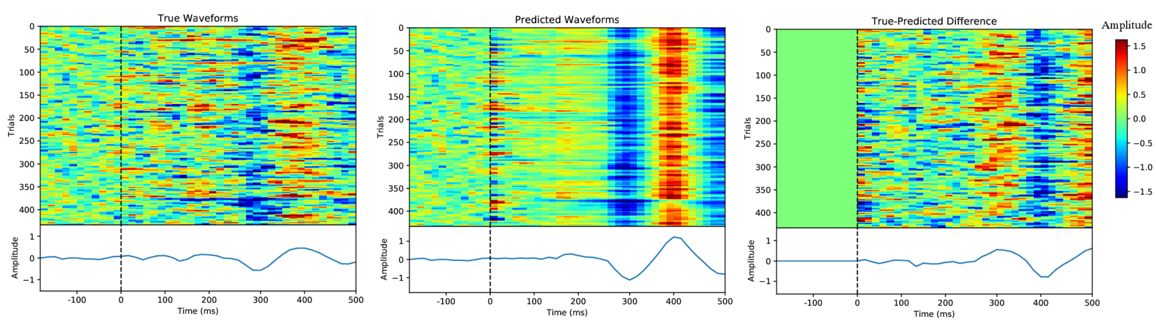
The ERPImage plot (with a 5-trial moving average) in **Figure 4.4** of the true, and z-scored predicted, waveforms as well as the true-predicted differences provides another view of the data. These figures demonstrate that the model can accurately capture the general underlying trend of the data. However, the model at times fails to capture prominent peaks and higher-frequency spectral components. Additionally, it is clear that the model generally underestimates the amplitude of the predicted waveforms. Possible reasons for this include



(a) (b) (c)

Figure 4.3: Three examples of single-trial outputs from a randomly selected test subject. The columns (a)-(c) demonstrate a prediction with high correlation, a prediction with medium correlation, and a prediction with low correlation, respectively

a lack of sufficient data to make a highly accurate prediction for each single trial, or having insufficient degrees of freedom in the model to capture variability across subjects and trials. We note that under conditions where a model has insufficient data or information to make an accurate time-series forecast, an optimal predictive estimator is often simply a prediction of the mean (or trend).



(a) (b) (c)

Figure 4.4: Three ERP Images are shown for the predictions of a single training fold, demonstrating the distribution of amplitudes over time for all of the trials. (a) and (b) show the true and predicted waveforms, respectively, while (c) shows the difference between the true and predicted.

A representative set of component templates learned on a randomly selected LOOCV training fold are shown in **Figure 4.5**. Their respective single-trial latency and amplitude parameters, estimated for all trials of eighth (test) subject of the same fold, are shown in **Figure 4.6**. While the component templates show peaks which may be characteristic of a (phase shifted) evoked ”target” response, the most prominent feature here is a relatively low-frequency oscillatory response. Further analysis in comparing these components to those found, for instance, by blind-source separation techniques such as ICA can lead to insights on how closely these components reflect individual ERP components.

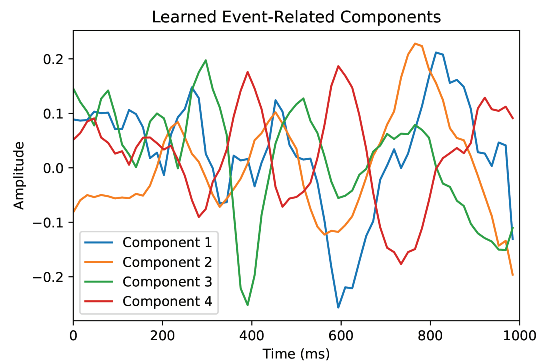


Figure 4.5: Event-related component templates learned by the AR+CG network for a representative training fold. To predict the post-event waveform, components are scaled and temporally shifted for each trial and summed with the prediction of the ongoing activity.

