BREAKING DOWN THE BARRIERS TO OPERATOR WORKLOAD ESTIMATION: ADVANCING ALGORITHMIC HANDLING OF TEMPORAL NON-STATIONARITY AND CROSS-PARTICIPANT DIFFERENCES FOR EEG ANALYSIS USING DEEP LEARNING

DISSERTATION

Ryan G. Hefron, Major, USAF

AFIT-ENG-DS-18-S-012

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENG-DS-18-S-012

BREAKING DOWN THE BARRIERS TO OPERATOR WORKLOAD ESTIMATION: ADVANCING ALGORITHMIC HANDLING OF TEMPORAL NON-STATIONARITY AND CROSS-PARTICIPANT DIFFERENCES FOR EEG ANALYSIS USING DEEP LEARNING

DISSERTATION

Presented to the Faculty

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy in Computer Science

Ryan G. Hefron, B.S., M.S.

Major, USAF

September 2018

AFIT-ENG-DS-18-S-012

BREAKING DOWN THE BARRIERS TO OPERATOR WORKLOAD ESTIMATION: ADVANCING ALGORITHMIC HANDLING OF TEMPORAL NON-STATIONARITY AND CROSS-PARTICIPANT DIFFERENCES FOR EEG ANALYSIS USING DEEP LEARNING

Ryan G. Hefron, B.S., M.S.
Major, USAF

Committee Membership:

Brett J. Borghetti, PhD
Chairman

James C. Christensen, PhD
Member

Christine M. Schubert Kabban, PhD
Member

ADEDJI B. BADIRU, PhD
Dean, Graduate School of Engineering and Management

AFIT-ENG-DS-18-S-012

Abstract

Accurate assessment of operator mental workload in the flight environment could enable a myriad of benefits including improved training and safety. Typically when measuring operator functional state, a variety of psychophysiological features are collected and aligned with the associated operator state at a given time. Then signal processing and machine learning techniques are used to train models and make predictions about the operator's state based on the sensor data. The electroencephalograph (EEG) is a useful mental workload sensor because it directly measures brain state, is portable, and provides excellent temporal resolution. However, several barriers exist to using EEG data for workload assessment in operational settings. This research focuses on two barriers: 1) Temporal non-stationarity in feature-to-target mappings when using EEG data, commonly known as day-to-day variability, 2) The myriad of individual differences which lead to cross-participant applicability challenges–a model trained on one individual may not work well for another. Several signal processing techniques and deep learning approaches are developed and evaluated in multi-task environments which account for temporal, spatial, and frequential data dependencies. Application of these techniques and approaches yield the following findings: 1) Incorporating a new feature–variance of Power Spectral Density (PSD) distributions for cross-day workload classification significantly improves classification accuracy. 2) PSD skewness and kurtosis are not significant in an environment absent of workload transitions, but are salient when workload transitions are present. 3) Long Short-Term Memory (LSTM) networks decreased classification error by 59% compared to the previously best results in the presence of day-to-day non-stationarity. 4) A convolutional-recurrent model using multi-path subnetworks and bi-directional,

residual recurrent layers results in significant increases in predictive accuracy and decreases in cross-participant variance.

Additionally, deep learning regression approaches are applied to a complex, multi-task, remotely piloted aircraft simulated environment with arbitrary mission workload transitions. Methods which account for temporal dependence significantly reduce workload estimation error and increase correlation between predictions and target values compared to baseline methods. Finally, visualization techniques for LSTM feature saliency are developed to understand model biases with respect to EEG analysis.

Acknowledgements

Thank you Dr. Borghetti for your advice, mentorship, and open door over the past few years. I enjoyed our conversations, especially when discussing new ideas, and appreciate the countless hours of editing and feedback you provided. I also want to thank Dr. Schubert Kabban and Dr. Christensen for your suggestions and feedback regarding experimental design, statistical rigor, and interpretation of results. To my mentors and colleagues at QuEST, thanks for opening up new intellectual horizons and for supporting interesting side projects. To my friends–you know who you are–its been a great time, thanks for making this fun. Most of all I would like to thank my family, and especially my wife and children, for their endless support, patience, and love.

Ryan G. Hefron

Table of Contents

Page

Page

List of Figures

List of Tables

Table Page

List of Abbreviations

2L-LSTM two-layer LSTM

AEP Azimuthal Equidistant Projection

AUC Area Under Curve

ANN Artificial Neural Network

BCI Brain Computer Interface

BDLSTM Bidirectional LSTM

BDRLSTM Bidirectional ResNet LSTM

CNN Convolutional Neural Network

CDF Cumulative Distribution Function

DBN Deep Belief Network

DLSVM Deep Learning Using Linear Support Vector Machines

ECG electrocardiogram

EEG electroencephalograph

ELU Exponential Linear Unit

EMG electromyograph

EOG electrooculograph

ERD Event-Related Desynchronization

ERP Event-Related Potential

ERS Event-Related Synchronization

ESN Echo State Network

FFT Fast Fourier Transform

fNIRS functional Near-Infrared Spectroscopy

FWNN Fuzzy Wavelet Neural Network

GPU Graphics Processing Unit

GRU Gated Recurrent Unit

GSR Galvanic Skin Response

HMM Hidden Markov Model

HMT Human-Machine Team

HEOG Horizontal Electrooculograph

HSD Honest Significant Difference

HVT High-Value Target

ICA Independent Component Analysis

IID Independent Identically Distributed

ILSVRC ImageNet Large-Scale Visual Recognition Challenge

IMPRINT Improved Performance Research Integration Tool

ISR Intelligence, Surveillance, Reconnaissance

KNN K-Nearest Neighbors

LDA Linear Discriminant Analysis

LSTM Long Short-Term Memory

MATB Multi-Attribute Task Battery

MLP Multilayer Perceptron

MPCRNN Multi-Path Convolutional Recurrent Neural Network

MSE Mean Squared Error

OFSA Operator Functional State Assessment

PCA Principle Component Analysis

PSD Power Spectral Density

RBF Radial Basis Function

RBM Restricted Boltzmann Machine

ReLU Rectified Linear Unit

ResNet deep residual network

RF Random Forest

RMSE Root Mean Squared Error

RNN Recurrent Neural Network

ROC Receiver Operating Characteristic

RPA Remotely Piloted Aircraft

RSEFNN Recurrent Self-Evolving Fuzzy Neural Network

RWENN Recurrent Wavelet-Based Elman Neural Network

SA Situational Awareness

SAE Stacked Autoencoder

SMOTE Synthetic Minority Over-sampling Technique

SNR Signal to Noise Ratio

SONFIN Self-Organizing Neural Fuzzy Inference Network

STFT Short-Time Fourier Transform

SVM Support Vector Machine

SWDA Stepwise Discriminant Analysis

TLX Task Load Index

TRFN TSK-Type Recurrent Fuzzy Network

UAV Unmanned Aerial Vehicle

VACP visual, auditory, cognitive, psychomotor

VEOG Vertical Electrooculograph

BREAKING DOWN THE BARRIERS TO OPERATOR WORKLOAD ESTIMATION: ADVANCING ALGORITHMIC HANDLING OF TEMPORAL NON-STATIONARITY AND CROSS-PARTICIPANT DIFFERENCES FOR EEG ANALYSIS USING DEEP LEARNING

I. Introduction

In the first decade of the 21st century, 13 of 18 commercial aircraft accidents attributed to in-flight loss of aircraft control were caused by crew loss of Situational Awareness (SA) [72]. The causes of loss of SA were identified as diverted attention, often due to high workload, and channelized attention, caused by focusing exclusively on an instrument or other stimuli at the expense of not processing other vital input channels [72]. These causes are not unique to civilian flight, and are even more prevalent in the military flight environment where pilots additionally operate a host of other systems including sensors, weapons, and electronic warfare systems.

Clearly, maintaining a consistently high level of performance is a requirement in these environments. To do so, it is important to maintain an operator's mental workload at a manageable level which minimizes the likelihood of a performance breakdown. This has long been an objective of human performance research rooted in the fields of human factors, autonomy, and human-machine teaming. These in turn have driven research into Operator Functional State Assessment (OFSA) [82].

Hockey defines operator functional state as, "The variable capacity of the operator for effective task performance in response to task and environmental demands, and under the constraints imposed by cognitive and physiological processes that control and energise behavior" [82]. This definition highlights that the operator has a

changing capacity to respond to imposed demands and still attain a level of performance. When piloting an aircraft, this capacity is subjected to psychophysiological constraints associated with controlling the aircraft at a desired level of precision while listening and responding to internal communications in the cockpit and external communications with air traffic control and other aircraft. These tasks require cognitive processing associated with decision making and action which in turn increase operator workload.

Due to the Air Force interest in the flight environment and the potential impacts of increased workload with regard to performance, this work is focused on a segment of OFSA–estimating operator workload. In their 1988 report on the development of the NASA-TLX Hart and Staveland define workload in the following manner, "Workload is a hypothetical construct that represents the cost incurred by a human operator to achieve a particular level of performance" [73]. They continue to describe that workload is not an implicit property associated with a given task, but rather is the result of an interaction between task requirements, the environment in which the task is being performed, as well as operator behavior, skills, and acumen [73]. Changes in any of these factors can lead to elevated workload levels for a given operator. Stated differently, workload can be viewed as the operator's realization of the result of a feedback loop between the system and the operator which is associated with a desired level of performance. Hart and Staveland delineate two types of effort required of the operator: physical and mental efforts [73]. Cooper and Harper also recognized these physical and mental efforts when defining workload in the context of the flight environment as "The integrated physical and mental effort required to perform a specified piloting task" [45]. Examining these definitions of workload, it is clear that there are both physical and mental components to workload. Furthermore, there is an interplay between the operator, the task, desired performance levels, and

the environment, modulated by operator characteristics including behaviors, experience, and other factors. High fidelity OFSA likely requires measurement of both psychophysiological and behavioral elements to model both the physical and mental components of workload. This work focuses on a component of this broader challenge: assessing mental workload using electroencephalograph (EEG) signals in a multi-task environment with novel deep learning and signal processing techniques.

1.1 Why EEG?

Typically when measuring operator functional state, a variety of psychophysiological features are collected: EEG to monitor brain activity, EOG for inter-blink and blink duration, heart rate and heart rate variability to capture sympathetic and parasympathetic responses to changes in workload, and respiration, among others. Generally, EEG provides the best temporal resolution compared to the other metrics [101]. Because temporal resolution is of utmost importance for the operational aircraft setting–due to the dynamic and safety critical nature of flight–this work focuses on examining EEG signals. EEG is also the most portable and least intrusive sensor which measures brain activity. Furthermore, EEG is information-rich, and has the highest potential of the psychophysiological measures to enable classification accuracies commensurate with the requirements for use in a military aircraft setting as shown by its high feature salience across both workload studies and studies in other fields as discussed in Chapter II.

EEG as a measure of brain function has a long history of use as a tool for diagnosis in the medical field [137]. With the advent of high performance mobile computing, EEG use in non-medical applications has blossomed as Brain Computer Interface (BCI) technologies were developed. Machine learning algorithms are a major component in these systems and are used to analyze the complex signals. There are

numerous supervised machine learning techniques used, starting with basic linear regression and progressing in complexity to Support Vector Machines (SVMs) and deep neural networks. In recent years, deep neural networks have been used to produce best-in-class results across a large number of applications including speech recognition, translation, image captioning, and most recently EEG analysis [65, 66, 167, 14]. A host of literature suggests that representational learning using deep neural networks results in more discriminative features and better overall performance of classification and regression systems than shallow techniques–those which do not use a learned, deep feature representation [103]. Despite a large body of researchers using EEG to assess workload, relatively few have taken a deep neural network approach to model workload and none have adequately addressed the temporal, spatial, and frequential dependencies inherent in oscillatory neuronal signaling as measured by EEG. These challenges are discussed in the next section. Because of its information rich content, the challenges associated with EEG processing, and the burgeoning field of deep learning, OFSA using EEG as a signal source appears ripe for research improving the algorithmic handling of these signals.

1.2 Challenges

There are three primary challenges that must be overcome to fully realize the benefits of using EEG as a functional state assessment tool. These challenges apply equally across all OFSA domains and must be addressed to produce high-quality models. Until these significant challenges can be adequately handled, it is unlikely that EEG will be useful in an operational aircraft setting. The three primary challenges are defined as:

1. Day-to-day variability: Feature distributions are not stationary over time, causing single models that are trained across several days to have lower classification

accuracy than models created for each individual day.

2. Cross-participant differences: Models trained on a group of participants tend to underperform individually-tailored models by a significant margin due to individual variation.

3. Cross-task applicability: Models that are trained on one task do not necessarily generalize to another task, but may if the tasks are similar enough. A continuum of similarity between tasks seems to exist that may be exploitable if the right features are learned.

This dissertation focuses on addressing the first two challenges by improving workload modeling in complex, multi-task environments where day-to-day variability and cross-participant differences are present.

1.3 Benefits

If progress can be made on these challenges by systematically addressing some of the algorithmic barriers to operational use of an OFSA system in the laboratory environment, the field may be advanced to a state where better results can be obtained in more realistic operational environments. There are many potential benefits an operational system capable of outputting an objective operator workload estimate would have in the military flight environment. Better wingman utilization would be possible if a flight lead has a workload index for each of his wingmen. Improvements in training for single-seat fighter aircraft could occur by allowing the instructor to know the state of the student without being in the same cockpit as the trainee. Several studies have expressed an expected reduction in aircraft accident likelihood [50, 140, 31]. Wilson and Russell identified that an objective measure of workload for test and evaluation purposes would be quite beneficial for human factors evaluations

[178]. Several physiological states can be nebulous to a pilot operating an aircraft that an operator functional state assessment system could clearly identify. Among these are identification of fatigue [26] during long-duration sorties and detection of imminent G-induced loss of consciousness (G-LOC) [175]. Ultimately, such technology could pave the way for advances in adaptive automation and improve human performance by machine augmentation [125]. However, most OFSA research has been limited to the laboratory due to the array of challenges associated with implementing such a system. This work focuses on the challenges of applying OFSA techniques that may be transitionable to operational settings and, in particular, to the military flight environment.

1.4 Approach

All analysis in this dissertation is focused on improving workload modeling using EEG for complex, multi-task environments. Chapter II explores the state of the art of the field, identifies strengths and weaknesses of current research highlighting gaps that this research addresses, and provides contextual information essential to developing an understanding of the methodology in later chapters. This sets the stage for the research methods and results presented in a series of papers which comprise Chapters III - VI. Overviews of the multi-task environments, the Multi-Attribute Task Battery (MATB) and the Air Force Vigilant Spirit Control Station, are discussed in the dataset or methodology sections of Chapters III - VI. These environments contain aspects representative of complex real-world tasks, so progress on them should yield meaningful methods to improve model accuracy in real-world conditions.

Many traditional machine learning and some deep learning techniques have been applied to analyze EEG data, yet further research is required to improve predictive accuracy if a workload sensing system is to be used successfully in a real-world,

in-flight setting. To improve accuracy, one can collect more data, generate better features, or select a more appropriate model for the machine learning task. Gathering more data can be time consuming and costly. Analysis of a given problem should inform whether more data is required, so that subject is not further discussed.

There are several ways to generate better features. Statistical or mathematical methods can be used to engineer new features from existing ones, a hierarchy of features can be learned using deep learning techniques, or measurements of new phenomena can be taken. In this dissertation, feature engineering is used to produce features with improved temporal stationarity. These results are reported in Chapters III and IV where new statistically significant features are developed for psychophysiological workload estimation. A host of literature suggests that representational learning results in more discriminative features and better overall performance of classification and regression systems than shallow techniques [103]. Specific methods and results obtained applying deep learning concepts to learn better feature representations are discussed in Chapters IV - VI. Background regarding the remaining method, measuring new phenomena, is discussed in Appendix A. However, its application is relegated to future work.

The last way considered in this work to improve predictive accuracy is by selecting a more appropriate model for the task. While there are a limited number of classification or regression forms, deep learning techniques currently afford a richer variety of capacity tuning and regularization methods than other traditional machine learning approaches and are well-suited, in conjunction with ensemble methods, for handling the challenges associated with EEG analysis. They also enable the network architect to impart domain-specific assumptions and priors onto the classification or regression model rather than being constrained by implicit assumptions, such as normality, that may be misaligned with the problem domain. Chapters IV - VI discuss applicable

advances associated with developing and selecting more appropriate models. Finally, Chapter VII presents conclusions, including contributions and future work.

1.5 Summary of Research Objectives, Contributions, and Findings

In this section, research objectives, contributions, and findings are presented for four primary studies which are contained in Chapters III - VI (studies A, B, C, and D respectively). For each study, a table summarizing research objectives, contributions, and findings are provided with a short paragraph providing context. The contributions and findings summarized in Tables 1, 2, 3, & 4 directly map to their expanded versions in Chapter VII where a full discussion enumerating the contributions and findings is present.

Table 1. Summary of objectives, contributions, and findings for study A.

	Objective	Section
AO1	Develop a feature engineering method which is more resilient to the day-to-day variability of EEG data	3.5
AO2	Identify salient features from cross-day random forest and Linear Discriminant Analysis (LDA) models	3.5
	Contributions and Findings	
A1	Variance of Power Spectral Density (PSD) distributions improves cross-day workload classification accuracy by 5.8% above models using only mean power	3.5
A2	Temporal gamma oscillations are salient group features across participants in multi-task environments, but significant variation persists at the individual level	3.5

The first contributions appear in Chapter III where new feature generation techniques are explored which result in improvements in predictive capability in multi-task settings [77]. Section 3.5 shows that the variance of frequency-domain power distributions for cross-day workload classification is statistically significant in improving accuracy in a multi-task environment [77]. Then, feature saliency analysis is used to

show that temporal gamma oscillations are salient across participants, but that significant variation is present at the individual level (Section 3.5) [27, 77, 101, 148, 185].

Table 2. Summary of objectives, contributions, and findings for study B.

	Objective	Section
BO1	Using deep Recurrent Neural Networks (RNNs) which account for temporal dependencies, determine if significantly improved cross-day workload classification accuracy results compared to traditional feedforward neural networks and SVMs	4.4
BO2	Statistically evaluate new feature generation techniques which include all combinations of mean, variance, skewness, and kurtosis of frequency-domain power distributions	4.4
	Contributions and Findings	
B1	Deeply stacked Long Short-Term Memory (LSTM) models account for temporal dependencies in brain activity data reducing classification error by 59% and achieving an overall accuracy of 93.0% for cross-day models	4.4
B2	Mean and variance of PSD distributions were statistically significant features, while skewness and kurtosis were not in multi-task environments absent of workload transitions	4.4

Chapter IV extends the work presented in Chapter III by evaluating skewness and kurtosis of frequency-domain power distributions. Skewness and kurtosis is not found to be statistically significant in a multi-task environment absent of workload transitions [78]. A significant gap in literature is also addressed beginning in Chapter IV: There was a complete absence of research into using deep learning to make predictions using EEG data in multi-task environments. A challenge associated with this gap is addressing temporal non-stationarity associated with day-to-day variability. It is hypothesized that deep learning techniques which take into account temporal context rather than making the assumption of temporal independence may result in superior performance compared to baseline methods. Using deeply stacked LSTM models, a feature representation is learned that improves temporal-stationarity of the feature-to-target mapping resulting in a 58% decrease in workload classification error

compared to baseline methods and a 59% reduction in workload classification error over the best published results for the MATB dataset described in Section 3.3 [78]. The reduction in error also implies that modeling the stateful nature of brain activity may improve feature stationarity.

Table 3. Summary of objectives, contributions, and findings for study C.

	Objective	Section
CO1	Compare four types of cross-subject model training techniques to understand the tradeoffs between ensemble and group-based training methods when using deep learning techniques	5.3.1
CO2	Understand the effect of signal sequence length on cross-participant mean accuracy and variance	5.3.2
CO3	Evaluate deep-neural network architectures which account for spatial, temporal, and frequency dependencies in EEG workload data relative to baseline techniques	5.3.3
	Contributions and Findings	
C1	Ensembles can be trained for a fraction of the computational cost compared to group-training methods, produce statistically indistinguishable results, and can be simply updated as additional participants are added to a study	5.3.1
C2	Increasing temporal sequence length improves mean accuracy, but magnifies the effect of individual differences as it increases cross-participant variance	5.3.2
C3	Multi-path convolutional recurrent networks improve mean accuracy and reduce cross-participant variance, diminishing the effects of individual differences	5.3.3

Having realized significant improvements with regard to the first primary challenge, day-to-day variability, a next logical step is to address cross-participant differences in a multi-task environment. This is done in Chapter V using Multi-Path Convolutional Recurrent Neural Networks (MPCRNNs) to improve cross-participant modeling. This effort represents the first time that multi-path convolutional recurrent networks and bidirectional, residual LSTMs are used to analyze EEG data. Statistically significant improvements in cross-participant generalization result from using the

MPCRNN model in terms of both mean workload classification accuracy (increase) and cross-participant variance (decrease) [76]. Additionally, the effect of varying the temporal feature sequence length shows longer sequences improve mean cross-participant model accuracy, but have a negative effect on variance [76]. This means that longer sequence length cannot singlehandedly solve the challenges posed by individual differences since cross-participant variance increases due to participant-specific distributional differences. The effect of training method for deep neural network approaches is also examined and found to be less important than other factors [76]. Rather, training method should be chosen based on computational cost and experimental design factors.

A limitation present in Chapters III - V is that each trial contains only a single workload level with no transitions present during the trial. This limitation combined with another gap in literature lead to the work present in Chapter VI. The gap in literature is that regression approaches which make predictions using EEG data are very sparse despite the continuous nature of mental states, and that no deep learning regression approaches have been attempted. In Chapter VI, several deep learning and baseline regression approaches for mental workload prediction are evaluated based on RMSE and correlation metrics in a complex, multi-task, Remotely Piloted Aircraft (RPA) Intelligence, Surveillance, Reconnaissance (ISR) simulated environment with arbitrary mission workload transitions. For the first time, Siamese-triplet networks are introduced for EEG analysis and are characterized based on their predictive capability compared to other algorithms. They perform comparably to best-in-class bidirectional LSTMs methods and result in statistically significant improvements in prediction compared to non-temporally aware models. Since Siamese-triplet networks are traditionally used in classification settings, a theoretical description is outlined explaining how and why using a variable margin is useful in a regression setting. Nu-

Table 4. Summary of objectives, contributions, and findings for study D.

	Objective	Section
DO1	Evaluate several deep learning regression methods for estimating cognitive workload	6.5
DO2	Determine if modeling temporal context improves workload estimation in a realistic, multi-task environment with numerous workload transitions	6.5
DO3	Introduce and evaluate Siamese-triplet networks for EEG analysis	6.4.1.4 6.5
DO4	Determine if skewness and kurtosis are salient features in the presence of workload transitions	6.6
	Contributions and Findings	
D1	Bidirectional LSTMs and Siamese-triplet networks significantly outperform methods which make Independent Identically Distributed (IID) assumptions in multi-task environments with many workload transitions	6.5
D2	Siamese-triplet networks perform as well as best-in-class bidirectional LSTMs for workload estimation in terms of Root Mean Squared Error (RMSE) and correlation	6.5
D3	A mathematical formulation for setting a variable margin for Siamese-triplet networks used in regression environments is presented, reducing the need for hyperparameter search	6.4.1.4
D4	A feature saliency visualization technique is introduced enabling rapid discrimination of feature saliency for neural networks with grouped feature types	6.6
D5	Skewness and kurtosis of PSD distributions are salient features in multi-task environments with workload transitions	6.6
D6	LSTMs bias feature saliency based on the depth of the backpropagation path	6.6

merous feature visualization techniques are also used to improve interpretability of EEG feature saliency as a function of feature type and time when using temporally-aware deep learning approaches. Insights from using these techniques imply that bidirectional LSTMs impact results due to their algorithmic assumptions causing greater feature salience at the beginning and end of a sequence of features.

1.6 Dissertation Structure

In Chapter II, literature regarding day-to-day variability and cross-participant differences is examined while identifying causes, manifestations, and potential techniques to handle the associated challenges. Application of deep learning methods to EEG analysis is also discussed in depth and gaps in the literature are identified to situate the contributions in this dissertation. Chapters III - VI comprise four studies and detail their methods, results, and discussion of findings. In Chapter III new feature engineering techniques are introduced to address the day-to-day variability challenge. Chapter IV extends these feature engineering methods and expands the modeling methods used to handle the temporal non-stationarity associated with day-to-day variability to better account for temporal dependence using deep RNNs. Chapter V shifts focus and thoroughly explores the influence of training method, feature sequence length, and neural network architectural considerations which affect cross-participant applicability. Consequently, novel network architectures designed to mitigate the influence of individual differences were developed which incorporated multi-path convolutional recurrent elements. Chapter VI wraps up the studies by applying deep learning and baseline regression approaches for mental workload prediction in an operationally realistic multi-task simulation environment. Additionally, Siamese-triplet networks are introduced and characterized based on their predictive capability using EEG data and were compared to other algorithms. Finally, Chapter VII summarizes the contributions and findings and presents three paths for future research.