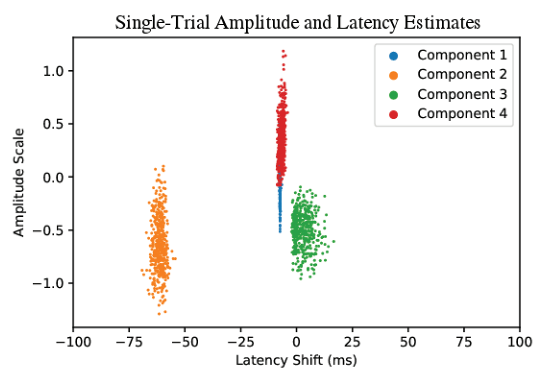


Figure 4.6: Distribution of single-trial amplitude and latency estimates for a representative test subject.

Chapter 5

Conclusion and Future Work

In this work, we established the groundwork for using deep neural networks to forecast neural time series data. Through exploring the use of machine learning for time series analysis in other domains, we identified that complex problems in EEG forecasting could benefit from recent work in nonlinear modelling using deep neural networks.

We proposed a novel algorithm for forecasting event-related brain dynamics from pre-event activity. We first considered the variable signal plus ongoing activity (VSPOA) model of the ERP, which suggests that the event-related neural response can be modelled by a set of modified temporal components and ongoing activity. We hypothesized that an adaptation of the VSPOA model can be used to design an architecture for forecasting this response.

To forecast the ERP, we implemented a deep neural network to concurrently predict ongoing EEG dynamics and event-related activity. The network jointly learns an autoregressive process and a set canonical ERP components in order to accurately forecast a single-trial ERP. For a given trial, the model produces a forecast of the ongoing activity and an estimate of parameters controlling the modification of the temporal components. These two predictions are then combined in order to generate a forecast of the post-event waveform for that single trial.

Our results show that the method is able to capture single-trial event-related activity compared to baseline methods. The pre-event EEG data contains sufficient information to generate low-error estimates of the post-event activity. Evaluating several modifications of the network revealed that the greatest improvement came from the autoregressive modelling of the ongoing signal, and that the inclusion of the component generator also contributed a small boost to the performance.

We regard this work as preliminary, with a number of open avenues for improving the predictive power of this approach. While the proposed approach using deep neural networks shows promise in predicting event-locked EEG data, improvements in pre-processing techniques, neural network architectures, and training procedures may further improve performance of these models. Using a model-based machine learning approach meant that the architecture was generated theoretically. While a theory-driven approach can help aid in domain-specific model development, faulty theoretical assumptions can lead to an ill-fit model. Although the VSPOA model has had success in ERP analysis, it is not proven that the model's estimates of the ERP parameters can be derived from the pre-event data. While prior work in ERP analysis has found correlation between pre-event brain dynamics and feature of the post-event waveform, more analysis is needed to prove whether this relationship is optimally modelled by the VSPOA framework.

Additionally, some aspects of our model were determined heuristically. We used our understanding of the VSPOA model, as well as our understanding of the characteristics of the ERP of interest, to inform several architecture choices. One example is the number of autoregressive models used in the mixture autoregressive model. This number was selected based on the assumed reasonable number of component distributions present throughout the non-stationary EEG signal. An empirical investigation through hyperparameter or architecture searches may provide better empirical evidence for these choices.

Modelling the ongoing EEG signal using a neural network remains an open question, with many unexplored deep learning techniques such as those mentioned in **Section 2.2.2** that may improve performance. In particular, recurrent structures such as LSTMs have show promise in sequence modelling. Dilated convolutional neural networks and other CNN structures offer computationally efficient ways to extract features from time series data. In future work we will investigate these techniques in order to determine whether there exists stronger architectures for neural time series forecasting.

As this predictive approach improves, we hope to train forecasting models in more complex contexts in order to understand how effective this approach can be for use in brain-computer interfaces. For example, training on a single event type allowed the model to overfit on the structure of this particular ERP. However, multi-task learning approaches may strengthen the model by allowing it to learn from various event types simultaneously. The model can then be structured to create conditional predictions based on the event label. Increasing the model capacity to handle multiple event types and to incorporate conditional knowledge of the event label of interest would be a step towards a more complex, real-world BCI context.