II. Background

In this chapter, the underlying factors influencing the major electroencephalograph (EEG)-specific challenges of cross-day and cross-participant variability are elucidated, and their manifestations are examined in light of others' research. The chapter begins relevant literature focused in the areas of day-to-day variability and cross-participant applicability. Next, background on constructing deep neural network architectures to learn useful feature representations is detailed in Section 2.3 where Recurrent Neural Networks (RNNs), Convolutional Neural Networks (CNNs), Deep Belief Networks (DBNs) and Stacked Autoencoders (SAEs) are introduced. This is followed with a thorough review of state-of-the-art deep learning approaches to any EEG analysis, since little research has been conducted applying deep learning approaches specifically to psychophysiological workload estimation. Additionally, since the target application for this work is in-flight workload estimation, an overview of in-flight and high-fidelity simulator research using EEG is available in Appendix B. All EEG studies examined in this chapter are summarized in Table 5.

2.1 Day-to-day Variability

A non-stationary environment is one where there is a change in feature-to-target mapping over time, $P(y|x)$, where P is the probability of y, the dependent or target variable, given x, the independent variable or feature vector [58]. This type of non-stationarity is also known as concept drift [58]. Christensen et al. [39] showed that day-to-day variability is caused not only by classifier generalization challenges such as overfitting, but also due to a change in the mapping between the feature and target distributions over time: The phenomena of day-to-day variability in EEG signals manifests itself as a non-stationary environment [39]. Because of this non-stationarity,

Table 5. Summary table of related EEG studies

	Algorithm of Interest											Challenge			Fusion			Task																				
Citation	RNN	CNN	Autoencoder	RBM	DBN	Single-hidden-layer ANN	SVM	LDA	RF	Naive Bayes	Ensemble	Cross-Subject	Cross-Task	Day-to-day Variability	Early Fusion	Late Fusion	Data Augmentation	Emotion Classification	Motor Imagery	Epilepsy/Seizure Detection	Movement-related cortical potentials	ERP	Musical Rhythm Detection	Driver Perfromance	Driver Fatigue	MATB	In-flight Workload	Flight simulator Workload	N-back	UAV Control Workload	Battery of Tasks	Reading Span	Sternberg	Tracking Task	Olfactory Stimuli	Neurodegeneration	Cognitive Linking	Air Traffic Control
[106]	X	X	X														X	X																				
[76]	X	X				X					X	X														X												
[14]	X	X			X		X		X			X					X																					
[164]	X	X										X								X																		
[78]	X					X	X							X												X												
[112]	X						X				X	X													X													
[113]	X						X					X													X													
[148]	X						X								X	X		X																				
[69]	X											X								X																		
[166]	X											X								X																		
[100]	X											X								X																		
[149]	X											X								X																		
[48]	X											X																						X				
[22]	X																		X																			
[7]	X																				X																	
[136]	X																																		X			
[133]	X																																			X		
[118]	X																																				X	
[158]		X	X				X												X																			
[71]		X		X								X												X														
[161]		X					X												X																			
[120]		X					X													X																		
[153]		X					X															X															
[102]		X										X							X		X	X																
[180]		X																	X																			
[168]		X																	X																			
[91]			X				X					X						X																				
[90]				X	X		X											X																				
[70]					X		X	X																	X													
[185]					X		X					X						X																				
[186]					X		X											X																				
[8]					X		X												X																			
[15]					X		X																										X					
[165]					X							X								X																		
[130]					X														X																			
[93]						X	X	X				X					X					X																
[39]						X	X	X						X	X											X												
[101]						X						X			X											X												
[178]						X						X														X												
[179]						X						X																		X								
[61]						X							X	X															X									
[177]						X							X		X																							X
[11]						X							X																X			X	X					
[123]						X								X	X												X											
[32]						X								X			X									X												
[176]						X								X												X												
[33]						X																				X												
[77]							X	X	X		X			X												X												
[114]							X				X			X					X																			
[95]							X						X													X			X									
[94]							X						X																X									
[56]								X			X	X							X																			
[109]										X				X				X																				
[128]												X			X																X							
[62]													X	X															X									
[25]														X												X		X										
[162]														X												X												
[89]																	X																					
[152]																											X											
[174]																											X											
[151]																											X											
[85]																											X											
[141]																											X											
[31]																											X											
[10]																											X											

one considerable challenge in the field of workload estimation using EEG signals is achieving high classification accuracy across multiple sessions spanning several days for a given individual. Generally, training a single workload model across several days leads to lower classification accuracy than creating a new model for each day. The high inter-day variance associated with these signals is a significant barrier to utilization of psychophysiological signals in operational environments.

The lack of cross-day generalizability of a classifier likely has many causes that may be nearly impossible to disentangle. However, it may be possible to adjust for some factors via normalization and account for others by learning new features that are invariant in time. Jahns [87] identified individual motivation as well as mental and physical readiness as factors related to the participant's mental state that contribute to inter-day variability. Confounding factors such as fatigue, experience, and emotional factors can all have an effect on EEG signals [26]. Christensen, et al. [39] identified changes due to circadian rhythm as another source of day-to-day variability. Changes in sensor properties such as conductance and variation in placement of electrodes are also causes of non-stationarity, while differences in artifacts from one day to another are additional sources of variation [114]. A final consideration is that people continuously learn while performing tasks, and learning manifests itself in changes to neuronal activity [25, 62, 162]. Therefore, the effects of practice and proficiency are also significant causes of temporal non-stationarity. These factors are controlled as much as possible in laboratory experiments, yet significant variation remains. While this list is not exhaustive, it illustrates the problem: How can a stable model be built when there are so many sources of temporal non-stationarity?

When analyzing EEG data, it is the temporal dynamics of the signals that are interesting, and it is important to distinguish between three temporal lengths when using time-frequency analysis [39, 42]. The millisecond-to-millisecond or second-to-

second time scale is important because at this scale, stationarity is assumed due to analytical requirements. Techniques such as the Short-Time Fourier Transform (STFT) or complex Morlet wavelet convolution which pick out the power of rhythmic activity at particular centered frequencies within a specific window require stationarity over the length of the window [42]. Changes at this time-scale form the foundation of the features used to build models for EEG. It is also reasonable to expect that the numerous sources of classifier-degrading temporal non-stationarity do not have significant effects at these time scales. The second temporal length is on the order of seconds to minutes. When machine learning techniques are used for classification of EEG signals, it is hoped that the distribution of features associated with signals for a particular individual, at a particular time, doing a certain task, is consistent. If not, at least to a certain degree, then no patterns will be found and predictions will be no better than chance, on average. This degree of stationarity is what is usually observed within a single testing session, assuming that learning is not occurring [39]. Changes over a timescale associated with a single session are what interest most researchers. The final degree of temporal non-stationarity manifests in hours to days across testing sessions [39, 32]. This is the problematic temporal non-stationarity which leads to poor model performance due to concept drift and is the target of improved algorithmic handling in this work.

Several researchers have performed experiments to both better understand and to address some of the challenges associated with temporal non-stationarity. While the effects of practice are a source of non-stationarity, they are not discussed here because the experimental procedures used to collect the data used in this dissertation were designed to mitigate the effects of practice. Rather, studies are examined that either characterize or address some of the problems associated with day-to-day variability due to other sources of non-stationarity.

2.1.1 Variations in Feature Distributions.

Christensen et al. evaluated cross-day classification performance of low versus high workload by training three classifiers: Linear Discriminant Analysis (LDA), Support Vector Machine (SVM), and a single-hidden-layer feedforward Artificial Neural Network (ANN) [39]. The experiment had eight participants who each performed Multi-Attribute Task Battery (MATB) tasks across five test days spread out over a month-long period. The data was time-synchronized across 19 channels of raw EEG, Horizontal Electrooculograph (HEOG), Vertical Electrooculograph (VEOG), and electrocardiogram (ECG) all sampled at 256 Hz. On each of the five days, each participant performed three five minute trials at low, medium, and high workload for a total of nine trials a day presented in a random ordering with transition periods in between. Only the low and high workload data were used to train and test their classifiers. Christensen, et al. trained on all possible combinations of 1 day, 2 days, 3 days, and 4 days while testing on the remaining days not included in the training data and averaged accuracies for each of these cases to determine how the different algorithms' cross-day classification performance changed as a function of the number of training days. The ANN performed the best and appeared able to incorporate data from multiple days to increase classification accuracy; showing a rise from approximately 73% to 83% accuracy between the 1-day training set and the 4-day training set. The results of the other two classifiers across multiple days were less successful compared to the ANN. The SVM method garnered no significant performance improvement when additional training days were added, while the LDA method only experienced minor improvement. They also evaluated classification accuracy as a function of the distance between the training data and the test data and saw a monotonic decrease in accuracy as temporal distance increased from seconds, to minutes, to hours, to days; although there was a slightly less pronounced decrease between the hours and days

categories [39]. This means the effects of non-stationarity become more pronounced as the temporal length between the training data and the test data increases. Christensen proposed that future work could use a small amount of calibration data at the start of each day as a possible way to adjust classifiers to account for the diverse sources of non-stationarity.

In a study very similar to Christensen's, Casson evaluated the effect of temporal stability of feedforward neural network classifiers trained on EEG signals with differences of seconds, minutes, hours, and days between collection of training data and testing data using the Cognitive State Assessment Competition dataset [32, 54]. He found that accuracy decreased monotonically from 86%, when training and testing sets were only separated by seconds, to 57% when spanning days [32]. Noise-enhanced testing was also attempted to see if better generalization would result. This involved adding Gaussian noise to the testing data to produce multiple instances of each condition. Then each instance was classified and the ensemble's majority class was used as a final prediction. The performance did not result in statistically significant improvements compared to baseline cases [32].

Lin, Hsu, and Jung conducted a cross-day study of emotion classification using EEG signals to evaluate the temporal stability of classifiers trained for this emotional application [109]. Twelve individuals had EEG data gathered over five separate test days while listening to several musical excerpts that had been previously labeled (outside of this study) as either happy or sad. Each individual labeled their elicited emotional response to the musical except as either happy, sad, or neutral. A Naive Bayes classifier was trained for each individual and differing combinations of days were used to classify the trials. The best results were obtained using within-day training. Using an optimal combination of two-day data (selected in a post hoc fashion), resulted in an increase in classification accuracy by approximately 3% [109]. However, significant

degradation in classification accuracy was observed with increased variability in the EEG feature distributions from one day to another, and as more days were included in the training set, the classification accuracy further decreased [109]. The authors theorize that increasing the database size and using an advanced normalization procedure may improve results [109].

Not all studies have found adverse effects associated with cross-day testing. Gevins, et al. conducted a workload study associated with working memory tasks [61]. A single-hidden-layer feedforward neural network was used to train and classify the data and a forward step-wise feature selection procedure was used in conjunction with the neural network to only select a salient subset of features. One component of this study was a final test date a month after the previous session to evaluate stability of the classifier over time. For those test subjects who returned a month later, their classification models appeared to be temporally stable with accuracy on the final testing day averaging 95% for the two-class problem (high and low workload) compared to 94% from a month earlier. The temporal dimension of this study demonstrates that the results were not significantly affected by up to a one-month break from task performance. However, the tasks were very well defined, designed to be completely reproducible, and did not require a high level of skill specialization to perform [61]. Because of this, these results may not extend to real-world, job-related tasks which evoke more complex and varied responses. The result of this study in combination with others in this section suggest that as task complexity increases, inter-day variability is more likely to occur and to be a significant factor needing attention during analysis. The reduction in feature space attained by using a forward stepwise procedure also likely contributed to classifier temporal stability.

Noel, Bauer, and Lanning tackled the cross-pilot, cross-day workload classification problem that is present in many in-flight physiological studies. Cross-pilot results are

discussed in Section 2.2. This is the only flight test study which examined cross-day factors. The experiment involved 10 pilots flying a Piper Cub aircraft along a pre-planned route constructed to have three distinct levels of mental workload– low, medium, and high. However, a major shortcoming of the study was that only 2 of 10 pilots' data was initially available for their study with a third pilot's data becoming available later. All reported results are only based on two pilots and the third pilot's data was used as a validation set. Each pilot flew the same route on two separate days. A neural network was used to analyze the psychophysiological signals and create a classifier for low, medium, and high workload. Futhermore, a signal-to-noise (SNR) screening method was used for feature reduction and saliency determination [123]. Noel found that most salient features differed across days for a given pilot and that multi-day salient features differed from single day sets [123]. After visual examination of the data, Noel determined that 4 features were relevant across days: HR, HRV, number of eye blinks, and blink duration. Using a linear combination of these features as input into an ANN, a model was developed which significantly outperformed a baseline model which included the four aforementioned features and 35 other salient features including EEG features. Classification accuracy of their baseline model across days was only 60% while the linear combination of the four features model resulted in a classification accuracy in excess of 80%. Using the third pilot's data as a validation set, the baseline model only achieved 57% accuracy while the linear combination model increased to 72% [123]. These results further highlight the challenges associated with non-stationarity of EEG signal distributions and indicate that new approaches may be necessary to appropriately handle these variations especially in real-world tasks.

Liyanage, et al. developed a dynamically weighted ensemble of classifiers to classify EEG data collected from two separate data sets. The first dataset had nine

participants performing four different motor imagery tasks across two separate days while the second motor imagery data set was produced by 12 individuals collected over two days [114]. The authors noted significant non-stationarity in the feature distributions from one session to the next. They trained multiple SVM classifiers based on different clusterings of features from the first training session for each participant. A weighted majority voting system was used to determine test classes and to effectively forget irrelevant data for given conditions [114]. The weighting was influenced by the estimated distance to cluster centers and classification accuracy for a given classifier in the ensemble [114]. Their results showed mean classification accuracy improved to 81.5% from 75.9% compared to a baseline SVM classification method that did not account for non-stationarity of the data sets [114].

Several trends were noted during examination of the literature that evaluates day-to-day variability. All models assumed temporally independent features which is a poor assumption for modeling stateful brain activity. In Chapter IV results are presented which indicate that accounting for temporal dependencies in the EEG signals results in significantly improved feature stationarity in cross-day complex task scenarios. Next, only Liyanage, et al. [114] conducted cross-day research which accounts for sources of temporal non-stationarity by adapting the weighting of an ensemble or using some other dynamic method to account for the non-stationarity. In Chapters IV - VI, the Long Short-Term Memory (LSTM) is used to model temporal context which has mechanisms capable of learning how to best improve predictive performance in the presence of limited non-stationarity. A final takeaway is that calibration data for domain adaptation may further improve results in a non-stationary environment. Next, the state of cross-participant research is assessed.

2.2 Cross-participant Applicability

Another finding requiring attention is that models trained on a group of participants tend to underperform individually-tailored models by a significant margin. Both physical and psychological differences exist between individuals which make cross-participant applicability a challenge. Cohen indicated some of the difficulties of cross-participant research are related to physical setup differences across participants [42]. The positioning of electrodes often varies slightly across individuals due to standard template positions being used rather than precise locations. This can cause some variation between participants. Furthermore, there are differences in skull and scalp electrical conductances between individuals and even within individuals depending on location and bone structure which are difficult to account for and measure [42]. On the other hand, Jahns identified several cognitive and psychological factors which are associated with cross-participant variability including experience, background, and personality traits [87]. Unfortunately, it is impossible to identify causation for cross-participant variability due to the number of confounding factors. However, based on results from using ensembles of classifiers and from several deep learning studies, it may be possible to significantly improve cross-participant classification accuracy. Before considering ensembles and deep learning techniques, more conventional attempts at cross-participant classification are first examined.

Wilson and Russell [178] conducted a study where 7 participants performed NASA's MATB testing (tracking w/mouse and joystick, monitoring lights/dials, and talking). Three different task levels were created: baseline, low workload, high workload. Five-second epochs were classified in real-time after initial trials were completed at each workload setting and were later classified offline after the testing was complete. For this test, the network was retrained incorporating all the data from the initial runs. Wilson and Russell then compared the offline to the real-time analysis and noted

very similar results. The online accuracies were 82.0% and 86.0% for low and high workload respectively while offline analysis was slightly improved at 87.4% and 89.2% respectively. Of interest, they determined the relative contribution of the different inputs/features from each individual to their particular neural networks and found the weighting and selection of the different features varied widely among all participants [178].

Cross-participant feature saliency was also considered by Noel. A description of the experimental setup was already described for Noel's study on pilot workload classification in Section 2.1.1. Noel found a drastic difference in salient features between the two pilots–one pilot had 36 salient features, while the other only had 6 [123]. Upon further analysis, it was determined that different pilots react in different ways to imposed workload or stress [123]. This indicates that cross-participant applicability of a group model may perform significantly worse than the individually tailored models.

Wilson and Russell [179] followed up their previous experiment with one which used a simulated Unmanned Aerial Vehicle (UAV) attack scenario where each participant had to simultaneously operate four air vehicles and use them to locate and designate targets using predefined rules of engagement. Task difficulty was varied based on the complexity of radar images the participant had to search to find targets. More complex images contained a larger number of non-target distractors in the image. Four easy and four difficult images were presented for each condition. The participant had to select and mark six targets for bombing based on the radar image before the UAV reached the target area. The speed of the UAV was varied to identify where the workload became so high that significant performance degradation became present. NASA-TLX was used to give subjective estimates of operator workload after each mission for each particular radar image. The goal of the project was twofold:

To demonstrate that adaptive aiding based on psychophysiological features would improve performance relative to randomly applied aiding, and to evaluate the impact of using a group defined high workload setting versus individually tailored thresholds. Adaptive aiding resulted in a 50% increase in the performance criteria of successful weapon releases than in the un-aided scenario for previously determined individually tailored high workload settings. The group defined high workload setting was not produced by grouping all the participant's runs together to train a new ANN; rather, each already trained individual ANN was used to determine a threshold UAV speed for that individual which resulted in high workload. These UAV speeds were then averaged together to obtain a group threshold speed for high workload. Performance using aiding with the group-determined mean resulted in an improvement in performance of 35% which was a decrease compared to the individually tailored results [179]. Wilson and Russell's results indicate that using a mean high workload threshold for a group of operators does not appear to be a useful grouping method because those individuals whose performance falls below the mean continue to perform poorly while those whose performance is better than the mean have excess capacity.

Laine, et al. [101] conducted a study to determine if a small set of salient psychophysiological features could be identified for accurate cross subject generalization for a MATB task. A secondary objective was to use a non-individualized group model to classify mental workload for all participants. In essence, Laine tried to identify an independent stand-alone set of features that worked for training ANNs for all individuals in the study and then tried to determine if one classification model could be used across all subjects to identify workload without modifying ANN model parameters. Laine, et al. [101] performed Stepwise Discriminant Analysis (SWDA) for feature selection and used a Signal to Noise Ratio measure to determine feature importance for the ANN. They identified Pz-gamma as the most salient feature and provided a

rank-ordered list of features for each individual and for the group as a whole. Parsimonious sets of features for individuals varied significantly from one person to the next; however, a set of group features was obtained that worked well. The group results were dominated by features in the gamma band. For individuals, interval of eyeblink data and EEG power in the gamma, beta, and alpha frequency bands appeared to be the most important.

In answering their secondary objective, Laine, et al. [101] found no statistical difference between group trained classification accuracy and individual trained accuracy. However, relatively low classification accuracies were attained with approximately 66% classification accuracy observed for the three class workload problem (low, medium, and high) and approximately 83% accuracy for the two-class problem–not high versus high workload [101]. Gevins, et al. [61] had strikingly similar two-class results when they created a group classifier for each individual by training on all but the tested individual in their experiment described in Section 2.1.1. This group classifier resulted in a mean classification accuracy of 83%; a significant reduction from the 94% accuracy for individually trained models [61].

A final noteworthy outcome of Laine's study is that the researchers started with a three class model for workload (low, medium, and high), but switched to a two class model to achieve satisfactory classification accuracy due to systemic misclassification of medium and low workload. This shows that there was not enough specificity in the single-hidden-layer model to accurately discriminate between low and medium workload or that the difference was not present in the signal. The reduction from a multi-class environment to binary classification appears to be a trend throughout all of the workload classification literature. Classification methods with greater capacity should be explored as classes increase in number.

Popovic, et al. described PHYSIOPRINT, a psychophysiological workload clas-

sifier based upon 20-channel EEG, EOG, ECG, and EMG analysis combined with the theoretical construct of the U.S. Army's IMPRINT model [128]. Popovic, et al. correctly identified two significant shortcomings that need to be addressed in future work. The first is that no large-scale psychophysiological workload studies have been accomplished–a major shortcoming of past research efforts. Secondly, there has been no ability to tune a classifier to account for individualized traits, which has resulted in models that do not generalize well across subjects [128]. They set out to perform a larger-scale study using a model built on data collected from 22 individuals. Workload was broken down into each component specified by the IMPRINT model. Of interest, one of the metrics is mental workload. Popovic, et al. found this metric to be the most difficult to accurately classify in a cross-participant manner, achieving an accuracy of only 72.5%. EEG artifacts were removed and spectral features were generated for each 2 second window with 50% overlap. The overall design of their experiment was not set up such that high vs low cognitive workload settings were determined. Rather, classification among four different tasks was attempted: No activity, an alternative selection task, a recall and encoding task, and an arithmetic task. The model was trained using leave-one-out cross-validation based on participant. Findings from their study indicate that having greater scalp coverage by using more EEG channels, and incorporating alertness and fatigue measurements improves cross-participant workload classification accuracy [128]. Combinations of psychophysiological signals appeared to be complementary [128]. Furthermore, using short-term history of acquired signals also improved results [128]. This hints at a temporal interdependence between cognitive states which is oft neglected by standard analytical techniques.

Fazli, et al. used existing Brain Computer Interface (BCI) data from 45 individuals across 90 sessions to train an ensemble of classifiers to identify imagined right hand

versus left hand movement [56]. The goal was to create an ensemble which could handle cross-participant differences and classify on new subjects with no prior data from the subject. After training on this set, a separate hold-out set with 29 individuals and 53 sessions was used to assess model performance against various baselines. Because nonlinear classifiers did not result in significant performance improvements, LDA was selected as the learning method. Distributional shifts were implicitly assumed in the model anytime a new subject was being evaluated. A unique forgetting mechanism was used to combine the relevant classifiers from the ensemble. Fazli, et al. used L_1 regularized quadratic regression to select final weightings and reduce the number of classifiers in the ensemble to only the relevant ones [56]. The researchers used cross-validation to tune the model [56]. Using baseline subject-dependent training and evaluation methods yielded an error rate of 28.9%. Using an ensemble method with no subject-specific data resulted in a 30.1% error rate, only 1.2% worse than subject-specific methods. The ensemble performed better than the best baseline group model which resulted in an error rate of 36.3% [56]. These results indicate that using ensembles of classifiers can improve classification accuracy over traditional grouping methods and achieve comparable results to individually trained and tested classifiers.

To summarize up to this point, four distinct grouping methods have been used:

1. Grouping by average performance to set a group threshold.

2. First grouping all individuals but one into a single group and then trying to find features of the group that are salient and testing on the remaining individual.

3. Examining and using a linear combination of individualized features that may not be shared equally by all individuals as input to a group ANN.

4. Using a weighted ensemble of individually-trained classifiers.

Of these grouping methods, the third and fourth options tended to yield the best performance and were both forms of cross-participant ensembles. For each individual, only the important factors from each ensemble have a significant impact on the final classification while the others turn into small noise terms. The success of these approaches lends credibility to using an ensemble of classifiers to handle cross-participant differences in future work. In Chapter V ensembles of deep neural networks are evaluated for cross-participant classification using the MATB environment.

Next, deep learning theory, including data augmentation, and its application to a variety of EEG analysis domains is discussed.

2.3 Deep Learning

Several shared themes are prevalent across a wide spectrum of deep learning application areas which are applicable to EEG analysis. The first is the idea of creating architectures which are capable of learning feature representations that are better than hand-crafted features. Bengio, Courville, and Vincent enumerated a number of properties that are desirable in a learned representation. Several are important to EEG analysis and are foundational to understanding the architectural decisions described in subsequent chapters. Two of these important properties are: capturing multiple explanatory factors, and identifying factors that are shared across participants [18]. By creating a representation that can model the interaction of many factors in a representation, it may become possible to disentangle these factors in a distributed representation [18].

The second theme involves incorporating domain-specific knowledge into the architecture of the deep neural network in order to enforce constraints and impose biases on the generation of features which hold true for that domain. Deep learning uses a neural network's architecture to learn a layered representation of the raw data where

each successive layer is obtained by simple non-linear transformations which create more abstract and useful features by amplifying or suppressing aspects of the previous layer's representation [103]. This abstraction enforces invariance to local changes in input [18]. One example where this abstraction is built-into the network architecture explicitly is by using pooling [18]. Pooling layers in neural networks output summary statistics of nearby values within the network [17]. This enforces local translational invariance, a very important property for many domains including EEG analysis and image classification–it does not matter exactly where a feature is, but rather that the feature is present [17]. Specific ways to enforce desirable properties such as the ability to handle temporal dependencies, or designing-in invariance to physical processes such as translation or rotation are discussed in the following subsections.

Manifold learning and sparsity are also desirable properties. Manifold learning involves creating an architecture that can identify a much lower dimensional structure that encompasses the majority of the probability mass compared to the much larger higher dimensional sparse space [18]. Manifolds can be learned by autoencoders, Siamese-triplet networks (as discussed in Chapter VI), and other encoder-decoder structures. Sparsity goes hand-in-hand with manifold learning, although it is generally not associated with identifying a manifold as much as using some form of regularization to add a penalty to the loss function [18]. Typically, well-regularized models will generalize better, especially if there are a relatively small number of training examples, as is the case for EEG data. These concepts apply to EEG analysis because most of the electrode/frequency band combinations at any given time do not play a large factor in determining workload or are highly correlated [77]. L_1 regularization and dropout are two techniques that will be used to enforce a sparse representation and to mitigate co-adaptation of hidden units [122, 150]. Dropout prevents co-adaptation of the hidden units by temporarily removing a percentage of

randomly selected nodes, including their input and output, in a given layer during a training pass [80]. This forces hidden units to learn features without depending upon particular nodes to correct mistakes made during learning [80]. Conversely, L_1 regularization penalizes model complexity by adding a regularization term to the loss function which forces less important parameters to be exactly zero [17]. In Chapters III - VI, these concepts are combined with theory from psychology and neuroscience to build representations that incorporate domain-specific knowledge regarding the way humans process workload. Next, specific deep learning techniques will be introduced. To minimize overlap of material, it is suggested the reader refer to Section 4.2.3 for a discussion on RNNs. CNNs, DBNs, and SAEs are covered next.

2.3.1 Convolutional Neural Networks.

CNNs are useful for processing data that comes in the form of tensors which have dependencies along certain tensor dimensions. Such data is often 1-d (single channel time-series), 2-d (images or spectrograms), or 3-d (videos or sequences of 2-d images). Often additional dimensions of a tensor are needed to define the data structures required for input into a deep learning architecture or to handle multiple channels, but in the descriptions in this section, those added dimensions are ignored. Each convolutional layer is typically the composition of several stages: a convolution stage where the input is convolved with a set of kernels, a detector stage where the output of the convolutions are passed through a nonlinear activation function such as a Rectified Linear Unit (ReLU) to produce an activation map, and a pooling stage where statistical summaries of local data are used to downsample the activation map [17, 103]. Batch normalization is often used to enhance trainability (rate of convergence and accuracy) and to regularize the model [86]. Throughout this section, canonical image examples are often used to describe concepts because they make intuitive sense

and are simple to understand. The remainder of this section will assume the reader has at least a basic understanding of CNNs and will cover conceptual design guidelines and recent developments that have drastically improved state-of-the-art results using CNNs. For a detailed review of CNN basics, see [17].

LeCunn, et. al. describe the four concepts that engender the unique properties that CNNs possess: parameter sharing, local connections, pooling, and many-layered depth [103]. Parameter sharing leads to an efficient network parametrization and also makes CNNs work well in cases where translational symmetry is present [43]. In the context of a CNN, translational symmetry means that both the classes and the distribution of the data are invariant to location shifts: a dog is still a dog whether it is in the upper left part of an image or the center of an image [43]. The combination of local connectivity and parameter sharing give rise to equivariance to translation [17]. In image processing terms, this means that translating an image and then doing a forward pass through a layer is equivalent to performing a forward pass on the original image and then translating the resulting feature map [43]. Translational equivariance is a property which preserves translational symmetry as a network's depth increases which enables the creation of deep convolutional networks [43].

Two types of pooling will be important for EEG processing. The first type is pooling within a feature map. In this case, pooling imparts invariance to local translation. It makes the assumption that identifying when a particular feature is present is more important than knowledge about the exact location of that feature, and trades some location specificity in order to more robustly detect a feature [17]. This type of pooling will be useful in high density channel EEG processing because it will allow the learned representation to become invariant to small spatial or frequential translations that may differ slightly from one individual to another, or from small changes in placement of the electrodes from one session to another. The second type of pool-

ing that will be important to EEG processing is pooling across feature maps, also known as global pooling [17, 107]. This type of pooling can be especially useful when using what is termed in this manuscript as a global correlational layer–a layer which identifies correlations globally between channels and frequency bands as illustrated in Figure 1. Each convolutional kernel will be n-channels times f-frequencies in height and will have a width of 1 time step. This type of convolution will search for a unique pattern of correlation between different channel-frequency combinations that result in a particular type of workload. When a network is trained using examples from a variety of task environments and individuals, this should allow the network to identify specific distributed patterns of brain activity which are invariant across groups of individuals or across different tasks. The activation maps resulting from these workload-specific kernels will be pooled together using a global max-pooling function across all the kernels, thus determining the global workload response to the input stimuli. This type of feature could provide an indication of the dominant pattern of brain activity at a given time if a strictly convolutional network is used. This is because each separate fork will then output its max activation (through global pooling) for a given time step which could then be processed sequentially with an RNN or 1-d CNN to determine the experienced workload level for a temporal segment. An example of the global max pooling process is also illustrated in Figure 1.

The most significant breakthroughs using CNNs have been produced during image classification competitions starting with the development of AlexNet in 2012 and continuing with VGGNet, GoogLeNet, and ResNet. The evolution of ideas will be studied through the lens of each of these architectures and concepts that could be useful for EEG processing will be identified. Before delving into the details of each network, it is prudent to mention one topic that will not be discussed until a later section, but that was instrumental to each of these competition-winning CNN architectures: the

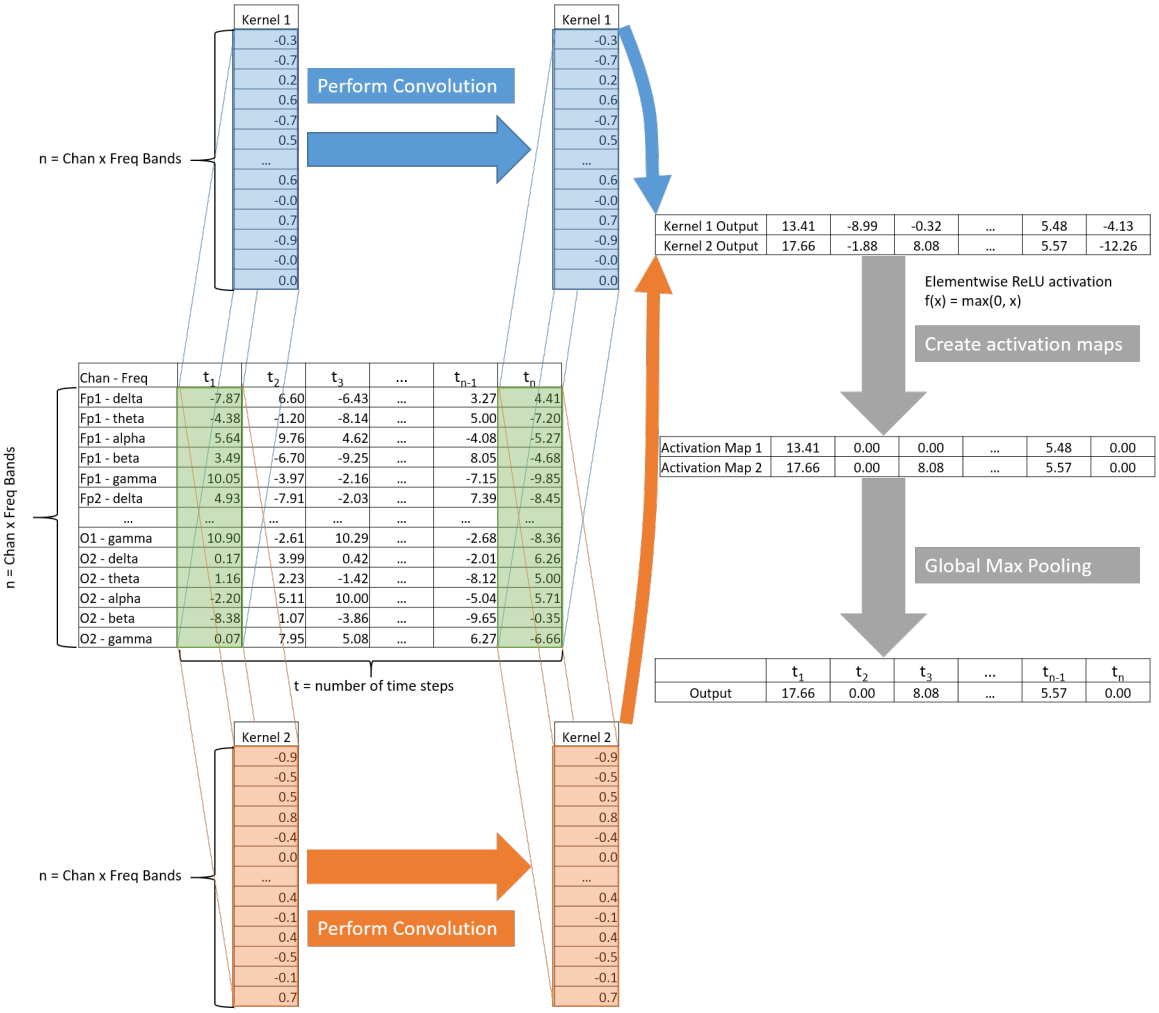


Figure 1. This illustrates the ideas of global correlational layers and global max-pooling. Each kernel is full length in a flattened channel-frequency dimension and of length 1 in the temporal dimension. Convolution is performed to learn kernels that correspond to channel-frequency patterns which correlate with a particular workload state producing a k kernels by t time steps matrix. Global max-pooling is then performed across each row of the activation map matrix for each time step resulting in the most activated kernel being identified for any given step.

prodigious use of physically-plausible data augmentation. Several image-specific data augmentation techniques including randomly selected fixed size cropping, random rescaling and cropping, horizontal flipping, randomly selected rotations, and random drawing of occlusions [117]. A thorough overview of data augmentation is provided in Section 2.4. The excellent results achieved in the image processing domain would not be possible without data augmentation as it helps reduce overfitting of very large networks [99].

These breakthrough networks will be discussed starting with the development of AlexNet in 2012. Krizhevsky's AlexNet architecture consisted of 8 layers: 5 convolutional followed by 3 fully-connected layers [99]. It achieved a top-5 test error rate of 15.3% and a top-1 error rate of 36.7% in the ILSVRC-2012 competition, representing a 42% reduction in error compared to the next nearest competitor [99]. The use of ReLUs, training on multiple Graphics Processing Units (GPUs), using a normalization scheme on the feature maps at each layer, and the use of overlapping pooling resulted in faster training, better classification accuracy, and better generalization [99]. The first two fully-connected layers used 50% dropout to act as a form of regularization and reduce overfitting [99]. AlexNet was the first large-scale GPU-trained CNN and ushered in the era of deep learning for image recognition.

Simonyan and Zisserman produced 16 and 19-layer versions of their deep architecture (VGGNet) for ImageNet Large-Scale Visual Recognition Challenge (ILSVRC)-2014. This architecture was significant because it was the first to feature small 3x3 filters in all convolutional layers with a stride of one [146]. Significant efficiencies were gained through the key realization that stacking a number of small-filter layers without pooling in between, enables an increased receptive field using a smaller number of overall parameters [146]. For example, if three 3x3 convolutional layers are stacked, they have a receptive field of 7x7, yet only use 27 parameters rather than 49

parameters; resulting in a commensurate reduction in computation [146, 157]. This type of architecture has the added benefit of allowing for more non-linear transforms to occur than a single larger layer, adding to the network's ability to build higher-order abstractions [146, 157]. The takeaway for EEG processing is that the use of stacked, small filters reduces computation and could help establish localized spatial regions or frequency representations of brain activity that are grouped together in a computationally-efficient manner.

Two other architectures which significantly improved the state-of-the-art are the different variants of Szegedy's GoogLeNet architecture with *inception* sub-networks and the development of deep residual networks (ResNets) [157, 156, 74, 75]. These are discussed at length in Section 5.1.2 and their understanding will not be required for the intervening sections as other EEG researchers have not yet leveraged these techniques. Now a brief transition to self-supervised and unsupervised deep learning techniques which have been used in EEG research is discussed for completeness.

2.3.2 Deep Belief Networks and Autoencoders.

Historically, both DBNs and SAEs were used as methods to perform unsupervised pretraining of feedforward neural networks followed by supervised fine-tuning [139]. This use case is still often found in EEG analysis literature and so these methods are briefly described in this section. Autoencoders are simply neural networks which are trained to reproduce the input as output after passing through a compressed representation layer which can learn the structure of a low-dimensional manifold embedded in a higher-dimensional space [17]. There are two parts to an autoencoder: an encoder section which takes the original input and produces an intermediate representation that captures some desired aspect of the input features; and a decoder section which takes the intermediate representation and learns to transform it back into the input

[17]. Some method is required to force the intermediate layer to deviate from the identity function such that a useful representation can be learned [17]. Two methods to do this are to use either an undercomplete hidden layer (reduced dimensionality), or a complete/overcomplete hidden layer with some form of regularization to enforce a sparse representation [17]. Aside from sparse autoencoders, denoising autoencoders are also frequently used in signal processing. These are trained by corrupting the input signal with noise and then trying to reproduce the uncorrupted original signal to learn a representation capable of denoising a signal [17]. Dimensionality reduction as a form of nonlinear Principle Component Analysis (PCA) is another common application for autoencoders. Deep autoencoders generally result in higher compression ratios than shallow ones due to the more complex representations possible [17]. These deep autoencoders are often trained in a greedy layer-wise fashion by training each new layer as a shallow autoencoder with the output of the previous layer used as the input for the next layer [17, 19].

DBNs are deep probabilistic models composed of stacked layers of Restricted Boltzmann Machines (RBMs) trained in a greedy layer-wise fashion [17]. They were some of the first deep models to be successfully trained, but are now rarely used compared to other methods [17]. Each individual RBM is a parameterized energy-based model consisting of a single densely-connected layer where only connections between input units and output units are allowed [17]. The desired objective is to learn the latent variables by maximizing an approximation to the log-likelihood of the probability of the energy function via contrastive divergence and block Gibbs sampling. This relies on the "restricted" aspect of the RBM since each hidden unit is conditionally independent from all other hidden units, given the visible units, and all the visible units are conditionally independent of one another, given the hidden units [17]. Once trained, the DBN weights historically have been used to initialize the weight

matrices of an Multilayer Perceptron (MLP) followed by supervised fine-tuning [17]. These methods have largely been supplanted by the aforementioned improvements in initialization techniques.

2.4 Data Augmentation

Data augmentation is a general term used to describe any process which expands a dataset by synthesis of new observations based on domain specific knowledge. Physically-plausible data augmentation has proven to be of utmost importance to the computer vision community as it significantly improves deep neural network performance. In fact, the performance of a learned representation generally scales with the size and quality of the training dataset [145]. Historically reductions in error in excess of 50% for a given model have resulted from using data augmentation in deep learning models [117]. These reductions in error can primarily be attributed to two primary factors: the ability to use a much larger (deeper and wider) network to analyze the data without overfitting, and the increased density of data points near decision boundaries which leads to better generalization [35, 99].

Relatively little work has been done to apply data augmentation to EEG analysis and most of it consists of Gaussian noise injection. Bashivan attempted to use Gaussian noise injection to spatially-projected, by-channel, Power Spectral Density (PSD) values during a working memory task which resulted in a slight increase in error rate [14]. Li also performed a form of data augmentation by injecting 5dB white Gaussian noise into the signal of each trial and creating 250 realizations of each of 40 trials per person (total of 10,000 trials per individual) [106]. The authors mention that this makes the model more robust to inherent noise, and that the increased number of trials available improves training and generalization [106]. However, no comparison to a baseline model trained without data augmentation is made, nor are other levels

of Gaussian noise evaluated; so the significance of the effect is not clear. Conversely, Casson evaluated the effect of adding various levels of white Gaussian noise to the testing data to determine if this would lead to lower generalization error. No statistically significant improvement resulted from adding the noise to the testing data [32]. Kalunga, et al. took an approach similar to Synthetic Minority Over-sampling Technique (SMOTE) and interpolated points on a manifold in Reimannian space to populate underrepresented classes in an unbalanced BCI application [93]. Minor, yet statistically-significant improvements were realized using this augmentation approach coupled with a 2-layer MLP classifier. This represents a prototypical interpolation technique and can be applied to nearly any dataset with varying effectiveness. This is discussed here due to the use of Siamese-triplet networks in Chapter VI. Siamese-triplet networks can be considered to use a form of data augmentation because they greatly expand the available unique training examples for a given network in a manner conceptually similar to SMOTE. Discussion of these networks are deferred to Chapter VI.

Aside from the relatively unsuccessful EEG-specific augmentation techniques, one method from time-series literature is used with all models trained in Chapters III - VI: window slicing. Cui, et al. [46] describe window slicing as a time-series data augmentation technique because it is a way to increase the number of training samples for a dataset. This could be considered the time-series equivalent of the extensively used random cropping method from image processing, except that all possible overlapping slices of a particular temporal length are used. Now that the fundamentals of deep learning have been discussed, examples of existing EEG analysis conducted using deep neural network models are examined.

2.5 Deep Learning Models for EEG Analysis

The majority of research analyzing EEG signals using deep learning techniques has not involved estimation of operator workload. Rather it has focused on other medical and BCI applications such as detection of epileptic seizures, sensorimotor imagery classification, fatigue level identification, and affective state characterization. Despite no direct connection to operator workload estimation, much can be learned by examining the preprocessing pipelines and architectural decisions these researchers made since similar techniques will likely apply across different EEG analyses. Mirroring the deep learning section, relevant EEG analytical results are surveyed beginning with RNN models, followed by CNN, DBN, and SAE models. Those models that are combinations of multiple categories are discussed at the first opportunity.

2.5.1 Recurrent Neural Networks.

The use of recurrent neural networks to analyze EEG data for various medical and BCI applications saw some early activity approximately a decade ago. This was followed by a gap in usage until recent successes in other fields, due to advances in deep learning, spurned renewed interest in RNN EEG analysis. Even given the great improvements in fields like natural language processing and automatic speech recognition, relatively few researchers have applied deeply recurrent neural network techniques to classify EEG data, and even fewer have applied it to examine operator workload.

Bashivan, et al. trained a deep recurrent-convolutional neural network accounting for both temporal and spatial dependencies in the network [14]. They began by performing a Fast Fourier Transform (FFT) on each time-series signal from each electrode and estimating the power in the theta, alpha, and beta frequency bands over 3.5 second working memory experiment trials with four classes of task difficulty. Then,

they created a time-series of images by performing a 2-d Azimuthal Equidistant Projection (AEP) for power in each of the three different bands. This preserved distances from each electrode to the center point of the EEG cap, thus accounting for some spatial dependencies. These images were fed into a series of convolutional and max pooling layers. Their best performing model fed the output from the convolutional portion of the architecture into a 1-d convolutional layer, as well as a LSTM layer, and merged the output from both of these into a fully connected layer. This was then connected to a softmax layer which provided the final classification result. This architecture used 1.62 million parameters and was able to reduce the classification error from 15.34% to 8.89%; an impressive 42% reduction compared to a baseline radial basis function SVM [14].

Like Bashivan, Li et al. implemented a deep convolutional-recurrent structure and used it to perform within-subject valence and arousal emotion classification for each of 32 participants based on 40 one-minute long music video trials from the DEAP dataset [106, 98]. Preprocessing consisted of creating scalograms using continuous wavelet transformations and selecting lower and upper cutoff frequencies (scales) based on the energy-to-Shannon entropy ratio. This ratio identifies frequency ranges where the spectral energy is high and Shannon entropy is low [106]. Each scalogram was then divided into 1 second windows and the energy in those windows was summed to form vectors for each channel at a given time step [106]. These were turned into frames for each channel with one dimension being frequency and the other being time. Li's architecture incorporated two convolutional and max pooling layers prior to flattening the data. The flattened data was input to a many-to-many LSTM with each output connected to a softmax classification layer. The values of each softmax activation were then averaged together and the class with the highest average probability determined the predicted emotion class [106]. There are several significant differences in structure

compared to Bashivan's implementation which are noteworthy. Rather than using a 2-d projection and a 2-d spatial convolutional filter, the first convolutional layer had filters of size $channels \times 1$ and was able to determine correlations between different channels at a specific frequency [106]. A 2×1 average pooling layer with stride 2 was then used to combine adjacent frequencies. The second convolutional layer then found correlations between the channel correlation feature maps (correlations of previously correlated channels) by using 16 1×1 filters. A final 2×1 average pooling layer with stride 2 was again used to combine adjacent frequency bands [106]. The output of these convolutional layers were passed into a LSTM which then accounted for temporal dependencies in the higher-level feature representation [106]. This model resulted in classification accuracies of 72% and 74% for valence and arousal respectively. This was a 10-15 percentage point improvement compared to SVM and Random Forest (RF) baseline implementations, and was in-line with highly specialized hand-crafted feature representations created by other researchers [106, 36, 98]. While Li, et al. created an excellent model by using the strengths of each structure in their deep architecture the overall design did not account for cross-frequency coupling from different EEG channels. This shortcoming is addressed by work in Chapter V.

Thodoroff, et al. used a slightly modified version of Bashivan's recurrent-convolutional architecture to achieve state-of-the-art results in epileptic seizure classification [164]. The modification to Bashivan's deep architecture included the use of a bi-directional LSTM following the 2-d convolutional layer and prior to the fully-connected layer [164]. A bi-directional layer was added here because neurologists typically use both past and future information to make a diagnostic decision on whether an EEG segment contained epileptic activity [164]. The initial convolutional and pooling layers enforce spatial invariance which is important for seizure classification since a seizure

can occur in any localized region of the brain, or globally [164]. This spatial invariance is also necessary to perform cross-participant diagnosis since seizure manifestations vary significantly in location, duration, and shape across individuals [164]. For patient-specific detection, the recurrent-convolutional model had an equivalent sensitivity and false positive rate as results reported by Shoeb who used a SVM with hand-tailored features [143, 164]. Furtheromre, it displayed a robustness to EEG channel dropout not present in other methods which could enable less EEG nodes to be used in future diagnoses saving both time and money [164]. Cross-patient performance outperformed the only commercially-available automated seizure detection system with a reduction in error for sensitivity of 54.5% and a 52.9% reduction in false positive rate [164]. Thodoroff's work stands as an excellent example of how to incorporate domain-specific knowledge into a deep learning solution.

In other work, Binz, et al. used RNNs to classify imagined sensorimotor imagery data for a BCI workshop competition [22, 23]. They applied a derivative of the LSTM unit called Dynamic Cortex Memory (DCM) to artifact-corrected EEG power measures. A single hidden layer was used with eight DCM units followed by a softmax output layer. While their results did not achieve state-of-the art accuracy, several advantages were present in their solution. Their network was able to provide real-time results and their solution did not need to specify a time window since the network learned appropriate temporally-dependent sequences [22].

Guler, et al. were among the earliest users of RNNs for analysis of EEG data in their application to epilepsy diagnosis [69]. An Elman RNN was used which connected a context layer to store the output of each hidden neuron for one time step. The connections to the context units all had a weighting of one, while the connections back to the hidden units were learnable, effectively allowing the network to incorporate temporal dynamics with explicit memory of one time step [52]. Guler used four

statistical properties (mean of absolute values, max, average power, and standard deviation) of Lyapunov exponent distributions as feature vector inputs to a single layer Elman RNN with 15 recurrent neurons. Training was performed using a Levenberg-Marquardt optimizer [69]. Lyapunov exponents are used to characterize the stability of chaotic systems, or systems which are sensitive to initial conditions, by examining the divergence rate of several time series with similar initial conditions [2]. Cross-participant models were created using data obtained from other individuals to train models for a given individual [69]. Three equally distributed classes of EEG data were used for training and testing from healthy individuals and individuals with a history of seizures: Healthy, seizure-free epileptogenic, and epileptogenic seizure segments [69]. Results of the Elman network significantly outperformed a feedforward MLP with overall classification accuracy of 96.8% compared to 91.3%, in this three-class problem [69]. An important takeaway from this study is that sometimes a smaller subset of features (distributional properties), as opposed to all elements of the distribution, may result in better generalization during testing. Furthermore, the results clearly show an improved accuracy when taking temporal dynamics into account.