With further refinement, this approach may act as a useful component of a larger BCI application as discussed in **Section 1.2.1**, such as reducing the latency of existing BCI systems or guiding closed-loop intervention systems. However, as discussed, more fundamental work on improving the performance of machine learning forecasting methods must be accomplished before these ideas are ready for use in complex interfaces. We hope that this work lays foundation for future work in understanding the temporal dynamics of the neural time series.

Chapter 1-5 include material published in the proceedings of the 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC), ”Deep Neural Networks for Forecasting Single-Trial Event-Related Neural Activity”, Ibagon, Gabriel; Bidgdely-Shamlo, Nima; Kothe, Christian; Mullen, Tim. The thesis author was the primary investigator and author of this material.

Bibliography

[AAC$^+$18] German Abrevaya, Aleksandr Y. Aravkin, Guillermo A. Cecchi, Irina Rish, Pablo Polosecki, Peng Zheng, and Silvina Ponce Dawson. Learning nonlinear brain dynamics: van der pol meets lstm. *CoRR*, abs/1805.09874, 2018.

[ALK$^+$97] Ljubomir I. Aftanas, Natalia V. Lotova, Vladimir I. Koshkarov, Serguei A. Popov, and Victor P. Makhnev. Nonlinear forecasting measurements of the human eeg during evoked emotions. *Brain Topography*, 10:155–162, 1997.

[AvdO17] Igor Babuschkin Karen Simonyan Oriol Vinyals Koray Kavukcuoglu George van den Driessche Edward Lockhart Luis C. Cobo Florian Stimberg Norman Casagrande Dominik Grewe Seb Noury Sander Dieleman Erich Elsen Nal Kalchbrenner Heiga Zen Alex Graves Helen King Tom Walters Dan Belov Demis Hassabis Aaron van den Oord, Yazhe Li. Parallel wavenet: Fast high-fidelity speech synthesis. 2017. arXiv preprint.

[BBD92] Ea Bartnik, K.J. Blinowska, and P.J. Durka. Single evoked potential reconstruction by means of wavelet transform. *Biological Cybernetics*, 1992.

[BCB14] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. arXiv preprint, 2014.

[BD86] Agnessa Babloyantz and Alain Destexhe. Low-dimensional chaos in an instance of epilepsy. *Proceedings of the National Academy of Sciences of the United States of America*, 83 10:3513–7, 1986.

[BdP04] D. P. Burke and Annraoi M. de Paor. A stochastic limit cycle oscillator model of the eeg. *Biological Cybernetics*, 91:221–230, 2004.

[Bis10] Christopher M. Bishop. Model-based machine learning. *Philosophical Transactions of the Royal Society*, 2010.

[BM91] K.J. Blinowska and M. Malinowski. Non-linear and linear forecasting of the eeg time series. *Biological Cybernetics*, 1991.

[BRYC15] Pouya Bashivan, Irina Rish, Mohammed Yeasin, and Noel Codella. Learning representations from eeg with deep recurrent-convolutional neural networks. arXiv preprint, 2015.

[BSVRM08] Nima Bigdely-Shamlo, Andrey Vankov, Rey R. Ramirez, and Scott Makeig. Brain activity-based image classification from rapid serial visual presentation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 16(5):432–441, 2008.

[CBK+06] Yonghon Chen, Steven L. Bressler, Kevin H. Knuth, Wilson A. Truccolo, and Mingzhou Ding. Stochastic modeling of neurobiological time series: Power, coherence, granger causality, and separation of evoked responses from ongoing activity. *Chaos*, 2006.

[CBS+15] Jan Chorowski, Dzimitry Bahdanae, Dmitriy Serdyuk, Kyunghun Cho, and Yoshua Bengio. Attention-based models for speech recognition. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, volume 1, pages 557–585, 2015.

[CG10] Hubert Cecotti and Axel Graser. Convolutional neural networks for p300 detection with application to brain-computer interfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(3):433–445, 2010.

[Che07] Guillaume Chevillon. Direct multi-step estimation and forecasting. *Journal of Economic Surveys*, 2007.