Ubeyli analyzed an expanded version of the same dataset as Guler which included two more classes [166]. Also similar to Guler, Ubeyli used an Elman RNN with a single hidden layer, but increased the number of recurrent nodes to 25 [166]. Ubeyli also used a different preprocessing methodology which utilized three eigenvector methods to extract PSDs for each data segment. Then, the min; max; mean; and standard deviation of the power levels for each segment were computed for each of the three eigenvector methods and used as 12-element feature vectors [166]. The RNN performed quite well and achieved a classification accuracy of 98.15% which compared favorably to the 92.9% obtained from a single-layer MLP with 30 hidden nodes [166]. These results also indicate that a PSD preprocessing technique may yield slight im-

provements over Lyapunov exponent preprocessing techniques.

The work of Davidson, et al. represents the earliest use of LSTM neural networks to analyze EEG data for a BCI application [48]. They created a system to identify lapses in attention based on EEG spectral data [48]. Of the 15 participants who performed a tracking task while EEG and video were recorded, 8 exhibited lapses at least once and were therefore used in this study [48]. The LSTM network was composed of six LSTM blocks each containing three memory cells which were fed feature vectors of 14 values every second [48]. It is not clear if the blocks were arranged in parallel or serially. Cross-participant, leave-one-out cross-validated training was performed for each of the eight participants [48]. Results indicated significantly worse performance for the LSTM lapse detector compared to a single-layer MLP trained on video data [48]. However, many advances in deep learning have occurred since their study (2005) which have significantly improved training of deep neural networks to include initialization strategies and abundant computation via GPU training that significantly improves results. Additionally, the small size of their feature vector and dataset, as well as the limited capacity of their LSTM implementation likely induced a significantly negative effect on model performance.

Ruffini, et al. trained an Echo State Network (ESN) to determine the expected progression of neurodegenerative diseases based on EEG signals gathered at the beginning of a multi-year study that followed both at risk and control group patients [133]. An ESN is a RNN with recurrent weights set to a particular value (typically tuned as a hyperparameter to mitigate the vanishing/exploding gradient problem) and which only has trainable output weights [17, 133]. Ruffini used PSD features based on 1 and 4 second windowing and found that the 1 second windows increased classification accuracy by approximately 10% compared to the 4 second windowing procedure [133]. Results of the ESN correctly classified 85% of the disease-progression/no progression

cases which was equivalent to the 85% accuracy results attained using a previous SVM implementation [133]. Based on these results, more robust alternatives to ESNs, such as LSTMs should be considered for EEG signal analysis.

Mazumder, et al. showed that a recurrent neural network outperformed SVM, LDA, and K-Nearest Neighbors (KNN) classifiers for a cognitive state classification task using autoregressive EEG features [118]. Nine participants performed three trials where they were shown three 40-second video clips that were somehow related and then were given 30 seconds to consider the clips and identify the common link during each trial [118]. The study classified whether the participant was watching the video or performing the cognitive task of finding the link between the clips [118].

An and Cho compared performance of five deep RNNs employing different recurrent structures for classifying EEG time-domain data for a set of grasp-and-lift tasks [7]. The five recurrent structures were an LSTM, a standard Gated Recurrent Unit (GRU), and three variants on the GRU explored by Jozefowicz, et al [7, 92]. Twelve participants performed a total of 328 grasp and lift trials, each of which was subdivided into six different phases sequenced in the following way: No motion, start hand movement reaching for the object, grasp object and lift to particular height, hold object still at particular height, retract and replace object, release grip on object and return hand to original position [7]. Trials were sliced into 0.512 second frames each containing 256 observations with an unspecified amount of overlap between frames [7]. An and Cho's intent was to allow the network to learn relevant features from raw time-domain data rather than hand-engineering the initial feature vectors [7]. Results were generally quite good with each recurrent structure performing approximately as well as the others with maximum accuracies between 86.5% and 88.8% for the six class problem [7]. Using 50% dropout between the second recurrent layer and a fully-connected dense layer notably improved accuracy by 4%. Their results suggest

using deeply recurrent networks with input vectors consisting of fine-grained temporal data may provide an acceptable alternative to using hand-engineered feature vectors. However, due to the repetitive sequential nature of their data and the fact that they reported temporally smoothed classification predictions, some caution is warranted.

Liu, et al. showed that a Recurrent Self-Evolving Fuzzy Neural Network (RSEFNN) slightly outperformed a battery of other neural network techniques in predicting a normalized drowsiness metric during a driving fatigue study [112]. A high-fidelity vehicle simulator was used to study EEG patterns of drowsiness in 20 participants by sensing occipital lobe activity prior to and during lane perturbation events while performing a simulated highway driving task [112]. Response time to the lane perturbation events were used as a measure of alertness–when participants remained alert, their response time to correct the lane perturbation event would be reduced compared to a drowsy state. Occipital lobe EEG features from the five second period prior to the onset of the perturbation event were used as input for supervised regression with normalized reaction time (0 to 1) as the target value. The benefit of using a RSEFNN is that its architecture allows for structural learning (addition or deletion of fuzzy groups during training) as well as traditional parameter learning. This allows the RSEFNN to adjust capacity during training based on threshold hyperparameters [112]. RSEFNN results were compared to the following regressors: SVM, Self-Organizing Neural Fuzzy Inference Network (SONFIN), Fuzzy Wavelet Neural Network (FWNN), TSK-Type Recurrent Fuzzy Network (TRFN), and a Recurrent Wavelet-Based Elman Neural Network (RWENN) [112]. Each participant had an ensemble of 10 models of each type trained using all other participants' data. The results from the 10 homogeneous models were averaged to give a mean normalized reaction time prediction for each perturbation event and an associated Root Mean Squared Error (RMSE) was calculated. The two recurrent models, the RWENN and the RSEFNN outperformed all

other models. The RSEFNN had the lowest RMSE and resulted in a 23.2% reduction in error compared to the SVM and a 9.4% reduction compared to the best performing feedforward neural network (FWNN) [112]. While the reported results indicated slightly better performance for the recurrent methods, a rigorous statistical treatment in their study could have confirmed this. Despite the lack of statistical results, this study highlights the importance of taking temporal correlations into account in model selection.

In a follow-up study, Liu et al. evaluated the performance of a recurrent fuzzy network, RSEFNNs, as well as feedforward neural networks and SVMs before and after Independent Component Analysis (ICA) preprocessing to determine if recurrent networks have an "adaptive noise cancellation" effect and perform equally well in both cases [113]. Their results indicated that ICA has a beneficial effect for all regression techniques evaluated, and that the RSEFNN without ICA performed approximately as well as the best non-recurrent network with ICA applied [113]. Since ICA is not appropriate for real-time BCI use, due to the need for offline processing, using a recurrent model could yield equivalent accuracy gains in such cases [113].

Soleymani, et al. found that a RNN using a two-layer LSTMs architecture containing 178 hidden nodes outperformed multi-linear regression, SVMs, and conditional random fields in classification of participant valence (positivity or negativity) while watching short emotional videos and recording EEG and facial expressions [148]. PSD in theta, alpha, beta, and gamma bands as well as the difference in PSD between symmetrical electrode pairs from the left and right hemispheres were used as EEG-extracted input features [148]. Another 271 features were derived from distances between facial features [148]. Soleymani, et al. examined the assumption that facial muscular artifacts would contaminate the EEG signals and found the highest correlations with valence in the beta and gamma frequency bands in frontal, parietal and

occipital lobes; thus concluding that a combination of muscular artifacts and brain signals were likely present and significant [148]. A final significant result was that models trained separately on EEG features and facial features and with results fused together performed better than models trained with all features concatenated into a single feature vector [148].

2.5.2 Convolutional Neural Networks.

While CNNs have been in use in the computer vision community for decades, only recently have they been applied to analysis of EEG data. A notable exception was the work of Piotr Mirowski, a graduate student of Yann LeCun, whose CNN achieved near human-level performance in detecting epilepsy circa 2008 [120]. Due to the proliferation of high-quality deep neural network frameworks over the past several years, the vast majority of EEG research employing deep neural networks has occurred since 2014. With CNNs, details of the network architecture are extremely important because they impart strong biases on the network and have implications regarding what types of features can/will be learned. Beginning with Mirowski's work, the impact of each researchers' architectural decisions on CNN performance across a variety of EEG domains is explained in detail and conclusions are drawn regarding how to apply CNNs for operator workload estimation.

Mirowski, et al. used a variety of feature generation techniques combined with a modified version of LeNet-5 to achieve excellent specificity and false positive rate for the Friedburg epilepsy seizure prediction dataset with 21 patients [120, 104]. The most unique aspect of their implementation is their feature generation technique. For all N-choose-2 pairs of electrodes, several features were calculated over five second windows and then turned into 1 or 5 minute sequences (12 or 60 frames each) with the following being the most influential: maximal cross-correlation with delays of +/-0.5 seconds

on raw EEG signals, bivariate non-linear interdependence, and statistical properties of wavelet-derived phase information averaged within each clinical frequency band [120]. Due to the large number of features generated, L_1 regularization was used for training all models [120]. This is important to reduce the number of irrelevant features and enable better generalization when there is a large feature space; it is especially important when the feature space is larger than the number of observations [122]. All models were trained on only the patient's data due to significant differences in cross-participant presentation [120]. Three convolutional layers were used with temporal max pooling layers in between that pooled 10 seconds worth of data in each case. The first and last convolutional layers implemented convolutions across temporal periods, but were shared across channel pairs while the second convolutional layer used a matrix similar to Tabar's implementation (discussed shortly) to allow for temporal convolutions that were specific to the channel pairs [120, 158]. Overall, the CNN models significantly outperformed SVM and logistic regression models and resulted in a near-zero false-alarm rate and 100% sensitivity [120]. These results indicate that utilizing enumerated combinatorial bivariate features by electrode and frequency band coupled with a sparsity enforcing deep framework can yield excellent results.

Yang et al. used augmented common spatial pattern features as input to a five-layer CNN (2 convolutional layers interleaved with 2 max-pooling layers follwed by one fully connected output layer) to evaluate different methods for selection of pair-wise frequency band combinations [180]. This was used as input to the CNN for a motor imagery data classification task involving imagined right-hand, left-hand, foot, and tongue movements [180]. Retaining all pairs performed the best and no sub-sampling scheme was recommended. This result is similar to Mirowski's and indicates that when computationally feasible, retaining all features with an appropriately regularized deep convolutional network often leads to better results than using a down-selection

criteria prior to input into the network.

Tang, et al. used a five-layer CNN to classify a motor imagery task involving imagining left hand or right hand movements [161]. Two individuals each performed 460 randomly selected motor imagery visualizations (230 left and 230 right) while 28 EEG electrodes placed over the motor cortex recorded signals. PSD between 8 and 30 Hz was calculated and the ratio of Event-Related Desynchronization (ERD) to Event-Related Synchronization (ERS) was calculated for each electrode in each 50 ms window [161]. This yielded a 28x60 matrix (electrode or spatial dimension by temporal dimension) which functioned as an input into the CNN. After the input layer, the next two layers performed 1-dimensional convolutions in the spatial dimension (8 spatial filters of length 28) followed by the temporal dimension (5 temporal filters of length 60). An important point is that while convolution was performed in two dimensions, they were done so independently and sequentially so as to not blend the temporal and spatial dimensions [161]. These convolutional layers were followed by a fully-connected 100-hidden node layer and a softmax output layer [161]. A different choice of activation function may have provided better results since *tanh* and sigmoid functions were chosen for the convolution activation and the fully-connected layer respectively. These functions have a tendency to saturate in deep architectures compared to ReLUs [64, 99]. Overall, this architecture attained an average classification accuracy of 86.4%, significantly outperforming three variants of SVMs trained with a variety of features [161]. A final important finding in this work was that the active frequency bands for each individual were different (12-16 Hz vs 18-22Hz) and had to be identified and selected during preprocessing. This highlights the importance of developing an architecture that is at least partially invariant to shifts in frequency by performing convolution and max-pooling along the frequency dimension–a finding in automatic speech recognition literature [3]. A more complex version of this concept

is used in models developed in Chapter V.

Tabar and Halici constructed a deep neural network architecture consisting of a 1-d convolutional layer, a max-pooling layer, a 6-layer SAE, and a softmax classification layer to classify motor imagery data corresponding to imagined left or right hand movements [158]. Preprocessing consisted of creating a matrix of PSDs from 6-13 Hz and 17-30 Hz for the C3, Cz, and C4 channels with 0.256 seconds per window with a step size of .056 seconds. This resulted in 32 time steps for each 2 second motor imagery trial. Frequency-space was interpolated to 31 frequencies equally distributed between the aforementioned lower and upper bands for each electrode. Data from the three electrodes were then concatenated vertically in frequency space resulting in an input matrix that had 32 time steps horizontally and 93 frequency/electrode combinations vertically. A total of 30, 2-dimensional kernels with height of 93 (same vertical dimensions as the input) and width 3 were used in the convolution filter bank to perform 1-d convolutions across the time axis. This ensured no mixing of frequency or electrode information and allowed the network to respond differently to different frequencies and electrode locations [158]. Using a max-pooling width of 10 imparted an invariance to local temporal translations of approximately a half second which was noted to provide much better results than smaller time periods [158]. Futhermore, max-pooling resulted in far better performance than average-pooling since high values correspond to activations of neurons while average values fail to highlight this discriminating factor [158]. Unsupervised pretraining of the SAE was followed by supervised fine-tuning. The combination of the CNN and SAE into the final architecture resulted in the best network performance and outperformed either component individually as well as a competition-winning SVM implementation [158]. In the context of motor-cortex workload, this study can inform how to handle signals emanating from the motor cortex. The first takeaway is that the left and right

hemispheres should not be combined because signaling is specific to each side. Within a given side, temporal max-pooling may be useful to identify high-activity times more broadly and a variety of max-pooling time-lengths should be explored to determine an optimal value. Furthermore, comparing symmetrical differences in activity between the hemispheres within the same temporal neighborhood may identify time segments with increased or decreased motor signaling which may be correlated with physical workload.

Walker used CNNs to perform motor imagery classification of imagined left or right hand movement in a BCI experiment [168]. Walker recognized that anatomically correct spatial distances between electrodes should be used in a model that performs convolution in the spatial dimension [168]. Because of this, a spherical projection of the electrodes was created and electrodes were grouped using the K-disjoint nearest neighbors algorithm based on minimum arc-length between electrodes [168]. The size of the groups was set to $K = 3$ based on the thought that nearby electrodes should be related to each other and because only 31 electrodes were available in the dataset [168]. However, data was arranged in matrix form for input into the CNN with the closest electrodes arranged in adjacent rows and time as the horizontal dimension. While this arrangement takes into account some notion of closeness, it fails to adequately capture either the geometrical or anatomical structure which likely led to their mediocre classification results. A better approach from a geometric perspective was provided by Bashivan and described in section 2.5.1 [14].

Lawhern, et al. developed a CNN architecture with a small number of parameters that was able to generalize well across multiple EEG BCI analysis domains including visual stimulation of P300 Event-Related Potentials (ERPs), neural oscillations associated with movement-related cortical potentials, and sensorimotor rhythms evoked by real or imagined movements [102]. The architecture consisted of four convolu-

tional layers and a softmax classification layer [102]. Each convolutional layer used Exponential Linear Units (ELUs) rather than ReLUs since the mean activation value is closer to 0 with ELUs. This resulted in reductions in training time and better generalization performance [102, 40]. The first layer used 16 1-d kernels (each the same length as the number of electrode channels), which were convolved without zero-padding with each of the input tensors. This produced 16 outputs of 1 x 128 timesteps and resulted in an output tensor of shape (16, 1, 128) [102]. The second and third layers each employed 2-d convolutional filters, and 2-d max-pooling was performed as a dimensional-reduction technique [102]. Following each convolution operation, batch normalization, and a 25% dropout were applied prior to input to the next layer [102]. Heavy regularization by using a combination of dropout, batch normalization, and L_1 and L_2 regularization in each layer resulted in improved robustness due to avoiding training to the noise on relatively small EEG datasets [102]. The most interesting aspect of Lawhern's model is that each of the kernels in the first layer had the ability to learn useful channel interactions and had the effect of abstracting away the need to explicitly model locational dependencies inherent in an EEG system. Their model was trained in a cross-validated, cross-participant manner so that it was user-agnostic [102]. Statistically significant improvements in Area Under Curve (AUC) ($p < .05$) were achieved over state-of-the-art reference algorithms in three of the four domains with no statistical difference noted in the fourth domain (movement-related cortical potential dataset) [102]. Their excellent cross-domain results indicate that a derivative of this architecture is worth exploring, and architectures in Chapter V borrow some concepts from Lawhern's work.

Stober, et al. trained a wide variety of convolutional neural networks using a Bayesian hyperparameter search technique to optimize input features and network structure for a musical rhythm identification task based on EEG signals [153]. A mean

classification accuracy of 24.4% was achieved in the 24-class (1 class per rhythm) classification problem for 13 participants with individually tuned CNNs. A comparison of train and test performance between CNNs trained with frequency-domain features versus raw EEG waveforms resulted in an order of magnitude reduction in training time and slightly improved test classification accuracy using frequency-domain features [153]. A unique characteristic of their CNN was the use of a Deep Learning Using Linear Support Vector Machines (DLSVM) output layer with a hinge-loss cost function rather than the traditional softmax output layer with a categorical cross-entropy loss function [153]. In other work, the DLSVM output layer used a linear SVM in the output layer coupled with a margin-maximizing loss function and improved state-of-the-art results on a variety of deep learning datasets including MNIST and CIFAR-10 [160]. Stober also performed an analysis regarding what cutoff frequency to use for frequency-domain input which showed that all frequency content should be retained [153]. A final comparison to polynomial SVMs was done showing approximately a 10% improvement in classification performance when using the CNNs [153]. The most important result of this work was that the comparison between raw features and frequency-domain features clearly illustrated the performance gains associated with using frequency-domain input features.

Hajinoroozi, et al. constructed two unique CNNs which were designed to perform convolutions across 1-second temporal periods of raw EEG data from each channel resulting in improved cross-participant and within-subject classification performance for a driving simulator task compared to a large array of baseline algorithms [71]. A total of 37 participants' EEG data from three studies examining driver fatigue in a high-fidelity simulator, similar to that described in Liu's work [112] described in Section 2.5.1, were used to predict driver performance during a lane perturbation event as either poor or good [71]. Both CNNs used a kernel size of 1×250 to convolve

across the time domain, effectively searching for ERPs present in each individual channel. The first CNN used 10 kernels while the second used only 1 kernel which was pre-trained as an RBM followed by fine-tuning. Each CNN was followed by a max-pooling layer, two fully-connected layers, and an output layer [71]. The CNN with 10 filters significantly outperformed all other models with an AUC of .8608, while the RBM CNN performed far better than any other model in the cross-participant classification environment, achieving an AUC of .7672 [71]. These results suggest that either the reduced model capacity of the RBM CNN led to better cross-participant generalization, or that the process of performing unsupervised pre-training helped learn shared features across individuals. Overall, the use of an architecture which uses raw EEG to find per-channel ERP signatures was novel and warrants further investigation as a merged component in a large deep neural architecture which also incorporates time-frequency domain features.

2.5.3 Deep Belief Networks and Stacked Autoencoders.

DBNs and SAEs have been used by several research groups to perform EEG-based classification for various tasks. Generally, the results of these attempts have been less successful than those which used RNNs or CNNs. However, one research group achieved excellent results using autoencoders to perform early-fusion in a multi-modal emotion recognition setting. This implies that early-fusion using autoencoders is a technique worth exploring further. Additionally, Siamese-triplet networks, which are introduced and used in Chapter VI are conceptually similar to SAEs.

An, et al. used DBNs to classify sensorimotor imagery data [8]. For each of four participants, 30 left-hand and 30 right-hand imagined motor imagery trials were conducted. electrooculograph (EOG) was captured and used to correct artifacts and an 8-30 Hz band pass filter was applied to the data. A Fast Fourier Transform was

then used to convert each 4 second trial into the frequency domain. It was not clear how many, nor which channels were used as input into the classifiers. Greedy layer-wise unsupervised training was performed using Guassian RBMs. Deep belief networks of 4-16 layers were trained and it was determined that a depth of 8 was optimal resulting in an average binary classification accuracy (left vs. right) of 81% [8]. This represented an average improvement of approximately 4% over a baseline SVM method [8].

Jia, et al. recognized low Signal to Noise Ratio (SNR), and non-stationarity as significant challenges which markedly impair traditional classifier performance during classification of affective state [90]. To address these challenges, they realized that several areas of the brain did not yield salient information and developed an EEG channel selection procedure based on a two-stage RBM ranking strategy. They then trained a deep belief network to classify whether each of 32 participants liked or disliked each of 40 one-minute long music videos from the DEAP dataset [90, 98]. Due to the costly nature of labeling human subject experiments involving EEG data, a semi-supervised, active-learning method was employed to reduce variance in their overall model by leveraging unlabeled data. Their semi-supervised training consisted of two aspects. The first was an unsupervised component in the loss function which served to regularize the training. They also created a classifier based on the already labeled data, and then using it to determine what unlabeled data the classifier was most uncertain about by examining the posterior probabilities of each class for a given observation. Those observations the classifier was most uncertain about were then labeled and the model was retrained based on the new dataset. This procedure comprised the second aspect of their semi-supervised training scheme and was repeated until they ran out of budget for labeling data [90]. Jia, et al. used a SVM, as well as a SVM coupled with PCA and Fisher Criterion for feature selection as baseline

models. They found that PCA and Fisher Criterion did not meaningfully improve results compared to using a SVM with no feature selection procedure [90]. Furthermore, their semi-supervised deep belief network significantly outperformed all other models and resulted in an average AUC of 0.808.

Jirayucharoensak, et al. used a three layer SAE with a softmax output layer to classify valence and arousal states in a leave-one-out cross-participant analysis using the DEAP dataset [91, 98]. They also show that preprocessing PSD features using PCA and applying covariate shift adaptation both improve performance of their network by approximately 5-6% and resulted in the best accuracy for valence classification (3 classes) of 53.42% and the best arousal classification performance (3 classes) yielding 52.03% accuracy [91]. These compared favorably to results obtained using a Radial Basis Function (RBF) SVM. However, it should be noted that the SVM's hyperparameters were not tuned which may account for the difference in classification accuracy. The most interesting idea presented in this paper was that using covariate shift adaptation could help mitigate signal non-stationarity. Covariate shift adaptation normalizes data within a small sliding window, in this case over a 10 second window, and results in a strong bias that only local variation is of importance [91]. This mitigates the effect of any slowly-changing process that may affect the EEG signal.

Ren and Wu implemented a convolutional DBN to perform unsupervised classification of BCI motor imagery data from a variety of competition datasets [130]. The convolutional DBN slightly outperformed other unsupervised techniques including common spatial pattern, multivariate adaptive autoregressive, and band power techniques [130].

Zheng and Lu investigated the importance of frequency-band and channel selection on affective state classification using a DBNs with 2 RBM layers and SVMs by using

a variety of input features including differential entropy, PSD, and features based on hemispheric asymmetrical signaling [185]. Zheng found that beta and gamma features were the most important for classifying affective state by examining the weights of the DBN [185]. The DBN weights were also used as a guide for downselection of electrodes to determine if models using only a subset of electrodes would perform as well as those incorporating all electrodes [185]. Four profiles containing 4, 6, 9, and 12 channels respectively were identified and used to train a SVM. In 3 of the 4 cases (6, 9, and 12 electrodes), the SVM trained on the reduced-electrode scheme outperformed the model trained with all electrodes [185]. In fact, the 12-electrode SVM model outperformed even the best DBN trained using all electrodes, achieving a classification accuracy of 86.65% [185]. However, it should be noted that the DBN was not highly regularized which may have contributed to the underperformance of the model with all electrodes included. This work indicates that similar areas of the brain are used to process emotional content across participants making it a reasonable expectation that similar functional areas may be activated in separate individuals with regard to operator workload. It also highlights the utility of using learned weights as a channel downselection metric and the importance of regularization.

Bashivan and Bidelman compared performance of wavelet entropy features to traditional mean PSD features for a four-level Sternberg working memory cognitive workload task by assessing classification accuracy of different feature combinations using SVMs and DBNs [15]. Wavelet entropy was computed by determining the Shannon entropy of a distribution of the normalized energy for each wavelet frequency band. This was found to be inferior to mean PSD for cognitive workload classification [15]. However, for the four-class problem, an SVM with fused wavelet energy and mean PSD resulted in the best cognitive load classification accuracy of 92.13%–a slight improvement over using only PSD features [15]. The SVM also slightly outperformed

a L1-regularized DBN constructed with one Gaussian-binary RBM, two binary RBMs, and a softmax layer [15]. A finding in this study was that DBNs could be used like an autoencoder to generate a reduced-sized optimal set of features [15].

Hajinoroozi, et al. applied a variety of DBNs to PSD features and ICA-transformed features to learn reduced-dimensionality features for input into LDA, SVM, boosting, and bagging classifiers [70]. The objective was to improve classification of driver cognitive state. The best performing features were generated by a DBN which used each channel independently as input and ignored correlations between channels [70]. This methodology resulted in improved results compared to baseline PCA techniques and was most effective when combined with a bagging classifier [70].

Zheng, et al. trained a DBN and integrated a Hidden Markov Model (HMM) to account for temporal dynamics associated with changing emotional states when performing emotion classification in a six-participant study [186]. Each participant was shown 12 four-minute long movie clips which were selected to elicit a particular emotion–6 positive and 6 negative [186]. A DBN with two Bernoulli RBM hidden layers was pretrained using contrastive divergence followed by supervised fine-tuning [186]. A HMM was stacked on top of the trained DBN and resulted in the best classification performance with 87.6% accuracy compared to the DBN which achieved 86.9% [186]. Both of these methods outperformed other classifiers including an extreme learning machine, a SVM, and a KNN classifier by small margins [186].

Turner, et al. compared results of using logistic regression on an array of simple time-series features to results obtained after feature transformation using a two hidden-layer DBN when performing classification of ten patients in a seizure identification study [165]. Both cross-participant and within-subject training was performed. Little difference was found between baseline logistic regression and DBN results for within-subject training, while a slight improvement when using the DBN was appar-

ent in the cross-participant classification case [165]. These results add to the corpus showing that feature transformations with DBNs or SAEs result in a mild cross-participant classification improvement, but little change for within-subject results.

Liu, et al. achieved excellent results using a bimodal deep autoencoder, where the contractive layers were trained as a DBN, to perform fusion of EEG and eye tracking data for emotion recognition using the SEED and DEAP datasets [111]. By combining PSD and differential entropy EEG features as well as eye movement features into a single high-level representation, large improvements in SVM classification accuracy were obtained compared to using all features directly–83% to 91% for the SEED dataset and approximately 14-25% improvements in classification accuracy for each of four types of emotion present in the DEAP dataset [111]. This implies that performing multi-modal feature fusion using a deep autoencoder could enable a synergistic mixing of features into a representation that is better than the sum of the parts which is important to keep in mind when designing a multi-modal workload model.

2.6 Summary

In the preceding sections, classification accuracies between 26% and 95% were reported across a variety of experimental setups, with the majority around 80% accurate. However, a clear trend was present showing that the more controlled and less complex the task environment, the better the results. To be useful in an operational setting, excellent performance in multi-task environments will be required. Rouse indicated that a 95% accuracy rate for workload estimation may be required for a system to be acceptable [131]. Parasuraman went further and suggested that if the system does not approach 100% accuracy that the costs of inaccuracy and lack of trust may lead to the system being unacceptable in the safety-critical flight environment [125]. This high level of required model performance highlights the challenge of

creating an acceptable model for the flight environment.

While traditional machine learning methods have been used to make predictions in multi-task environments, no deep learning techniques have been attempted. Since deep learning techniques afford the network architect the ability to better model spatial, frequential, and temporal dependencies in the data compared to traditional machine learning methods, significant gains in performance can be realized. Across a broad range of single-task studies, decreases in error of 50% or more were reported using deep learning techniques compared to traditional methods. Chapters IV and VI report similar reductions in error for multi-task settings.

The following gaps in literature are addressed in Chapters III - VI:

1. In each of the studies using deep learning, no attempt was made to address the challenge of day-to-day variability. This is explicitly addressed using LSTMs in Chapter IV.

2. While many researchers performed cross-participant modeling, none used deep learning techniques in a multi-task environment with networks explicitly designed to mitigate cross-participant differences. Chapter V focuses on this challenge.

3. Different training techniques have not been investigated for deep neural network modeling using EEG data. There are many ways to train neural networks using group models and ensembles. Chapter V characterizes the effect of training method on a variety of deep neural network architectures.

4. It is notable that all the literature discussed in this chapter used classification rather than regression methods. While a very limited amount of regression work has been performed in EEG analysis, as discussed in Section 6.2, no research

has used deep learning regression approaches to model phenomena using EEG data. This is accomplished in Chapter VI.

5. Characterization of the performance of Siamese-triplet networks using EEG data has not been attempted. As previously mentioned, these networks are described in Section 6.4.

The reader can see that the contributions outlined in Chapter I represent progress in overcoming the algorithmic barriers to operational psychophysiological workload estimation above and beyond other state-of-the-art research. In the following chapters, several feature engineering, ensemble, traditional machine learning, and deep learning techniques are applied to data acquired during multi-task, non-stimulus aligned experiments. The combination of techniques resulted in statistically significant and meaningful improvements in addressing the challenges of day-to-day variability and cross-participant applicability.

III. A New Feature For Cross-day Psychophysiological Workload Estimation

3.1 Introduction

As teams of humans and machines become ubiquitous, the machine's ability to understand the functional state of the human operator becomes ever more important. Machines need to know the state of the human operator in order to make good decisions which improve team performance. The potential benefits of such a system are far reaching and could drastically change the way individuals learn, train, and perform jobs while paving the way for advances in adaptive automation [125].

Operator functional state assessment (OFSA) techniques fall into two categories– objective and subjective. Subjective methods ask the operator to self-assess his state or have an observer provide an assessment. Subjective measures require interrupting the operator during the task, and thus are often incompatible with real-world operations outside of the laboratory environment. Alternatively, objective techniques do not interrupt the operator's task; they use sensors and machine-learned models to infer operator state. A key area in OFSA is operator workload assessment, which allows the machine teammate to measure the workload the human teammate is currently experiencing. Accurate operator workload assessment enables successful task allocation decisions within the team.

To build models for operator workload assessment, the relationship between physiological signals and workload must be learned. Physiological signals such as electroencephalograph (electroencephalograph (EEG)) are measured while an individual performs a set of tasks of different known workloads. Machine learning is then used to build a model mapping the signal to the workload level for the individual.

One desired property of Operator Functional State Assessment (OFSA) classifiers

is that the classifiers are resilient to the day-to-day variability of psychophysiological features. Such classifiers are said to be stable [54] or low variance because small variations in input during training result in negligible differences in the resulting classifier [88]. Producing stable classifiers for operator functional state over temporal periods greater than a single session lasting several hours has proven difficult due to the nonstationarity of EEG feature distributions [53].

The goal in this chapter is to advance objective operator workload assessment by proposing improvements to the machine learning modeling methods which improve classifier performance. The primary objective of this study was to develop a method which is more resilient to the day-to-day variability of EEG data. It is shown that using an additional set of features based on the variance of the EEG frequency-domain power enables a cross-day workload classification accuracy gain of 5.8% over models trained with features which use the mean of the power alone.

3.2 Related Work

There is a growing body of research evaluating cross-day variability, and several studies have used the same workload dataset described in Section 3.3 [54]. However, much of the research performed on this dataset was related to cross-subject variability rather than inter-day variation of individual participants, while other work evaluated both simultaneously, thus confounding the results. This section focuses on only work which addressed temporal non-stationarity within subject.

Researchers have explored the causes of temporal non-stationarity. Jahns identified a number of factors related to the participant's mental state that contribute to inter-day variability: participant motivation as well as mental and physical readiness [87]. Christensen, et al. identified two of the major reasons why classification accuracy often decreases across days: poor classifier generalization caused by over-

fitting to particularities on a given day, or changes in the feature distributions of easy versus difficult tasks across days [39]. Other reasons for cross-day performance decrease include the curse of dimensionality, and tradeoffs in bias and variance [116]. As data dimensions grow during feature generation, the required length of the dataset grows faster than the feature space [116]. Lotte, et al. recommend using a minimum of 5-10 times as many training samples as features to achieve stable (low variance) classification results. Unfortunately, the size of the feature space, and the time and resource-intensive nature of collecting EEG during human experiments, often make collecting this many training samples impractical [116].

In the workload literature using this dataset, only one research team focused research solely on the performance of classifiers in cross-day situations. Christensen et al., performed cross-day classification of high versus low workload by training a classifier on multiple days to improve classifier training using the same dataset as this chapter [39]. Christensen identified that if training data was used from just a single day, the classifier may be overfitting to peculiarities in the data particular to that day. However, it was hypothesized that if multiple days were used for cross-validated training, a robust set of features may result in a more generalizable model [39]. Christensen divided the 5-days worth of data into all permutations of possible training days for a particular individual with each test set being the complement of the corresponding training set. Christensen, et al. trained on all permutations of 1 day, 2 days, 3 days, and 4 days, using cross-validation, while testing on the remaining days not included in the training data. Then the average across individual participants and across permutations of the same number of days was calculated to attain an overall accuracy for all possible 1, 2, 3, and 4 training day combinations. Cross-validating at the day-level was a reasonable assumption to establish a baseline accuracy with a limited quantity of data. Christensen found that Linear Discriminant

Analysis (LDA) performed the poorest with classification accuracy increasing from approximately 59% to 66% as the number of training days increased from 1 to 4. The SVM exhibited decreased performance as the number of days in the training set increased. Classification accuracy for 1 training day was approximately 72% while accuracy for 4 training days dropped to 68%. The neural network performed the best and appeared able to incorporate data from multiple days to increase classification accuracy; showing a rise in accuracy from approximately 73% to 83% between the 1-day training set and the 4-day training set. They found a significant reduction in classification accuracy across-days as illustrated by a best-case reduction from 99% to 83% when four days were used to train the Artificial Neural Network (ANN) with testing on a fifth day compared to single-day train-test models [39]. These results indicate that ANNs have some innate ability to handle non-stationarity in feature distributions. However, the significant reduction in classification accuracy compared to single day results indicate that other models and feature generation techniques should be investigated.

Another shortcoming of current literature using this dataset is the lack of identification of cross-day salient features. The majority of approaches used all available predictors in a model rather than using a technique such as stepwise selection of features. Firpi and Vogelstein were a notable exception in their use of a particle-swarm optimization-based feature selection procedure which assessed the importance of a variety of generated features [57]. However, they found their features to perform unacceptably for classification of high workload test points with mean classification accuracies ranging from 6.74%-23.21% [57]. Since this chapter is focused on prediction of both low and high workload levels, their salient features will not be covered here.

Cross-day feature salience research using other datasets has yielded varied results.

In a separate workload experiment, Gevins et. al. demonstrated that increases in frontal-midline theta power and decreases in parietal alpha power were associated with increases in cognitive workload [62]. Borghini, et. al. further confirmed these results in a survey paper which reported these same trends across a multitude of experiments during high workload trials compared to low-workload trials [26]. However, in a Multi-Attribute Task Battery (MATB) workload experiment, Laine, et. al. found that gamma features in the 31-40 Hz range and beta features in the 13-30 Hz range were the most salient features [101]. They used a stepwise discriminant analysis procedure and a signal to noise ratio measure coupled with an ANN classifier for workload classification [101]. Bowers, Christensen, and Eggemeier observed similar results when examining workload transitions associated with changing task difficulty levels during execution of the MATB [27]. They noted that changes in temporal gamma oscillations were highly correlated with changes in task difficulty level [27]. Note that all of these results are from studies using only the mean of the power distribution of EEG over a time segment; the mixed feature salience in prior work suggests that finding a good set of features remains an open research problem–one which is explore in the remainder of this chapter.

3.3 Dataset

Methods in this study seek to maximize the cross-day classification performance: build a model trained on past data to make predictions on future operator workload. Data for this study was gathered from a prior study completed in 2011 [54]. Eight participants completed scenarios within the MATB [44] across five test days spread out over a month-long period. The task difficulty in MATB was manipulated to induce three levels of workload in the operator: low, medium, and high. Data was selected from the low and high workload conditions resulting in a total of 30 conditions–15

low and 15 high–for each individual. Only six of the participants were used in this study due to missing data from two of the original eight participants.

For each of the eight participants in the study, horizontal electrooculogram (HEOG), vertical EOG (VEOG), and 19 channels of EEG voltages (according to the International 10-20 System) were sampled at 256 Hz. On each of the five days, each participant performed three five minute trials at low, medium, and high workload for a total of nine trials per day. Trials were presented in a random ordering with transition periods in between. Each participant completed a 30 second resting baseline at the start of each session prior to the MATB task.

3.4 Methodology

The goal in this study was to develop a workload model which had robust cross-day prediction performance, and determine which features of the data were most salient for the workload model. The first step in developing the model was to preprocess the EEG data by converting it from time-series into frequency spectrum data for the clinical EEG bands. Next, the frequency data was carved into multiple short time segments for each day. In each clinical band, for each segment, mean power and power variance were computed on each of the segments to provide features suitable for supervised machine learning classification. A machine learning model was built for each individual participant. Each machine learning model was trained using the first four days of data from an individual. Each individual's model was assessed using the fifth day's data as a hold-out test set for that individual. Classification accuracy and an ordered set of salient features was produced for each individual. The following paragraphs detail the process outlined above.

Raw EEG data was transformed into features in clinical frequency bands to conduct time-frequency analysis. EEG rhythmic signals are typically analyzed in the

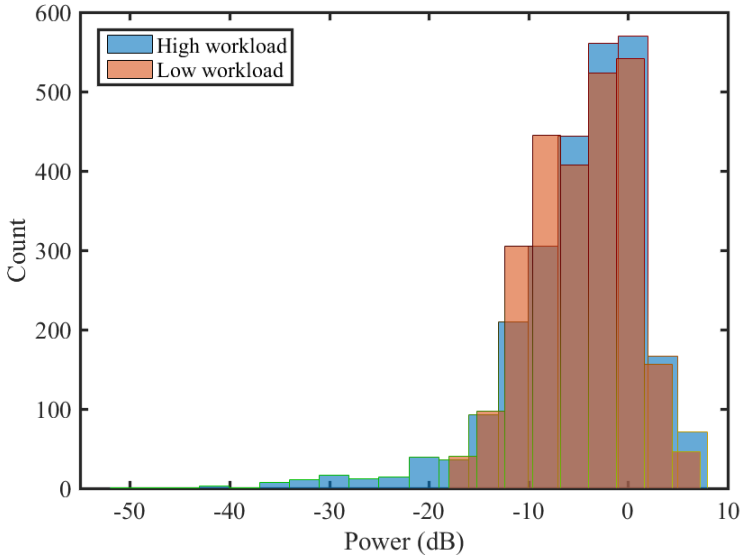


Figure 2. Ten second F8 theta band (4-8Hz) power distributions for low and high workload conditions experienced by participant 1.

following frequency bands: delta (1-4Hz), theta (4-8Hz), alpha (8-14Hz), beta (15-30), and gamma (30-55Hz). The power spectral density was determined for 30 points spread out over a logspace from 3Hz to 55Hz by extracting power from complex Morlet wavelets [42]. Each wavelet was 2 seconds in length and the number of wavelet cycles increased logarithmically from 3 to 10 in conjunction with the frequencies. Mean power in each band was determined by averaging each power value for the evaluated frequencies within each of the clinical bands. Power was then aggregated over a ten second sliding window with 9 seconds of overlap, allowing for a new update each second. The mean and variance of the power distribution in each of these ten second windows was calculated for each of 19 EEG electrode sites across five frequency bands yielding 190 features for each second. There were approximately 9,000 observations per individual across all five days.

Figure 2 illustrates why variance was selected as a feature. This figure shows a ten second power ratio distribution of low and high workload conditions compared to the baseline for participant 1's F8 theta electrode. Power ratio is displayed on the

horizontal axis while the number of counts for a particular bin over the 10 second period is shown on the vertical axis. The two distributions qualitatively appear very similar and have means of -4.57 dB and -4.78 dB for low and high workload respectively. However, the variance is much more discriminating due to the long tail in the high workload condition, yielding variance values of 23.77 and 46.18 for low and high respectively. While not explored in this research, Figure 2 also suggests that using skewness (low = -0.18; high = -1.56) and kurtosis (low = 2.37; high = 7.32) as features in future work could further enhance classification accuracy since these values reveal a strong difference between low and high workload conditions.

Once features were produced, four classifiers–LDA, random forest, K-Nearest Neighbors (KNN), and an ensemble classifier–were trained on the first four days of data for each individual with the last day used as a hold-out test set. Two separate datasets were produced: One containing just the means of the preceding 10 second power distributions by node and frequency band, and one containing features for both the mean and variance of the aforementioned distributions. The subsequent methodology was identically used for both datasets.

The LDA results provided a common point of comparison between Christensen's study and this study and serve as a benchmark to compare the other two algorithms' results against. Model selection for LDA was performed using forward stepwise selection based on training accuracy, while 12-fold cross-validation–one fold per session–was used to select the best functional form following the one-standard error rule [88]. Forward stepwise selection began with a null model with no predictors and greedily added new features one at a time based on the maximum relative increase in classification accuracy of all possible new features [88]. Once a minimum cross-validation error point was found, the functional form of the model was selected as the one with the least number of features that fell within one-standard deviation of the cross-validated

minimum error. Then, the final model was trained using all data from the first four days to achieve maximal generalizability.

The random forest was trained using all features. Random forests have two hyper-parameters which were tuned: the number of features m and the number of trees t. The number of features randomly resampled at each split was varied about the square root of the number of features using 12-fold cross-validation on the training data. It is important to note that cross-validation was used for model tuning rather than out of bag error due to the way the non-stationarity affected the out-of-bag error. With non-stationary feature distributions, using out-of-bag error results in too many temporally adjacent datapoints to be evaluated and artificially reduces cross-validation error to values that are not representative of error present in the hold-out dataset. Due to insignificant variation across cross-validated results, the final values for m were selected as the floor of the square root of the number of features for the mean and variance dataset: $m = 13$. This allowed for a fair comparison of performance between the random forest models trained using mean-only features and those created from mean and variance features. The random forest was then trained with a varying number of trees and a value of 500 trees appeared to offer the best results and was stable. Finally, predictions were made on the test sets.

KNN models determine that the observation has the same class as the largest class represented in it's K nearest neighbors in feature space. The model's hyper-parameter K is often determined experimentally. The KNN model was trained using 12-fold cross-validation while varying K until performance stabilized. The final value of K varied between 15 and 19 depending on the individual being modeled.

Inspired by Schapire's work showing weak learners can be combined to increase accuracy [138], an ensemble classifier was also developed. The ensemble combines each of the three simple classifiers by using an average rating from each classifier's

most recent 15 classifications. For each observation, a majority vote of the weak learners is used to determine the workload class.

Both the LDA stepwise feature selection procedure and the random forest provided a measure of feature saliency. The random forest ranked feature saliency based on the mean Gini decrease associated with a particular feature [88]. Receiver Operating Characteristic (ROC) curves were created for each classifier for the mean and variance models and Area Under Curve (AUC) results were compared to determine which algorithm performed the best given the condition of non-stationarity. Classification results were averaged across individuals enabling by-model comparisons. Finally, paired t-tests were used grouping by participant-algorithm combinations to examine by-feature set results.

3.5 Results

Analysis of the results required comparing models trained and tested using only mean power features compared to models which incorporated mean and variance features. A one-sided paired sample t-test with $\alpha = .05$ was used to determine whether there was an increase in mean classification accuracy for the mean and variance feature model compared to the mean-only model. The one-sided test resulted in a p-value of $< .0001$. The null hypothesis was rejected and it was concluded that the inclusion of variance features significantly improved classification accuracy across models. The mean increase was 5.8% with a 95% confidence interval spanning from a 3.6% to 8.1% increase in accuracy.

Table 6 shows results for each model by participant and dataset. Only three mean and variance models performed worse than the corresponding mean-only model. The worst performance was associated with participant 5's mean and variance LDA model. Participant 5 exhibited anomalous LDA results with an accuracy of only

Table 6. Classification accuracy for each model by participant and feature set.

	Mean Only				Mean and Variance			
Participant	LDA	KNN	RF	Ensemble	LDA (Δ)	KNN (Δ)	RF (Δ)	Ensemble (Δ)
1	0.681	0.605	0.627	0.695	0.695 (+0.014)	0.672 (+0.067)	0.739 (+0.112)	0.771 (+0.077)
2	0.757	0.699	0.763	0.739	0.864 (+0.107)	0.820 (+0.121)	0.851 (+0.088)	0.853 (+0.114)
3	0.912	0.775	0.887	0.908	0.896 (-0.016)	0.799 (+0.024)	0.912 (+0.025)	0.910 (+0.002)
4	0.625	0.702	0.763	0.828	0.810 (+0.185)	0.689 (-0.013)	0.834 (+0.071)	0.891 (+0.063)
5	0.634	0.616	0.663	0.661	0.391 (-0.243)	0.702 (+0.086)	0.740 (+0.077)	0.672 (+0.011)
6	0.622	0.675	0.659	0.708	0.713 (+0.091)	0.675 (+0.000)	0.673 (+0.014)	0.728 (+0.020)
Average	0.705	0.679	0.727	0.757	0.728 (+0.023)	0.726 (+0.047)	0.792 (+0.064)	0.804 (+0.048)

39.1%. This poor performance is also illustrated in the below-chance AUC metric of .390 shown in Table 7. Additional analysis confirmed that the source of the anomalous performance was due to excessive feature pruning during application of the one-standard error rule when selecting the best cross-validated functional form. The addition of one more feature would have increased classification accuracy on the test set by an additional 20%. Because of the disproportionate effect the anomalous point exerted on the statistical analysis, it was excluded from the paired t-test analysis. However, even if the point were included, statistically significant results were present well below the $\alpha = .05$ level, but a non-parametric approach became necessary due to a gross departure from normality in the lower tail of the distribution. Interestingly, the ensemble method's classification accuracy for the mean and variance model was adversely affected by participant 5's LDA results since the accuracy of that classifier was worse than chance, yet it still yielded a small improvement over the mean-only feature set model. The KNN mean and variance model for participant 4 and LDA mean and variance model for participant 3 both exhibited slightly worse performance than the mean-only model.

The ensemble model was consistently the best or near the best model across all participants with the exception of participant 5 in the mean and variance case due to the impact from poor LDA results. LDA results were generally quite good and competed with random forests for the best non-composite model results for all

Table 7. ROC AUC for LDA, random forest, and KNN models by participant and feature set.

	Mean Only			Mean and Variance		
Participant	LDA	RF	KNN	LDA (Δ)	RF (Δ)	KNN (Δ)
1	0.717	0.706	0.634	0.764 (+0.047)	0.812 (+0.106)	0.737 (+0.104)
2	0.832	0.839	0.783	0.937 (+0.105)	0.926 (+0.087)	0.915 (+0.132)
3	0.949	0.959	0.860	0.958 (+0.009)	0.969 (+0.011)	0.881 (+0.021)
4	0.677	0.862	0.774	0.837 (+0.160)	0.890 (+0.028)	0.764 (-0.009)
5	0.717	0.723	0.682	0.390 (-0.328)	0.797 (+0.074)	0.796 (+0.114)
6	0.660	0.731	0.736	0.781 (+0.121)	0.746 (+0.015)	0.715 (-0.021)

participants except participant 5. The LDA classifier using the mean and variance feature set compared favorably to Christensen's LDA model improving accuracy by approximately 7% to 72.8%. The KNN classifier attained 72.6% while random forest displayed 79.2% accuracy which was the best single classifier in this study. Table 7 shows that the KNN models were generally the worst performing. The curse of dimensionality was the likely cause since no feature reduction was performed prior to using the KNN classifier. The time-smoothed majority ensemble classifier achieved a classification accuracy of 80.4% yet fell short of Christensen's ANN which achieved 83% accuracy on this dataset confirming that neural networks may outperform other traditional classifiers for cross-day classification of EEG data. These results show that cross-day classification accuracy can be improved by combining different feature generation and model training techniques. Accuracy can be further improved by using temporal smoothing and ensemble techniques.

The secondary objective was to identify the important features from random forest and LDA models. Participant-averaged random forest results in Table 8 show that cross-day feature saliency results were in-line with previous results presented by Laine [101] and Bowers [27] with temporal gamma features being the most important. McDonald and Soussou noted that electromyograph (EMG) artifacts associated with

Table 8. Top: Rank order random forest feature importance for each participant and associated mean Gini decrease for each feature. The cross participant saliency ranking shows the most important features and average mean Gini decrease across all participants. Bottom: Each participant's LDA model features in rank order based on forward stepwise selection.