[CTB78] Richard Coppola, R D Tabor, and Monte S. Buchsbaum. Signal to noise ratio and response variability measurements in single trial evoked potentials. *Electroencephalography and clinical neurophysiology*, 44 2:214–22, 1978.

[FB85] Piotr J. Franaszczuk and Katarzyna J. Blinowska. Linear model of brain electrical activityeeg as a superposition of damped oscillatory modes. *Biological Cybernetics*, 53:19–25, 1985.

[FL10] Philip Hans Franses and Rianne Legerstee. A unifying view on multi-step forecasting using an autoregression. *Journal of Economic Surveys*, 2010.

[FSG17] Valentin Flunkert, David Salinas, and Jan Gasthaus. Deepar: Probabilistic forecasting with autoregressive recurrent networks. arXiv preprint, 2017.

[Hin12] Geoffrey Hinton. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97, 2012.

[HS97] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9:1735–1780, 1997.

[HSB18] Kay Gregor Hartmann, Robin Tibor Schirrmeister, and Tonio Ball. Hierarchical internal representation of spectral features in deep convolutional networks trained for eeg decoding. *2018 6th International Conference on Brain-Computer Interface (BCI)*, pages 1–6, 2018.

[HVB+95] Jorge L. Hernández, Juan Valdés, Rolando J. Biscay, Juan C. Jiménez, and Pedro J Valdes. Eeg predictability: adequacy of non-linear forecasting methods. *International journal of bio-medical computing*, 38 3:197–206, 1995.

[JMM+01] Tzyy-Ping Jung, Scott Makeig, Martin J. McKeown, Anthony J. Bell, Te-Won Lee, and Terrence J Sejnowski. Imaging brain dynamics using independent component analysis. *Proceedings of the IEEE*, 2001.

[KD65] Hans Helmut Kornhuber and Lder Deecke. Brain potential changes in voluntary and passive movements in humans: readiness potential and reafferent potentials. *Pflugers Archiv: European Journal of Physiology*, 1965.

[KSH12] Alex Krizhevsky, Alex Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems*, volume 1, pages 1097–1105, 2012.

[LCL+02] F Lotte, M Congedo, A Lecuyer, F Lamarche, and B Arnaldi. A review of classification algorithms for eeg-based brain-computer interfaces. *Journal of Neural Engineering*, 2002.

[LS98] Hartmut Leuthold and Werner Sommer. Postperceptual effects and p300 latency. *Society for Psychophysiological Research*, 1998.

[LSW+16] Vernon J. Lawhern, Amelia J. Solo, Nicholas R. Waytowich, Stephen M. Gordon, Chou P. Hung, and Brent J. Lance. Eegnet: A compact convolutional network for eeg-based brain-computer interfaces. arXiv preprint, 2016.

[Luc12] Stephen J. Luck. *An Introduction to the Event-Related Potential Technique*. MIT Press, 2012.

[MBFK87] P. Mitraszewski, Katarzyna J. Blinowska, Piotr J. Franaszczuk, and Marek Kowalczyk. A study of stability of electrocortical rhythm generators. *Biological Cybernetics*, 56:255–260, 1987.

[MCL16] Mathieu Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. 2016. arXiv preprint.

[MDG+17] Farrokh Mansouri, Katharine Dunlop, Peter Giacobbe, Jonathan Downar, and José Zariffa. A fast eeg forecasting algorithm for phase-locked transcranial electrical stimulation of the human brain. In *Front. Neurosci.*, 2017.

[MG15] Ran Manor and Amir B. Geva. Convolutional neural network for multi-category rapid serial visual presentation bci. *Frontiers in Computational Neuroscience*, 9(146), 2015.

[MKG+16] Soroush Mehri, Kundam Kumar, Ishaan Gulrajani, Rithesh Kumar, Shubham Jain, Jose Sotelo, Aaron Courville, and Yoshua Bengio. Samplernn: An unconditional end-to-end neural audio generation model. arXiv preprint, 2016.

[MSW10] Massimiliano Marcellino, James H. Stock, and Mark W. Watson. A comparison of direct and iterated multistep ar methods for forecasting macroeconomic time series. *Journal of Econometrics*, 2010.