Random Forest Salient Features (Gini Decrease)														
Participant 1		Participant 2		Participant 3		Participant 4		Participant 5		Participant 6		Cross Participant	
FP2_gamma_var	93.11	T5_gamma_mean	151.04	O2_theta_mean	165.01	O2_gamma_mean	102.81	T6_gamma_mean	308.24	T6_gamma_mean	83.90	T6_gamma_mean	112.06
F7_delta_mean	80.24	T6_gamma_mean	144.95	CZ_theta_mean	103.56	O2_beta_mean	97.74	T5_gamma_mean	178.64	O2_gamma_mean	81.27	T5_gamma_mean	84.65
FP2_beta_var	66.17	O1_gamma_mean	137.13	O1_gamma_mean	95.21	C3_beta_mean	66.89	FP1_theta_var	164.87	O1_gamma_mean	81.04	O2_gamma_mean	75.23
O2_beta_mean	61.90	O2_gamma_mean	122.05	T5_gamma_mean	91.26	FP1_beta_mean	58.95	FP2_theta_var	147.36	O2_beta_mean	72.62	O1_gamma_mean	66.55
O2_gamma_mean	60.01	FP1_delta_var	105.38	O1_theta_mean	85.59	FP1_gamma_mean	53.30	T6_beta_mean	141.17	O2_gamma_var	72.53	FP2_delta_var	60.03
T6_gamma_mean	57.81	FP2_delta_var	93.08	PZ_theta_mean	85.36	T6_gamma_mean	52.27	T5_gamma_var	122.39	C3_alpha_var	66.69	FP1_delta_var	57.59
O2_alpha_mean	57.80	O1_theta_mean	80.80	P4_theta_mean	78.75	F3_gamma_mean	52.11	T5_beta_mean	113.05	T5_gamma_var	63.80	FP1_theta_var	53.10
F7_gamma_mean	53.21	P3_gamma_mean	80.13	C3_theta_mean	72.38	F7_gamma_mean	51.65	FP2_delta_var	89.61	PZ_gamma_var	62.31	O1_theta_mean	50.77
F8_delta_mean	53.07	T4_gamma_mean	79.84	FP2_delta_var	71.49	F3_beta_mean	51.48	FP1_delta_var	85.13	FP2_delta_var	60.23	O2_theta_mean	50.70
F7_beta_mean	50.75	F8_delta_var	73.57	FZ_theta_mean	66.72	O2_alpha_mean	51.42	O1_theta_mean	84.68	C3_beta_var	60.14	T5_gamma_var	49.44
LDA Salient Features													
F8_theta_var		FP2_delta_var		O2_theta_mean		O2_beta_mean		T6_gamma_mean		T5_gamma_var			
T6_alpha_var		O1_gamma_mean		O1_gamma_var				T5_gamma_var		O2_gamma_mean			
CZ_theta_mean		C4_theta_mean		O1_beta_mean						FP2_delta_var			
F7_delta_mean		T5_gamma_var		FP1_delta_var						C3_beta_mean			
				FP1_alpha_var									
				PZ_theta_mean									

jaw clenching or upper back tightening are often associated with increased workload and can influence signals in the gamma frequency band [119]. Since no artifact correction methods were performed on the EEG dataset, this may be a factor in the observed increased temporal gamma feature utility for workload classification. Future work can determine if artifact removal in EEG eliminates potentially important information in applied workload estimation experiments. In general, an increase in mean frontal-midline theta power and decrease in parietal alpha power were not found to be salient features across models as observed by other researchers [62, 26]. Further examination of feature saliency in Table 8 reveals that features varied from one individual to another and that different algorithms identified different features as being important. In general, random forest models tended to find a majority of features in the higher frequency gamma and beta bands. LDA on the other hand had a more even mixture of features from the different bands.

Approximately half the features used in the LDA model were also found in the top ten features for the corresponding random forest model demonstrating that feature salience varied based not only on the individual, but also on the classifier used. This observation, coupled with the notion that temporal smoothing should improve

results, provided additional rationale for the improved performance noted in the time-smoothed ensemble since each classifier in the ensemble appeared to contain sufficiently decorrelated features. A final important finding was that variance features accounted for 28.3% of the top ten features for the random forest models and 47.7% of the salient LDA features; further bolstering the importance of the variance-based features.

3.6 Conclusion and Future Work

Statistically significant results were obtained confirming the importance of variance of frequency-domain power distributions for cross-day workload classification. These results were demonstrated using an LDA, random forest, KNN, and ensemble algorithm for each participant by comparing mean-only dataset results with mean and variance results. Additionally, when evaluating feature saliency, variance features accounted for a non-trivial portion of salient features for both the random forest and LDA models further confirming the importance of these new features. In examining feature saliency for LDA and random forest models, approximately half of the salient features were unique across models which inspired the creation of a time-smoothed ensemble model. This model generally performed the best yet fell short of Christensen's ANN implementation. Comparing LDA results to previous work demonstrated how training and feature generation techniques can yield appreciably different results given the observed 7% increase in classification accuracy.

There is an abundance of future work to be explored. In examining the EEG frequency power ratio distributions, the large differences between the low and high workload values of skewness and kurtosis in Figure 2 suggests these statistics may form the basis of future salient features. Evaluating classification accuracy and feature salience of models using all combinations of the first four central moments and their

interactions would provide additional evidence of the utility of these feature generation techniques. Evaluations of these proposed features should be extended to ANN models and more complex ensembles. KNN classifier performance could be further evaluated with a reduced feature set and simultaneous cross-subject/cross-day models should be created to extend the generalizability of the models. Finally it appears that different algorithms may be required to achieve an acceptable classification accuracy for handling inter-day non-stationarity outside the laboratory environment. Future work should consider using a temporally stateful representation such as recurrent neural networks to address this non-stationarity.

IV. Deep Long Short-Term Memory Structures Model Temporal Dependencies Improving Cognitive Workload Estimation

4.1 Introduction

Teams composed of both humans and machines can potentially work together to mitigate their respective inherent weaknesses. A computer's strength is manifested in its ability to quickly and correctly compute answers, while humans exhibit superior flexibility of response to unexpected situations. Thus, Human-Machine Teams (HMTs) promise to mitigate inherent limitations on computational decision-making in all-human teams while simultaneously reducing the brittleness and inflexibility of fully-autonomous systems [47]. Team outcomes are improved when one agent (human or computer) assists another in the right way at the right time [38]. For computers to help humans in HMTs, they must know the human's cognitive state; this knowledge can be obtained through Operator Functional State Assessment (OFSA) [179]. Several methods of OFSA exist, which can generally be broken into two classes of measures–objective and subjective. Subjective measures usually ask the operator to evaluate themselves either during or after the task, while objective measures use a physiological sensor such as electroencephalograph (EEG) or electrocardiogram (ECG) to provide inputs to an algorithm that assesses the operator's functional state. The benefit of objective measures is that they do not interrupt the operator while performing the task [172, 173]. Continuous non-interrupting state assessment is an important characteristic for viable HMTs outside the laboratory.

A key subarea of research within OFSA is mental workload estimation. Enabling the machine in a human-machine team to unobtrusively and continuously ascertain the operator's mental workload is the first step in closing the machine-to-human augmentation loop. In order for augmentation to be effective, it must be driven

by an accurate estimate of mental workload [38]. A common method for estimating mental workload is to first use statistical machine learning to fit a model which enables prediction of mental workload from the physiological signals, and then use that model to make mental workload estimates from newly-gathered physiological signals [178].

The utility of an OFSA system will depend on the benefits of accurate assessment and the costs of errors. This cost-benefit trade-off will be application-specific and different for correctly identifying high and low workload states depending on the types of augmentation tied to a given state and the consequences of incorrect/inappropriate activation or lack of activation. These errors directly impact an operator's trust in the automation, in-turn affecting future utility of that automation in a closed loop-fashion [105]. Rouse et al. [131] indicated that a 95% accuracy rate for workload estimation may be required for a system to be acceptable. Parasuraman et al. [125] went further and suggested that if the system does not approach 100% accuracy then the costs of inaccuracy and lack of trust may lead to the system being unacceptable, especially in safety-critical environments.

Unfortunately, current state-of-the-art systems are not yet able to achieve the required accuracy, due in part to the challenge of temporal non-stationarity in psychophysiological signals. This challenge relates to variation over longer periods of time and dependence within shorter periods. Both can negatively impact the generalizable long term accuracy of workload assessment systems [38]. Within shorter spans of time, signals tend to exhibit hysteresis or serial dependence. This suggests that there is inherent structure in the statefulness in the brain that can be exploited with appropriate machine learning techniques. While it is difficult to attribute this dependence to any discrete set of factors, some of the likely possibilities include consistency in default mode activity [129] and hysteresis exhibited by most physiological systems.

In the context of machine learning, temporal non-stationarity can be addressed in two ways. The first is through feature generation or selection. A better set of features will exhibit less long-term non-stationarity and will lead to better model performance. In this work, several feature generation techniques are examined to determine empirically if certain feature sets are superior to others. The second way to address non-stationarity with machine learning is to use algorithms that make different assumptions about the nature of the data being processed. As it stands, most published research on operator workload estimation implicitly assumed temporal independence from one time segment to the next. This is likely a poor assumption due to both the factors discussed above as well as longer term effects such as fatigue and performance hysteresis with mental workload transitions [84]. An example from aviation illustrates this nicely. If a pilot has just completed flying an instrument approach in instrument meteorological conditions (IMC) when an unexpected emergency requires attention, pilot workload will increase differently than if the pilot had the same unexpected emergency arise following a period of autopilot-on flight at cruising altitude in visual meteorological conditions (VMC). This simple example illustrates that what has happened in the recent past temporally, matters for operator workload assessment.

Machine learning algorithms that consider past information as well as current information when fitting models should perform better. Such algorithms must be able to learn a temporal representation of the data. A common model used for modeling temporal data is the Recurrent Neural Network (RNN). RNNs are neural networks that are able to learn sequences that are not composed of independent, identically distributed observations [65]. Rather, they are able to elicit the context of observations within sequences and accurately classify sequences that have strong temporal correlations [65]. Historically, RNNs had limitations when training models with more than 10-20 time steps which led to poor performance. Incorporating longer

time-series data streams would cause computational sensitivity problems that stymied RNN training.

Recent developments have resulted in RNN architectural and training advances which mitigate these computational problems and allow much longer temporal sequences to be processed. One approach is the Long Short-Term Memory (LSTM) layer. LSTM architectures extend the length of sequences that can be considered by a RNN by overcoming computational sensitivities encountered during backpropagation [81]. For these reasons, they may offer improved workload classification accuracy over other methods when using EEG data. With these improvements in machine learning, there is no longer a reason to avoid incorporating temporal context in a workload model. These machine learning developments are capitalized upon in this research.

The primary contribution of the research in this chapter is demonstration of significantly improved cross-day workload classification accuracy by integrating contextually relevant algorithmic architectures with improved feature generation techniques. All combinations of mean, variance, skewness, and kurtosis of frequency-domain power distributions are statistically evaluated and a variety of RNN architectures are contrasted, to include deeply stacked LSTMs, with baseline algorithms and features. Both linear and Radial Basis Function (RBF) Support Vector Machines (SVMs) and single-layer feedforward Artificial Neural Networks (ANNs) using mean-only features are used as baseline cases. It is shown that by accounting for temporal dependence using deep LSTM models trained with new feature combinations, cross-day workload estimation accuracy can be maximized resulting in a 58% reduction in classification error over baseline methods and a 59% decrease in error compared to the best published results for this dataset.

4.2 Background and Related Work

Temporal non-stationarity of electroencephalograph (EEG) signals within individuals is likely caused by a large number of intrinsic and extrinsic factors. Participant motivation and mental or physical readiness are examples of some intrinsic factors; extrinsic factors include significant differences in EEG electrode placement, changes in conductance, and different motion artifacts [39, 87, 114]. Due to the challenge of handling these factors, cross-day non-stationarity of EEG signals has motivated a number of related studies including several using the same dataset described below.

4.2.1 Dataset.

Data for this chapter's study was used in the 2011 Cognitive State Assessment Competition [54] and was recorded during a prior human research study performed by Wilson et al. [176]. Eight participants completed scenarios within the Multi-Attribute Task Battery (MATB) [44] environment across five test days spread out over a month-long period. Monitoring, communication, resource management, and tracking tasks were presented and manipulated to induce three levels of difficulty: low, medium, and high [39, 176]. Resource allocation errors, monitoring task reaction times, and communication response times were recorded and used to validate that participants experienced distinct low and high difficulty levels. Participants were trained to asymptotic proficiency prior to the first test day [176].

For each participant, horizontal electrooculogram (HEOG), vertical EOG (VEOG), and 19 channels of EEG voltages (according to the International 10-20 System) were sampled at 256 Hz. On each of the five days, each participant performed three five-minute trials at low, medium, and high difficulty for a total of nine trials per day. Trials were presented in a random ordering with transition periods in between. Each participant completed a 30 second resting baseline at the start of each session prior to

the MATB task. Only six of the participants were used in this study due to missing data from two of the original eight participants [77]. Similar to Christensen et al. [39] and Casson [32], data in this study was selected from the low and high workload conditions resulting in a total of 30 conditions–15 low and 15 high–for each individual.

4.2.2 Within-Participant Cross-day Variability.

Many researchers using this dataset focused their efforts on cross-participant variability or a combination of cross-participant and cross-day models rather than independently investigating cross-day variability within individuals. This section will only consider those works that exclusively examined within-participant cross-day variability.

In the workload literature using this dataset, three research teams focused solely on the performance of classifiers in cross-day analyses. Hefron and Borghetti evaluated the variance of frequency-domain power distributions as a feature and found it to improve accuracy for random forest, Linear Discriminant Analysis (LDA), and K-Nearest Neighbors (KNN) classifiers for the two-class–high versus low workload–problem. Models built with variance and mean features exhibited a 5.8% increase in accuracy over models built using only the mean of frequency-domain power distributions [77]. In their study, it was postulated that the use of skewness and kurtosis as features could generate further gains in classification accuracy [77].

In another study, Christensen et al. evaluated cross-day classification performance of low versus high workload by training three classifiers: LDA, Support Vector Machine (SVM), and a single-hidden-layer feedforward Artificial Neural Network (ANN) [39]. Two SVM implementations were used–one used a radial basis function kernel and the other a linear kernel. The authors noted the linear kernel performed best and was used for reporting results. One portion of the experimental design used

leave-one-out cross-validation, holding out one day's data in each case. This procedure produced five separate testing periods that were used to evaluate algorithmic performance. Of note, the SVM achieved 68% accuracy, while the ANN attained 83% accuracy [39]. The results demonstrated significantly better cross-day performance for the neural network compared to more traditional machine learning techniques, namely LDA and SVM classifiers. These results indicated that neural networks may have more capacity to handle non-stationarity of feature-to-target mappings.

In a study very similar to Christensen's, Casson [32] evaluated the effect of temporal stability of feedforward ANNs trained on EEG signals with differences of seconds, minutes, hours, and days between collection of training data and testing data. While other interesting results were presented in the paper, the most relevant to this chapter concerned participant-specific, leave-one-session-out, cross-validated models. These models were aimed at producing excellent cross-session predictive performance and attained an average classification accuracy of 73%. Differences in preprocessing, training, and network architecture contributed to the differential between these results and ours.

In all of these investigations, the demonstrated classification accuracy fell short of recommendations for use in many operational settings. While accuracy requirements will be task specific, as discussed in Section 5.1, neither Rouse's desired 95% accuracy rate [131] nor Parasuraman's near 100% accuracy [125] condition were met for workload estimation. Since all of these methods assumed temporal independence between time segments, new models that account for temporospatial dependencies and the use of new features should be explored.

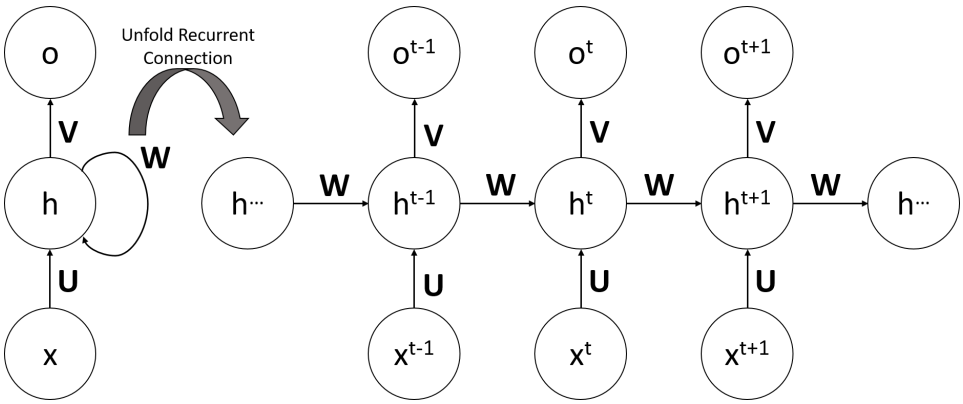


Figure 3. Temporal unfolding of the recurrent unit in the computational graph. On the left side is the cyclic graph. All cycles have been removed from the graph on the right side by unfolding the cyclic graph in time. Note the weight matrices U, V, and W are shared across time.

4.2.3 RNNs and LSTMs.

Deep neural networks have been used extensively to achieve state-of-the-art results across a wide range of categories including image recognition, speech recognition, translation, and image captioning [65, 66, 99, 167]. Recurrent Neural Networks (RNNs) are a type of deep network where depth is added via a recurrent connection in the hidden layer which is used to process sequential data. Processing sequential data is fundamentally different than processing independent identically distributed observations because of the distributional dependence on the sequence. In a traditional feedforward ANN, each observation is processed individually and the network state does not persist while other potentially sequentially-dependent points are processed. Conversely, in a RNN, recurrent connections pass state information across time steps allowing previously processed observations to affect the subsequent observations [110].

Feedforward ANNs have a directed acyclic computational graph–one layer feeds to another with no cycles. RNNs can be understood as an extension to the feedforward structure allowing for cycles in the graph structure. Figure 3 shows how very deep computational graphs can be created when the recurrent connections are unfolded

in time. The process of unfolding a recurrent network simply requires removal of all cycles in the graph to form a directed acyclic graph [17]. To allow these networks to learn important pieces of information that may be located at different positions in the sequential data, the input, recurrent, and output weight matrices' parameters are shared [17]. If different parameters were used for each time step, no generalization across time would be possible; sharing parameters allows the network to keep track of state. Since temporal non-stationarity of EEG signals across days is a challenge, it was important to ensure the temporal sequences supplied to each model did not exceed periods of time over which feature-to-target non-stationarity would occur.

In addition to the depth of a RNN due to unfolding over time, depth can be added to the network when recurrent layers are stacked in a sequence-to-sequence fashion. This means that the output from one layer of a RNN returns a sequence of vectors which form the input to the next layer. Graves et al. demonstrated that deeply layering RNNs has a more beneficial effect than merely adding memory cells [66]. The lower layers transform input sequences into more easily learned representations in the higher layers, resulting in better classification accuracy [17].

One problem with the simple recurrent structure shown in Figure 3 is a result of shared weights which produce vanishing and exploding gradients during backpropagation through time. This limits the sequential depth of simple recurrent units to sequences of no greater than 10-20 observations because the signal from distant positions will not propagate to the current time [17]. A significant innovation which solved this problem was Long Short-Term Memory (LSTM) [60, 81]. LSTM provides the algorithm fine control over what is put into memory and removed from memory in the hidden layer. This is achieved by a combination of three gates which control flow into and out of the memory cell: an input gate, a forget gate, and an output gate. Each gate works by using an element-wise sigmoid function, σ, to scale each

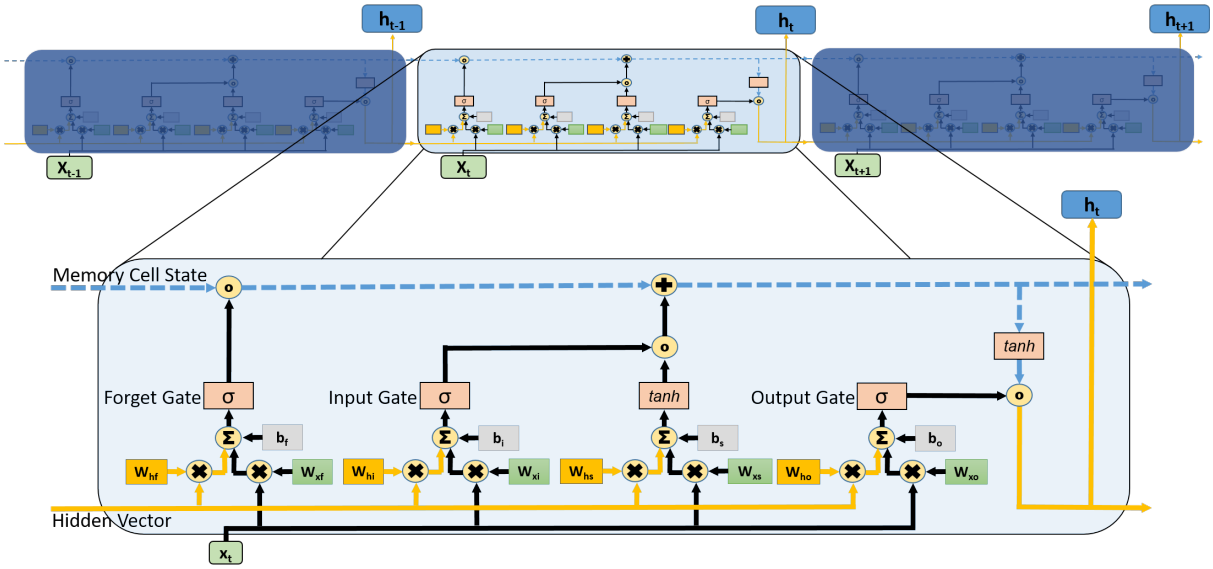


Figure 4. This illustrates the internal structure of a single LSTM memory cell unfolded in time where x_t is the windowed sequence input from time t, and h_t is the hidden-vector state at time t. Weight matrices are labeled using a from-to subscript convention, $W_{from,to}$. Biases, b, have subscripts associated with their corresponding gate or activation. The following operations are annotated symbolically: X represents matrix-vector multiplication, $\circ$ is a Hadamard (element-wise) product between two vectors, $\sum$ is used for summations of 3 or more vectors, $+$ indicates the addition of two vectors, finally σ and $tanh$ are respectively a sigmoid function and hyperbolic tangent function applied element-wise to a vector. The memory cell state vector propagates along the dashed blue line at the top while the input vector and hidden vector buses are connected as shown by the gold and black lines respectively. All three of these vectors should be thought of as column vectors. The size of the hidden vector and memory cell state vector are equal to the number of "memory cells" specified during initialization of the LSTM layer. Three sigmoid functions act as gates and are labeled: forget, input, and output. If a LSTM returns a sequence, the sequence is a series of the hidden vector states with each corresponding to the output from a particular timestep within the supplied temporal window. If a sequence is not returned, typically the last hidden state is returned.

element of the gate vector to a value between 0 and 1. The gating functionality is accomplished by taking this vector of values between 0 and 1 and performing an element-wise multiplication with another vector, thus specifying what proportion of the second vector passes through the gate; and conversely, determining which parts are blocked. Figure 4 illustrates the internal structure of the LSTM. For clarity, the state of a memory cell at time t with recurrent connections from time $t-1$, and to time $t+1$ is described.

As shown in Figure 4, there are two vectors that persist from time $t-1$: the hidden vector, h_{t-1}, and the memory cell state, s_{t-1}. The forget gate f_t, as shown in Equation 1, determines what to remove from the memory cell state. In other words, it forces the memory cell to forget things that are not important based on error backpropagation:

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f), \tag{1}$$

where the weight matrix W_{xf} is the weight matrix from the input x to the forget gate f_t, x_t is the input at time t, W_{hf} is the weight matrix from the previous hidden vector h_{t-1} to the forget gate f_t, and b_f is the forget gate bias. The from-to subscript convention is used to describe each weight matrix. The input gate i_t determines how much each element of the candidate update vector, $\tilde{s_t}$, should be added to the corresponding memory cell element at time t based on the recurrent connection from the hidden vector h_{t-1} and the sequential input at time t, x_t. The gate scales the candidate update vector which is the output from a fully-connected *tanh* layer:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \tag{2}$$

$$\tilde{s_t} = tanh(W_{xs}x_t + W_{hs}h_{t-1} + b_s). \tag{3}$$

The delicate balance required to maintain memory cell state over long sequences is attained by forgetting old information and incorporating new information. Forgetting is accomplished by multiplying the old memory cell state by the output of the forget gate, while the new information is supplied by adding the portion of each value specified by the element-wise product of the input gate with the candidate update:

$$s_t = i_t * \tilde{s_t} + f_t * s_{t-1}. \tag{4}$$

Finally, the output gate determines what should be output to the hidden vector from the memory cell state, given the temporal context of the time-step, to minimize error. The output gate, o_t, and hidden vector, h_t, are given by:

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \tag{5}$$

$$h_t = o_t * tanh(s_t). \tag{6}$$

4.2.4 RNN Models for EEG Analysis.

Despite excellent results in other fields, relatively few researchers have applied deep neural network techniques to classify EEG data. Bashivan et al. [14] trained a deep recurrent-convolutional neural network accounting for both temporal and spatial dependencies in the network. They began by performing a Fast Fourier Transform (FFT) on each time-series signal from each electrode and estimating the power in the theta, alpha, and beta frequency bands over 3.5 second working memory experiment trials with four classes of task difficulty. Then, they created a time-series of images by performing a 2-d Azimuthal Equidistant Projection (AEP) for power in each of the three different bands. This preserved distances from each electrode to the center point of the EEG cap, thus accounting for some spatial dependencies. These images were fed into a series of convolutional and max pooling layers. Their best performing model fed the output from the convolutional portion of the architecture into a 1-d convolutional layer, as well as a LSTM layer, and merged the output from both of these into a fully connected layer. This was then connected to a softmax layer which provided the final classification result. This complex architecture used 1.62 million parameters and was able to reduce the classification error from 15.34% to 8.89%; an impressive 42% reduction compared to a baseline radial basis function SVM [14]. While a large reduction in error over baseline methods was achieved, the complexity

and amount of computation required to train such a deep network is significant. The research in this chapter sought to examine if similar reductions in error over baseline methods could be achieved using different preprocessing techniques and a less complex deep architecture which only used recurrent neural networks.