[MWJ+02] Scott Makeig, S Westerfield, T.P. Jung, S. Enghoff, J. Townsend, E. Courchesne, and T. Sejnowski. Dynamic brain sources of visual evoked responses. *Science*, 2002.

[NDR09] Hariharan Nalatore, Mingzhou Ding, and Govindan Rangarajan. Denoising neural data with state-space smoothing: method and application. *Journal of neuroscience methods*, 179 1:131–41, 2009.

[NNS91] Shigeto Nishida, Masatoshi Nakamura, and Hiroshi Shibasaki. Method for predicting an eeg waveform as an aid to the accurate recording of evoked potentials. *Journal of biomedical engineering*, 13 5:433–8, 1991.

[ODZ+16] Aaron van den Ooord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. arXiv preprint, 2016.

[PD95] Walter S. Pritchard and Dennis W. Duke. Measuring ”chaos” in the brain: a tutorial review of eeg dimension estimation. *Brain and cognition*, 27 3:353–97, 1995.

[PRT96] J Pardey, S Roberts, and Lionel Tarassenko. A review of parametric modelling techniques for eeg analysis. *Medical engineering physics*, 18 1:2–11, 1996.

[Reg89] David Regan. *Human Brain Electrophysiology: Evoked Potentials and Evoked Magnetic Fields in Science and Medicine*. Elsevier, 1989.

[SKDKM13] Nima Bigdely Shamlo, Kenneth Kreutz-Delgado, Christian Kothe, and Scott Makeig. Eyecatch: Data-mining over half a million eeg independent components to construct a fully-automated eye-component detector. *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 5845–5848, 2013.

[SSF+17] Robin Tibor Schirrmeister, Jost Tobias Springenberg, Lukas Dominique Josef Fiederer, Martin Glasstetter, Katharina Eggensperger, Michael Tangermann, Frank Hutter, Wolfram Burgard, and Tonio Ball. Deep learning with convolutional neural networks for eeg decoding and visualization. arXiv preprint, 2017.

[SSK+02] AS Shah, Bressler SL, KH Knuth, M Ding, AD Mehta, I Ulbert, and CE Schroeder. Neural dynamics and the fundamental mechanisms of event-related brain potentials. *Science*, 2002.

[SSOG15] Sebastian Stober, Avital Sternin, Adrian M. Owen, and Jessica A. Grahn. Deep feature learning for eeg recordings. arXiv preprint, 2015.

[Str94] Steven H. Strogatz. *Nonlinear Dynamics and Chaos*. CRC Press, 1994.

[SvWK97] Cornelis J. Stam, T. C. A. M. van Woerkom, and R. W. M. Keunen. Non-linear analysis of the electroencephalogram in creutzfeldt-jakob disease. *Biological Cybernetics*, 77:247–256, 1997.

[TH16] Yousef Rezaei Tabar and Ugur Halici. A novel deep learning approach for classification of eeg motor imagery signals. *Journal of Neural Engineering*, 14(1), 2016.

[TKS+03] Wilson Truccolo, Kevin H. Knuth, Ankoor Shah, Steven L Bressler, Schroeder Charles E., and Mingzhou Ding. Estimation of single-trial multi-component erps: Differentially variable component analysis (dvca). *Biological Cybernetics*, 2003.

[VSP+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.

[WHH+89] Alexander H. Waibel, Toshiyuki Hanazawa, Geoffrey E. Hinton, Kiyohiro Shikano, and Kevin J. Lang. Phoneme recognition using time-delay neural networks. *IEEE Trans. Acoustics, Speech, and Signal Processing*, 37:328–339, 1989.

[WLW+15] Chun-Shu Wei, Yuan-Pin Lin, Yu-Te Wang, Tzyy-Ping Jung, Nima Bigdely Shamlo, and Chin-Teng Lin. Selective transfer learning for eeg-based drowsiness detection. *2015 IEEE International Conference on Systems, Man, and Cybernetics*, pages 3229–3232, 2015.

[ZW06] Eric Zivot and Jiahui Wang. *Modeling Financial Time Series with S-PLUS*. Springer-Verlag, 2006.