Two other researchers used a form of the LSTM to analyze EEG data. Davidson, et al. found that a small single-layer LSTM was able to identify lapses in attention based on EEG spectral data better than a tapped delay-line Multilayer Perceptron (MLP) during performance of a visuomotor tracking task [49]. They trained their networks in a leave-one-out, cross-subject manner. Their results evaluated temporal dependencies out to a length of six seconds and showed that accounting for temporal dependence of up to four seconds prior to a lapse improved detection. An extended temporal window of 30 seconds is used in this chapter's methodology. In other work, Binz et al. [22] used a derivative of the LSTM unit, called Dynamic Cortex Memory (DCM), to classify imagined sensorimotor imagery data from a Brain Computer Interface (BCI) workshop competition [23]. A single hidden layer was used with eight DCM units followed by a softmax output layer. While their results did not achieve state-of-the art accuracy, several advantages were present in their solution. Their network was able to provide real-time results and their solution did not need to specify a time window, since the network learned appropriate temporally-dependent sequences [22]. Binz's work largely differed from methods used in this chapter since a mixture of within-participant and cross-participant models were used to evaluate trials that were separated from the training data by only seconds-to-minutes rather than days. The only similarity between the two studies was that both used forms of the LSTM architecture to account for temporal dependencies in EEG signals.

In a medical application, Guler et al. [69], Srinivasan et al. [149], Ubeyli [166], and Kumar et al. [100] successively improved performance of epilepsy diagnosis us-

ing RNNs. All four research groups used various input features to train small Elman RNNs in a cross-subject manner using the dataset described by Andrzejak et al. [9]. The Elman RNN architecture has limited temporal representational capacity compared to the networks trained in this chapter's investigation. Like in this chapter, each research group demonstrated the superiority of an RNN compared to other neural networks that did not incorporate temporal dynamics. However, unlike experiments in this chapter, the data sequences analyzed did not span temporal lengths great enough to examine day-to-day variability. Finally, Guler and Ubeyli both demonstrated that summary statistics (min, max, mean, standard deviation) of time-varying feature distributions can be useful input features for a recurrent model; however, no comparison to a baseline feature set was provided in either case.

The primary difference between previous research and that in this chapter is that results in this chapter demonstrate improved within-participant, day-to-day feature stationarity by accounting for temporal dependencies in cognitive activity; whereas the aforementioned studies do not address day-to-day variability. Another difference is that the preceding research focused on either medical uses or state estimation unrelated to workload. A final differentiating factor was that many previous studies were conducted prior to breakthroughs that enabled drastic improvements in network performance such as advances in initialization, optimization, and regularization techniques, as well as the rise of abundant computational capacity via Graphics Processing Unit (GPU) computing which enables the training of larger, deeper networks.

4.3 Methodology

The goal of producing several workload models to evaluate the efficacy of new features and LSTM based classification algorithms required preprocessing of the EEG data to convert it into the frequency domain and to extract power in different fre-

Table 9. Test matrix of all combinations of mean, variance, skewness, and kurtosis features. * denotes feature sets that were included in models that incorporated the mean while all others are models that did not incorporate the mean.

Test Run	Features Included In Dataset
1*	Mean
2	Variance
3	Skewness
4	Kurtosis
5*	Mean, Variance
6*	Mean, Skewness
7*	Mean, Kurtosis
8	Variance, Skewness
9	Variance, Kurtosis
10	Skewness, Kurtosis
11*	Mean, Variance, Skewness
12*	Mean, Variance, Kurtosis
13*	Mean, Skewness, Kurtosis
14	Variance, Skewness, Kurtosis
15*	Mean, Variance, Skewness, Kurtosis

quency bands as outlined by Hefron and Borghetti [77]. Raw EEG data was transformed into features in clinical frequency bands (delta (1-4Hz), theta (4-8Hz), alpha (8-14Hz), beta (15-30), and gamma (30-55Hz)) to conduct time-frequency analysis using the following process: The power spectral density was determined for 30 points spread out over a logspace from 3Hz to 55Hz by extracting power from complex Morlet wavelets [42]. Each wavelet was 2 seconds in length and the number of wavelet cycles increased logarithmically from 3 to 10 in conjunction with the frequencies. Mean power in each band was determined by averaging each power value for the evaluated frequencies within each of the clinical bands. Power was then aggregated over a ten second sliding window with 9 seconds of overlap, allowing for a new update each second.

Final features were generated by determining the mean, variance, skewness, and kurtosis of the power distribution in each of these ten second windows for all 19 EEG electrode sites across the five frequency bands. This process yielded 380 features for each second and approximately 9,000 observations per individual for the five day period. These features were then centered and scaled by session so that each session

had a mean of zero and variance of one since none of the algorithms used for analysis were scale invariant. The data were split such that the first four days were used for training and cross-validation while the last day was reserved for testing.

To examine the effect of inclusion/exclusion of particular types of features, the test matrix shown in Table 9 was constructed to document features included in each test run. A total of six algorithms were used to train models: linear SVM (SVM-L), Radial Basis Function (RBF) SVM (SVM-R), feedforward ANN (ANN), deeply stacked simple RNN (RNN-D), single LSTM (LSTM-S), and deeply stacked LSTM (LSTM-D). All model development used 4-fold cross-validation to select hyperparameters for each model and then final models were trained using all data from days 1-4. This process was repeated for each feature set in Table 9 resulting in 90 final models for each algorithm. Final models were trained using each set of features and differences in classification accuracy due to choice of algorithm and feature set were considered. It is important to note that the cross-validation data was split so that each fold was a full day rather than splitting the folds by random selection of observations from the entire dataset. There were two compelling reasons for this choice. The first reason was that randomly selected cross-validation points allow for too many temporally adjacent points to be split between the training and test sets which artificially inflates cross-validation accuracy due to the non-stationarity of the datasets. The second reason was to preserve temporal context of the data for processing when using RNNs.

Once final models were produced, classification accuracies for the holdout day 5 test set were determined for each individual and the algorithm average was calculated across participants. This enabled by-algorithm and by-feature set comparisons. Due to confounding effects of algorithm selection and feature sets on classification accuracy, an ANOVA test with five factor outcomes was performed to elicit the effect of

varying levels of each factor. The first factor was algorithm selection which had six levels corresponding to each of the aforementioned algorithms. The remaining four factors grouped the feature sets in a binary fashion based on whether a feature was included or excluded from a particular test run. For example in Table 9, all asterisked runs produced models that included the mean, while the others were models that excluded mean features. Interaction effects were not examined. A significance level of $\alpha = .05$ was used to indicate if a factor had a statistically significant impact on classification accuracy. The Tukey Honest Significant Difference (HSD) test was performed following the ANOVA to determine which classification accuracies were different from each other in the six-level algorithm factor, and to determine the direction and magnitude of differences across all two-level factor comparisons.

All neural networks were created using the Keras [37] and Theano [163] frameworks. The feedforward ANN had a single hidden layer that was fully connected with the input layer and the output layer. The input consisted of the appropriate number of features for a given test run, while the output layer was a single node with a sigmoid activation function which forced a classification as either high or low workload. Mini-batch gradient descent was performed using 600 observations per batch for all neural network implementations, and the *Adam* optimizer was chosen to optimize mini-batch gradient descent due to its ability to handle non-stationary targets and noisy data [96]. A binary cross-entropy loss function was selected as the cost function [17]. The number of nodes in the hidden layer was tuned by performing 4-fold cross-validation while varying the number of nodes from 50-800 in steps of 50. This resulted in 3,960 cross-validation models being trained. The lowest cross-validation error rate was used to determine both the number of nodes to use in the hidden layer and how many epochs to train the network.

As illustrated in Figure 5, the deep LSTM architecture consisted of an input

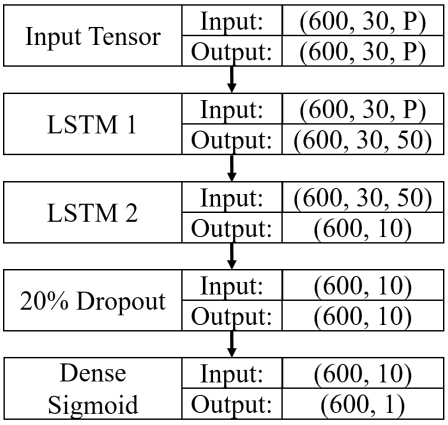


Figure 5. Deep LSTM Architecture: This illustrates the size of the tensor in terms of batch size, temporal depth in seconds, and number of features used at each level of the network. P represents the number of features used based on the test matrix shown in Table 9, and ranged from 90 to 380 features. Since LSTM 2 does not return a sequence, the tensor becomes two-dimensional at that point.

layer, the first sequence-to-sequence LSTM layer, a many-to-one LSTM layer, a 20% dropout layer, and a final sigmoid activation function for binary classification. The first hidden layer contained 50 LSTM units while the second hidden layer used 10 units. With unlimited resources, the number of hidden layer nodes could have been tuned exhaustively using grid cross-validation. However, due to computational resource constraints, several hidden layer sizes were tested for each layer on smaller representative sets of data and it was found that reducing the number of LSTM units in each layer improved generalization and that empirically, 50 and 10 appeared to work well. Each network had a lookback of 30 seconds of pre-processed features or, for the case of the second layer, features generated from the output of the first LSTM layer. Dropout on the input gates to each LSTM layer and between the final LSTM and fully-connected sigmoid layer served as a method of regularization and was set to 20% [17, 150]. Dropout prevents co-adaptation of the hidden units by temporarily removing a percentage of randomly selected nodes, including their input and output, in a given layer during a training pass [80]. This forces hidden units to learn features without depending upon particular nodes to correct mistakes made during learning

[80]. While dropout is the most widely used regularization method for deep neural architectures, it is also important to understand when it is not appropriate to use. The recurrent connections within the LSTM structure are one such case. The purpose of the recurrent connection in a LSTM is to store important long-term dependencies. Pham et al. [127] showed that if dropout is applied to the recurrent connection, then the long term memory becomes corrupted and inhibits learning rather than improving generalization. Similar to Zaremba et al. [183], it was found that using a dropout of 20% on the input gates and between the final LSTM output and the sigmoid classification layer provided better results than the typically recommended 50% dropout for fully connected layers. The number of training epochs was tuned using cross-validation as previously specified in this section. The length of training with the lowest cross-validation error rate for each participant/feature set combination was selected for final network training. All training data was then used to retrain the network. To ensure that anomalous behavior was not present in the reported results from day 5, all other combinations of cross-validation with hold-out test day were performed for the deeply stacked LSTM model. No significant deviations from the reported upon results in Section 4.4 were observed due to permuting the test and training days.

Two recurrent networks were trained which were derivatives of the deep LSTM architecture. The differences between the architectures are detailed next. The deeply stacked simple RNN used the same architecture as the deep LSTM except that instead of using LSTM units, simple recurrent units were used. Due to parameter sharing in the weight matrix for the recurrent connections and a lack of gating functions, the vanishing gradient problem was present and restricted the effective temporal period to 10-20 seconds for this model despite 30 seconds of data being provided as input [17]. The second network consisted of an input layer, a many-to-one LSTM layer using 50

hidden units, and a sigmoid output layer. Training and classification methods using these networks mirrored those of the deep LSTM architecture.

Two categories of SVMs were trained, one with a linear kernel as recommended by Christensen et al. [39], and one using a RBF kernel. The resulting models were used to compare neural network results to those from a more traditional machine learning algorithm. To train the linear SVM, the appropriate number of features for a given test run were provided for each observation supplied to the SVM. The tuning parameter C set a tolerance for number and severity of margin and hyperplane violations–effectively determining smoothness of the decision surface [88, 126]. This hyperparameter was optimized using a cross-validated grid search across exponentially spaced values of C from 10^{-4} to 10^{2} resulting in 2,520 cross-validation models for the linear SVM. Nearly all C values selected for the final models were either 10^{-4} or 10^{-3}. All training data was then used to retrain the linear SVM with the selected values for C. The procedure for training the RBF SVM was nearly the same as in the linear case, except a cross-validated grid search over values of γ and C was performed where the hyperparameter γ controlled the radius of influence for a single observation. The same range of C values were evaluated while γ was exponentially varied from 10^{-3} to 10^{2}, resulting in 15,120 cross-validation models for the RBF SVM. The best hyperparameters were selected and the final models were trained. This procedure ensured proper tuning to establish a valid baseline for comparison with new techniques.

4.4 Results

Results of the five-factor ANOVA indicate that algorithm type, mean features, and variance features had a statistically significant effect on classification accuracy (all p-values $< .0001$). Skewness and kurtosis in the presence of mean and variance

Table 10. ANOVA table with five factors. Algorithm is a six-level factor indicating the algorithm used: LSTM-D, LSTM-S, RNN-D, ANN, SVM-L, or SVM-R. The final four factors have two levels and indicate the feature was either included or excluded.

Source	DF	Sum of Squares	R^2	F Ratio	Prob >F
Algorithm	5	0.7876	0.2774	29.4052	<.0001
Categorized by Mean	1	1.2113	0.4266	226.1277	<.0001
Categorized by Variance	1	0.2579	0.0908	48.1448	<.0001
Categorized by Kurtosis	1	0.0033	0.0011	0.6235	0.4301
Categorized by Skewness	1	0.0028	0.0009	0.5252	0.4689

were not significant (p-values > 0.43). Table 10 summarizes the results from the ANOVA while Tables 11 and 12 display the post hoc Tukey HSD results. Results from the Tukey HSD test showed that the ANN and both SVM models' mean classification accuracies were not statistically different with p-values ranging from 0.8906 to 1. The deep LSTM models demonstrated statistically significant accuracy increases of 8.9% over ANN results as well as 8.8% and 7.8% over linear and RBF SVM results respectively (all p-values $< .0001$). The participant averaged classification accuracies for each model and feature set in Table 13 show that there are 6 feature combinations that result in classification accuracies greater than 90% using the deep LSTM architecture. The highest accuracy feature set was attained using all features with the deep LSTM model. This model achieved an average classification accuracy of 93.0% across participants. This compares favorably to 84.8% and 84.6% for the best ANN and SVM cross-participant feature set performance. Furthermore, this represents a 58% decrease in error compared to the best baseline case–the mean-only RBF SVM. These results illustrate the importance of accounting for temporal dependencies in workload data. Further examining the results shows that the influence of using a temporally-stateful model on classification accuracy for workload estimation cannot be understated. In all cases tested, every temporally-stateful model outperformed the best performing non-stateful model and were found to be significantly different than all non-stateful models with Tukey HSD p-values all $< .0001$ (Table 11).

Table 11. Tukey HSD results for all pairs of algorithm levels.

Model 1	Model 2	Diff	Std Err	t Ratio	Prob>|t|	95% Conf Int
LSTM-D	ANN	0.089	0.011	8.19	<.0001	.058 to .121
LSTM-D	SVM-L	0.088	0.011	8.06	<.0001	.057 to .119
LSTM-D	SVM-R	0.078	0.011	7.12	<.0001	.046 to .109
LSTM-D	RNN-D	0.020	0.011	1.82	0.4536	-.011 to .051
LSTM-D	LSTM-S	0.010	0.011	0.89	0.9493	-.022 to .041
LSTM-S	ANN	0.080	0.011	7.31	<.0001	.049 to .111
LSTM-S	SVM-L	0.078	0.011	7.17	<.0001	.047 to .109
LSTM-S	SVM-R	0.068	0.011	6.23	<.0001	.037 to .099
LSTM-S	RNN-D	0.010	0.011	0.93	0.9382	-.021 to .041
RNN-D	ANN	0.070	0.011	6.37	<.0001	.038 to .101
RNN-D	SVM-L	0.068	0.011	6.24	<.0001	.037 to .099
RNN-D	SVM-R	0.058	0.011	5.30	<.0001	.027 to .089
SVM-R	ANN	0.012	0.011	1.08	0.8906	-.019 to .043
SVM-R	SVM-L	0.010	0.011	0.94	0.9351	-.021 to .041
SVM-L	ANN	0.001	0.011	0.13	1	-.03 to .033

Table 12. Tukey HSD results comparing models where the feature of interest was included versus those where it was excluded.

Model 1	Model 2	Diff	Std Err	t Ratio	Prob>|t|	95% Conf Int
Mean	-Mean	0.096	0.006	15.04	<.0001	.083 to .108
Var	-Var	0.044	0.006	6.94	<.0001	.032 to .057
Skew	-Skew	0.005	0.006	0.72	0.4689	-.008 to .017
Kurt	-Kurt	0.005	0.006	0.79	0.4301	-.007 to .018

Table 13. Cross-participant averaged classification accuracy for each model and feature set. Mean, variance, skewness, and kurtosis features are denoted by M, V, S, and K respectively. Bold values are models with greater than 90% classification accruacy.

Feature Set	SVM-L	SVM-R	ANN	LSTM-S	RNN-D	LSTM-D
M	0.823	0.834	0.816	0.871	0.884	0.891
V	0.762	0.769	0.754	0.865	0.850	0.861
S	0.680	0.694	0.678	0.757	0.760	0.765
K	0.672	0.686	0.662	0.732	0.736	0.762
M/V	0.836	0.846	0.844	**0.911**	0.866	**0.911**
M/S	0.823	0.836	0.828	0.878	0.861	0.897
M/K	0.831	0.843	0.830	0.896	0.884	**0.908**
V/S	0.762	0.771	0.748	0.847	0.869	0.862
V/K	0.758	0.769	0.757	0.863	0.833	0.882
S/K	0.716	0.733	0.709	0.834	0.770	0.807
M/V/S	0.839	0.840	0.848	**0.909**	0.890	**0.927**
M/V/K	0.835	0.837	0.838	**0.907**	**0.918**	**0.930**
M/S/K	0.824	0.838	0.831	0.897	**0.911**	**0.918**
V/S/K	0.759	0.771	0.753	0.846	0.847	0.863
M/V/S/K	0.835	0.842	0.836	**0.913**	0.899	**0.930**

Statistically, inclusion of mean and variance were significant, while skewness and kurtosis were not. ANOVA results show a significant effect dependent upon inclusion or exclusion of mean features ($p < .0001$). Post hoc comparisons using the Tukey HSD test indicate that average classification accuracy for models including the mean features results in an accuracy increase of 9.6% versus models excluding the mean features (Table 12). A significant effect is also present dependent upon inclusion or exclusion of variance features ($p < .0001$). The Tukey HSD test shows that average classification accuracy for including the variance features increased 4.4% versus excluding the variance features. ANOVA and Tukey HSD results for skewness ($p = 0.4689$) and kurtosis ($p = 0.4301$) were not significant. However, by comparing models with and without a single feature, these results merely indicate that kurtosis does not add significantly to a model with skewness included and vice-versa. It is hypothesized that skewness and kurtosis may be more important in situations involving workload transitions. Workload transitions were not investigated in this chapter, but are in Chapter VI. It is expected that transitions between different workload levels may first become apparent in the tails of the distributions.

4.5 Conclusion and Future Work

Cross-day workload estimation based on EEG is a difficult domain due to temporal non-stationarity of feature-to-target mappings. Previous research on cross-day workload estimation implicitly assumed independence of a participant's workload from one instance to the next due to the algorithms used for analysis. Theory and practical experience show that workload can build in a cumulative fashion and that a temporal dependence exists. RNN models, in particular those that use LSTM architectures, can account for both long-term and short-term temporal dependencies inherent in brain activity data. This work also statistically evaluated the utility of mean, vari-

ance, skewness, and kurtosis of frequency-domain power distributions and found only the mean and variance to be statistically significant.

The research in this chapter demonstrated the utility of deep RNN models and particular feature sets for cross-day workload estimation and showed that drastically improved model accuracy can be achieved over SVM and feedforward ANN models when working with non-independent data. Previously, the best accuracy achieved using this dataset was 83%. Models built during this study with deep LSTMs increased that accuracy to 93.0%, representing a 58% reduction in classification error over baseline methods and a 59% decrease in error compared to the best published results for this dataset. This pushed us closer to the 95% threshold where adaptive operator augmentation may become feasible.

There is an abundance of future work to be pursued in this area. Due to time constraints and computational complexity, only a select number of deep architectures were examined during this research. A thorough evaluation of different deep RNN architectures to include variations in the depth of hidden layer recurrent connections, stacking of different sized LSTM layers, and interleaving fully-connected feedforward layers between sequence-to-sequence recurrent layers may yield additional improvement.

Another enhancement for future work would be to include workload transitions. It is believed that skewness and kurtosis may be relevant in datasets where the target workload is transitioning across high and low workload conditions since distributional changes may first become evident in the tails of the distributions. Other ideas to pursue include creating ensembles of deep RNNs. This would almost certainly improve results as long as enough diversity could be added to the ensemble. In this chapter's research, only 30 second sequences were fed to the recurrent networks. Exploring variations in temporal length supplied to a RNN to examine stationarity of target-

to-feature mapping for workload estimation using EEG would also be an interesting subject for future work. Deep RNN architectures could also be used to improve cross-participant and cross-day classification simultaneously by training and testing on all participants grouped together rather than individually. Finally, significant improvement could be realized if time-series data augmentation methods are developed capable of forcing a learned invariance to the sources of temporal non-stationarity.

V. Enhancing cross-participant EEG modeling with multi-path convolutional recurrent neural networks

5.1 Introduction

One critical area for research aimed at improving overall performance in human machine teams has been the development of models which better predict human cognitive workload: when a machine knows the human's workload it can make better decisions. Many of these efforts use neurophysiological signals, such as electroencephalographic (EEG) data, to infer the cognitive workload that the human is experiencing while performing a task. As participants complete tasks their electroencephalograph (EEG) signals are recorded and their cognitive workload is assessed by subjective ratings. Then the neurological signals and the workload measurements are used to fit a machine learning model which can infer the participant's workload from the signals alone.

A historical standard for human model performance is the tailored single-participant model. Single-participant models are fit using only data from the participant being modeled, not data from other people. Since single participant models are specifically trained to perform well on the individual, these models will often have the highest performance with respect to that particular individual. However, training a separate model on each individual is resource intensive for both collection and processing. If the data for models could be collected from many people instead of just the one being modeled, the collection burden could be spread over many individuals.

Models trained on data from multiple people are known as group models or cross-participant models. In cross-participant model-fitting, data from one set of people is used for training and the models are later used to make workload predictions on those people, or possibly other people. The benefit of these cross-participant models

is that they can be prepackaged and used in many settings with many individuals, requiring little or no calibration for each individual.

A model should perform well on an arbitrary individual independently of which set of other people it was trained on. But individual differences apply here: variation in EEG response to workload between people makes it challenging to design, train, and validate models using data from some people to make good workload assessments on others. As in most other machine learning settings, a desirable model is one in which the predictions are accurate and the variance over predictions is low. Another desirable characteristic is being able to make accurate, low variance predictions on short EEG data sequences, as this should result in minimum lag in accurate assessment. However, machine learning models almost always make better predictions on longer streams of data, so there is a trade-off between temporal specificity and model performance.

In this chapter, the trade-space of model accuracy, variance, temporal specificity, and computational efficiency is explored in three thrusts: 1) develop new tailored architectures designed specifically for cross-participant classification of EEG signals, 2) evaluate efficiency and performance of various training methods for these architectures, 3) characterize the effect of varying the temporal length of EEG features available to a model.

New methods are presented for cross-participant estimation of operator workload in a non-stimulus-locked, multi-task environment by developing and evaluating 7 neural network architectures. The case where no data from the test participant is used in any way to improve feature distribution similarity with the training or validation participants is examined. Findings in this chapter show that a novel Multi-Path Convolutional Recurrent Neural Network (MPCRNN), designed to learn cross-participant frequency and temporal representations, simultaneously resulted in

a statistically significant improvement in accuracy and decrease in variance compared to all six other highly-tuned network architectures using a Multi-Attribute Task Battery (MATB) dataset with eight participants. This contribution moves away from reliance on the clinical frequency bands and towards learning of appropriate frequency representations–a significant departure from previous deep learning work in the field.

It is shown that increasing sequence length–a common method to improve accuracy in non-stimulus-locked settings–increases both mean accuracy and variance at statistically significant levels for cross-participant models. Since variance grows with sequence length, increasing temporal sequence length does not adequately address the cross-participant modeling challenge: Producing high-accuracy models with low variance across participants.

The remaining research thrust examines differences in performance, computational efficiency, and use cases of 4 training methods including single models incorporating all training participants and ensembles of individual-participant models. No significant differences were found between ensemble and group-based methods, so selection should be driven primarily by experimental design and computational cost constraints which generally favor the use of an ensemble. In addition to the primary thrusts, this chapter fills gaps in current research by performing a direct comparison between different convolutional and recurrent network architectures in a multi-task, non-stimulus-locked environment. Multi-path convolutions and residual connections in bi-directional recurrent networks are shown to both have a positive effect on network performance over baseline models.

5.1.1 Related Work.

Cross-participant modeling techniques use data from multiple individuals to fit statistical machine learning models that later are used to make predictions on people.

These models can be divided into two categories: shared-data and zero-data methods. Shared-data methods fit models using data from all people *including* those on which the model will make predictions. Zero-data models use no data from the evaluated individual; these models only use data from other participants to fit the model.

In the first subsection, shared-data studies are reviewed. This subsection also examines cross-participant feature saliency. In shared-data cross-participant studies, finding data features which have good predictive value and are generalizable across all participants (salient features) is one of the goals. In the second subsection, zero-data studies are reviewed - these studies maintain a strict data boundary between the individual to assess and the set of individuals used to fit the model.

Application domains reviewed in this section include both cognitive modeling and medical prediction, which have similar desirable characteristics. Cognitive modeling applications cover detecting cognitive load, fatigue, attentional lapses and neural oscillations from movements or imagined movements. Medical field applications include epileptic seizure detection and classification. This section differentiates the use of time-locked stimulus models from non-time-locked models in the research. In a laboratory experiment, isolated, well defined, causal stimuli can be generated, allowing these stimuli to be used in time-locked models. In real-world multi-task environments, cognitive activity is not always associated with individual, causal, well-defined event stimuli. Because real-world tasks lack these well-defined stimuli, time-locked models may not be applicable.

5.1.1.1 Shared-data cross-participant modeling and feature saliency.

Several researchers developed group-trained models where a portion of data was used from each participant for training while the remaining data from those same participants were used for testing. Since these shared-data models were trained using

some of the data for the individual being assessed, these models yield the highest possible expected classification accuracy for cross-participant models. Wang et al. [170] achieved 80% accuracy while showing a hierarchical Bayes model outperformed baseline neural network models in a three class, cross-participant MATB workload classification setting. Cross-participant variation was accounted for by learning parameters of a hidden Gaussian representation using segments from all 8 participants' data and testing on other data segments from those same 8 individuals [170]. Zhang et al. [184] used adaptive exponential smoothing to improve feature stationarity and adaptive bounded Support Vector Machines (SVMs) to improve cross-participant generalization by iteratively adding misclassified examples from the test set to the training set to adapt the model performance to a new participant. Yin et al. [181] trained models using the transfer recursive feature elimination technique with linear SVMs and found that by adapting features from other individuals based on a small validation set from the test individual, statistically significant improvements in classification performance resulted when evaluating the remaining test data.

Wilson and Russell [178] conducted a within-participant MATB study and determined the relative contribution of different features from each individual varied widely among participants. This highlights the challenge of cross-participant distributional differences in the MATB environment. A similar result was reported by Noel et al. [123] for an in-flight workload experiment where a drastic difference in the number of salient features between pilots was noted. Additionally, for each pilot, most salient features differed across days causing salient features from multi-day experiments to diverge from single day experiments [123]. This indicates a coupling of temporal non-stationarity and cross-participant differences can compound the problem. In a separate MATB experiment, Laine et al. [101] also found feature saliency at the individual level to be highly variable, but were able to identify a stand-alone

set of features that worked for training Artificial Neural Networks (ANNs) on all individuals by using Stepwise Discriminant Analysis (SWDA) to select common features across the group. This yielded a binary classification accuracy of 83% which did not significantly differ from the within-participant modeling result. Their finding is important because it suggests when using all participants for feature selection, a set exists that does not reduce classification accuracy from the accuracy level achievable in individually-tailored models.

5.1.1.2 Zero-data cross-participant modeling.

Since the objective of many EEG application domains is to be able to deploy the technology with little to no user-specific data available for model tuning, numerous researchers have explored zero-data cross-participant modeling. For zero-data cross-participant models, this section examines how training methods, algorithmic assumptions, and features affect assessment accuracy and variance across participants.

While shared-data methods have shown that cross-participant model performance can approach within-participant accuracy in some cases, zero-data models often perform worse than within-participant models, due to individual differences. Gevins et al. [61] trained cross-participant single-hidden-layer ANN models using all individuals except the hold-out test participant for binary classification of stimulus-aligned spatial and verbal working memory tasks. The mean classification accuracy for the group classifier was 83% which represented a significant reduction from the 94% accuracy reported for individually-trained models [61].

In zero-data cross-participant modeling, different algorithms can significantly affect performance. Using Improved Performance Research Integration Tool (IMPRINT) workload profiles [6] as regression targets for a simulated remotely piloted aircraft tracking task, Smith et al. [147] showed algorithm type could have a statistically sig-

nificant effect on zero-data cross-participant operator workload estimation for non-stimulus aligned tasks, a result confirmed by this chapter. Additionally, random forests improved group-trained model performance when compared to non-ensemble methods in the same algorithmic family, suggesting that in complex workload environments, ensemble models may yield better performance than their non-ensemble counterparts.

Now the impact of deep learning on zero-data cross-participant modeling is discussed. Certain deep neural networks outperform other methods for zero-data cross-participant modeling because they better model two of the conditions present in human state assessment: 1) the temporal ordering of signals which result from brain activity, and how those time-series signals map to temporally ordered sequences of human state assessment, and 2) the spatial relationship between EEG collection sites on the scalp. Temporal context can be accounted for using Recurrent Neural Networks (RNNs) and/or Convolutional Neural Networks (CNNs) while the spatial contribution can be modeled by CNNs.

Accounting for temporal context using Elman RNNs [52] improved diagnosis of epilepsy using cross-participant modeling of EEG data [69, 166]. Utilizing cross-validated, group-trained models, Guler et al. [69] and Ubeyli [166] reported reductions in diagnostic error of 63% and 74%, respectively, compared to non-recurrent methods. Recurrent networks were also effective in a high-fidelity vehicle simulator study using EEG to sense occipital lobe activity prior to and during lane perturbation events when performing a simulated highway driving task [112]. While the reported results indicated slightly better performance for ensembles of group-trained Recurrent Self-Evolving Fuzzy Neural Networks (RSEFNNs) compared to a battery of other neural network ensembles in predicting a normalized drowsiness metric [112], a rigorous statistical treatment in their study could have confirmed this. Despite the lack of

statistical results, Liu et al. [112] demonstrated that ensembles of recurrent networks can produce excellent results in a stimulus-aligned cross-participant task environment.

Several researchers have accounted for both temporal and spatial relationships in EEG data by using CNNs or combinations of CNNs and RNNs. Lawhern et al. [102] developed a small CNN architecture that was able to generalize well across several EEG Brain Computer Interface (BCI) analysis domains including visual stimulation of P300 Event-Related Potentials (ERPs), neural oscillations associated with movement-related cortical potentials, and sensorimotor rhythms evoked by real or imagined movements [102]. Convolutions across the electrode channel dimension as well as temporal dimensions were used. The first layer of their model used 16 1-d kernels (each the same length as the number of electrode channels), which were convolved without zero-padding with each of the input tensors. This layer was the most interesting development of Lawhern's model in that each of the kernels had the ability to learn useful channel interactions and had the effect of abstracting away the need to explicitly model locational dependencies inherent in an EEG system. Whenever possible within the constraints of a given dataset, Lawhern et al. [102] trained models using a cross-validated, cross-participant group method so that it was user-agnostic [102]. Importantly, Lawhern et al. [102] found that cross-participant variability of classification accuracy correlated with the Signal to Noise Ratio (SNR) of the signal associated with the phenomenon of interest. This means that for operator workload experiments in a non-stimulus-aligned environment such as the MATB, high cross-participant variability could be expected.

Hajinoroozi et al. [71] constructed two unique CNNs which were designed to perform convolution across 1-second temporal periods of raw EEG data from each channel resulting in improved cross-subject and within-subject classification for a driving simulator lane perturbation task compared to a large array of baseline algo-

rithms. The CNNs convolved across the time domain in a manner which effectively searched for ERPs present in each individual channel. The first CNN used 10 kernels while the second used only 1 kernel, but was pre-trained as a Restricted Boltzmann Machine (RBM) followed by fine-tuning. The first CNN significantly outperformed all other models for within-participant prediction with an Area Under Curve (AUC) of 0.8608, while the RBM CNN performed far better than any other model in the cross-participant classification environment, achieving an AUC of 0.7672 [71]. These results suggest that either the reduced model capacity of the RBM CNN led to better cross-participant generalization, or that the process of performing unsupervised pre-training helped learn shared features across individuals. Overall, the use of an architecture which uses raw EEG to find per-channel ERP signatures was novel and warrants further investigation as a merged component in a large deep neural architecture which also incorporates time-frequency domain features.

Bashivan et al. [14] trained a deep convolutional-recurrent neural network to predict cognitive load during a working memory task [12]. A time-series of 3-channel images were created by performing 2-d Azimuthal Equidistant Projections (AEPs) of Power Spectral Density (PSD) features from the theta, alpha, and beta frequency bands. Models were trained using early stopping based on a randomly-selected validation sample selected from within the training set of a 13-fold, leave-one-participant-out train/test setup. Results showed a 30% reduction in error compared to random forest models and indicated strong frequency-band selectivity meaning the filters applied to specific channels of input feature space [14]. However, since mean spectral powers in EEG clinical bands were used, and the definition of these bands were organically developed over a century of experiments, it is unlikely that features developed only from combinations of these bands will be optimal for all human state assessment activities. Models which can learn the most applicable frequency responses at a finer

granularity may perform better and should be considered in future research.

In the recurrent models discussed so far, the temporal direction is always forward such that early signals influence the model's understanding of later signals. This architecture ensures causality of brain activity is not violated but does not allow for reflection: a model cannot learn how to interpret the early signals using signals which are experienced later. An example of a type of signal in which reflection is important is speech. In the speech recognition task, an audio signal is converted into a string of characters or words. It is common to estimate the probability distribution of possible next words as conditioned on the signal and the previous words (or audio signal associated with those words). However, it is likely necessary that the conditional dependencies in speech be considered in both the forward and reverse directions to maximize transcription accuracy. Graves and Schmidhuber [67] showed that by using a model capable of understanding both forward and reverse dependencies in speech, performance was improved. The team used bi-directional Long Short-Term Memorys (LSTMs) that could effectively exploit contextual dependencies in both directions to improve speech processing.

Recently, research using bi-directional LSTMs for brain signal analysis has begun [164]. Thodoroff et al. [164], implemented a bi-directional LSTM following a 2d convolutional architecture and prior to a fully-connected layer for cross-participant epileptic seizure classification. Their architecture performed spatial convolutions similar to Bashivan et al. [14]. This combined with pooling layers enforced spatial invariance which is important for seizure classification since seizures can occur in any localized region of the brain, or globally [164]. Thodoroff's reason for incorporating a bi-directional layer was because neurologists typically use both past and future information to make a diagnostic decision on whether and EEG segment contained epileptic activity [164]. Thodoroff's application domain resulted in a minor limita-

tion which needs further investigation if this technique is to be applied in real-time workload classification because the task for this study was to use all the data to classify seizures. When all the data is available, bi-directional models can be used with impunity. However in a real-time classification task, the future information is not yet available. Therefore, care must be taken to have a model in which the bi-directionality updates respect the lack of future knowledge - updates can occur backwards from the present towards the beginning of the current temporal-data stream, and separately, forward from the beginning of the temporal-stream to the present.

Of all research discussed thus far, none have used an ensemble of participant-specific, individually-trained models despite excellent performance of ensembles in other domains where distributional differences are present. Fazli et al. [56] used existing BCI data from 45 individuals across 90 sessions to train an ensemble of classifiers to identify imagined right hand versus left hand movement. The goal in this stimulus-aligned experiment, was to create an ensemble which could handle cross-participant distributional differences and classify new participants with no prior data from the new participants. After training on this set, a separate hold-out set with 29 individuals and 53 sessions was used to assess model performance against various baselines. Final ensemble weightings of individually-trained Linear Discriminant Analysis (LDA) models were determined using ℓ_1 regularized quadratic regression to select and reduce the number of classifiers in the ensemble to relevant ones [56]. Cross-validation was used for model tuning [56]. Their results indicated that using ensembles of individually-trained classifiers can improve classification accuracy over traditional group-trained models (30.1% and 36.3% error respectively) and perform comparably to models trained and tested on the same individual (28.9% error) [56].

In summary, choice of model type and training methodology have been shown to have an effect on cross-participant EEG analysis across a variety of applications.

Ensemble methods, CNNs, and RNNs have generally improved results. However, no comparison using different training techniques has been characterized for deep neural network models. Additionally, aside from medical applications, cross-participant research using deep architectures used some form of stimulus to time-align the signals for analysis. A drawback to stimulus-aligned models is that most human tasks in real world environments do not experience time-locked stimuli; instead, humans often work in multi-task environments and make arbitrary decisions when to switch attention or tasks–exemplified by the MATB environment. In these unconstrained environments, obtaining temporal-specificity on environment changes or task switches is difficult, and we estimate that models that require stimulus-aligned information will have difficulty performing well in such environments. Since deep neural network techniques have not yet been applied to non-stimulus-aligned task environments such as the MATB for cross-participant analysis, performance in these environments is unknown. Furthermore, performance of an ensemble-of-individual-participant models has not been characterized using deep neural networks despite their effectiveness as described by Fazli et al. [56]. Finally, while shared-data modeling methods are commonly used in research when the number of participants in a study is low, ultimately the field should move toward zero-data methods because they do not require model refitting each time predictions are to be made on a new individual.

In the next section additional advances in deep learning architectures are described which will be fruitful in addressing some of the shortcomings of existing research.

5.1.2 Applicable Advances from Computer Vision (Multi-path Modules and ResNets).

Two of the primary advances in recent years that have pushed the computer vision field to new heights of accuracy are multi-path subnetworks and deep residual

networks (ResNets). These advances went mainstream in the different variants of Szegedy's GoogLeNet architecture, which use multi-path subnetworks, and the development of the numerous instantiations of ResNet [74, 75, 156, 157]. The main idea behind GoogLeNet is the *inception* module. Each *inception* module considers what type of local structure is required for a vision task since it can then be repeated spatially [156]. The most important contribution associated with the inception module was the idea of enabling multi-scale processing at a local level. To do so, each inception module takes the output tensor from the previous layer and passes it through several different-sized convolutional layers as well as one max-pooling layer in parallel, with appropriate padding to ensure consistent output dimensions [156]. The output from each of these operations are like-sized feature maps with varying depths that can be concatenated in the depth dimension. This results in a diverse set of features being learned at each layer.

He et al. [74] developed ResNets to address optimization problems associated with training very deep networks. The main idea behind a ResNet is to introduce unparameterized identity skip connections that change the layer-wise learning problem to one of learning the residuals rather than trying to learn an unreferenced output. Residual learning can be mathematically described as:

$$y = F(x, W_i) + x, \tag{7}$$

where y is the output vector, x the input vector from the unparameterized identity skip connection (added to force a residual mapping), and F is the function learned by the intervening layers as a non-linear function of the input vector and weights, W_i [74]. He et al. [74] compared training/validation performance of 18 and 34 layer networks each of which had two incarnations: a plain network (no skip connections) and a version whose only difference was the addition of the skip connections every two

layers to force a residual mapping. They found that their training error for a 34-layer network was higher than the training error of an 18-layer network for the non-ResNet networks and confirmed it was not due to vanishing gradients by checking the gradient norm at each layer [74]. This indicated that the optimizer was unable to find a solution as good as a simple 16-layer identity mapping (input layer, 16 identity mapping layers, output layer) which led them to believe the problem with training deep feedforward convolutional neural networks was due to challenges with non-convex optimization [74]. Empirically, they discovered that using residual learning resulted in network optimization working well even for extremely deep networks. Furthermore, they determined that increasing network depth seemed to monotonically decrease classification accuracy up to networks of 1000 layers; showing that adding network depth improves representational learning [75].

He et al. [75] also proposed pre-activation: a new ordering to apply batch normalization, activation functions, and then convolutions. This ordering led to a dramatic 35% reduction in test error as network depth increased. An important consideration when using pre-activation in a network is how to handle the first and last activations. It is recommended to initially use a convolution directly followed by an activation for the first elements in the network [75]. Following the last element-wise addition in the network, an extra activation function is recommended [75]. Furthermore, it is important to note that dropout should not be used along the skip connection [75].

Szegedy, et al. expanded upon He's work and applied residual networks to their GoogLeNet architecture [155]. They showed that scaling the residuals by a factor of between 0.1 and 0.3 improved stability during training and suggested this technique rather than making large adjustments to the learning rate for tuning the training. They also found that incorporating residual connections into their multi-path architecture caused training time to decrease drastically [155].

In summary, applicable advances from computer vision research include:

- Distribute computational load evenly across layers, while scaling network depth and feature map depth together with appropriate reductions in individual feature map size [157].
- Instead of using large convolution kernels, stack layers of smaller kernels to attain the same receptive field with significantly reduced parameters and computational requirements [157].
- Use batch normalization to produce stable input distributions prior to activation functions [75].
- Reduce computational load and overfitting, and improve interpretability by using global average pooling [107, 157].
- Use full-length kernels in one dimension to identify correlations among distant regions depending upon the tensor's structure [102].
- Scale residual activations by a factor of 0.1 to 0.3 prior to summing with the identity connection from the previous layer to improve training stability [155].

5.1.3 Applicable advances from natural language processing (LSTMs and Bidirectional LSTMs) .

RNNs have been used in natural language processing due to their ability to account for temporal context in speech and text. However, vanilla RNNs have struggled to model longer contexts well due to the mathematical sensitivities during the training process. These problems are referred to as vanishing or exploding gradients, and they lead to poor performance of vanilla RNNs. An LSTM is a gated recurrent neural network that was designed to overcome the vanishing or exploding gradient problem

[139]. The LSTM was developed by Hochreiter and Schmidhuber [81] and improved with the addition of a forget gate by Gers et al. [60]. A recent, detailed description of the LSTM is available in Hefron et al. [78].

One limitation of the LSTM is that it can only process the temporal stream of observations in the forward direction, making it inefficient for the network to learn how earlier elements in the stream might be conditionally dependent on later elements. Bidirectional LSTMs were developed to address this inefficiency. A Bidirectional LSTM (BDLSTM) can be conceptualized as two LSTMs with their outputs concatenated at each time step in the sequence. What makes it bidirectional is that one LSTM processes sequences in their original temporal ordering while the other processes sequences with the temporal order reversed. The advantage bi-directionality brings is that future context can inform past events. This even works for real-time environments, because the current state may be understood more clearly by processing older data in the context of newer information.

Bidirectional networks have proved useful in a variety of natural language processing tasks [65]. Because humans think in terms of language and plan actions with an understanding of temporal context, it is a reasonable expectation that brain activity may be better understood by considering its bidirectional context; bidirectional networks may perform well in human state assessment tasks.

5.2 Materials and Methods

The research in this chapter focuses on a sub-goal of human state assessment: estimating cognitive workload. To accomplish an exploration of new deep architectures which perform well in assessing cognitive workload, a dataset was first obtained. The remainder of work can be divided into four efforts: data preprocessing, network architecture design, network training, and development of a performance evaluation

strategy.

5.2.1 Dataset.

The dataset was collected during a human research study focused on neural sensor fusion. Eight participants completed four blocks of tasks over the course of two sessions in a single day with two blocks being performed in each session. One participant was left handed and two of the eight participants were female. Participants varied in age between 19 and 27 with a mean of 21.9 and standard deviation of 2.57 years. Each block in this experiment consisted of five-minute long trials of high-workload MATB [44], low-workload MATB, as well as verbal N-back 0, N-back 1, N-back 2, and N-back 3 presented in different orders within each block to control for fatigue and/or practice. This resulted in 30 minutes of data per participant per block. This study only analyzed the MATB data.

All four MATB subtasks were performed during both the low and high workload conditions. Temporal density of stimuli was adjusted to ensure a difference in task difficulty and performance between the two conditions. The low difficulty task setting was the same for all participants while high difficulty settings were individually tuned. Four 2-factor ANOVAs were used to determine if there were differences in mean performance metrics between low and high workload MATB tasks blocked by participant and with allowance for a participant by workload interaction. The two factors were workload with 2 levels (low and high), and participant with 8 levels: one for each participant. Each ANOVA corresponds to a subtask performance metric gathered during all trials in the experiment. The tracking task metric represents the percentage of time the aircraft flight path vector was in the desired region. The system monitoring task metric is the percentage of abnormal lights and dials conditions for which the user made the appropriate correction prior to the condition timing out.

Table 14. Summary statistics for performance metrics and 2-factor ANOVA results evaluating if there is a difference in mean human performance between low and high workload by task. Bold denotes statistically significant results at the $\alpha = 0.05$ level of significance.

	Low Workload		High Workload		
Task	Mean	Std Err	Mean	Std Err	Prob>F
Tracking	95.2	4.1	64.3	7.8	**<0.0001**
System Monitoring	93.6	3.3	76.7	8.1	**<0.0001**
Communications	93.5	14.0	95.9	8.2	0.5783
Resource Mgmt	82.9	20.9	80.2	16.2	**<0.0001***

*Significant difference between low and high workload for only participants 3 and 7.

The communications metric tracks percentage of correct responses to auditory inputs. The resource management task metric represents the proportion of time both tanks were within operational limits.

Table 14 shows that statistically significant differences in mean performance existed between low and high workload conditions for the tracking and system monitoring tasks for all participants. Additionally, significant differences in low versus high workload mean performance for the resource management task were present for only participants 3 and 7.

Participants were trained to asymptotic performance in the tasks during five training days prior to the test day to ensure learning effects were minimized during collection of the experimental data. The ordering of the scenarios varied by participant and by session in a randomized block design to control for ordering effects. Five-minute long baseline trials were performed at the beginning of each block and a 30-minute break separated the second and third blocks. Baseline segments were not used in the analysis. For each of the participants in the study, 128 EEG channels conforming to the BioSemi equiradial layout, along with 2 mastoid electrodes, 2 Horizontal Electrooculograph (HEOG) channels, 2 Vertical Electrooculograph (VEOG) channels, and 2 electrocardiogram (ECG) channels were recorded at a sampling rate of

4096 Hz using the BioSemi ActiveTwo recording system (BioSemi B.V., Amsterdam, The Netherlands). The skullcap was never removed during the test day and electrode channels were monitored throughout each session to ensure the highest data quality. Although not analyzed, 38 functional Near-Infrared Spectroscopy (fNIRS) channels were simultaneously recorded during the experiment and were interspersed with the EEG electrodes. Analysis was performed both including and excluding the single left handed individual due to possible motor cortex and prefrontal cortex differences. Since no significant changes in results were present based on inclusion or exclusion of this individual in analysis of the MATB experiment, the left handed participant's data was used for all analysis in this study.

5.2.2 Data Preprocessing.

A substantial amount of preprocessing was required. It largely focused on cleaning the data, downsampling both spatially and temporally to reduce input feature space, calculating PSD, performing normalization, and populating input-tensors of various shapes to match the desired structure for each of the architectures being explored.

All participant's data were trimmed to 303 seconds per trial, downsampled to 512 Hz and down-selected to 64 electrode channels (approximating the extended EEG 10/20 montage) to reduce computational complexity. The PREP pipeline [21] was used to identify and interpolate bad channels, calculate a robust average reference, and remove line noise. A high-pass filter with a cutoff at 1 Hz was then applied to the data. PSD at frequencies between 3 and 55 Hz inclusive were computed for each electrode channel using a 2-second Hanning-windowed Short-Time Fourier Transform (STFT) with an overlap of 1 second. This resulted in 3392 features for every 1-second observation.

Since the eventual goal is to use this type of system in real-world, real-time envi-

ronments, it is undesirable to remove data segments containing EEG artifacts because their removal would lead to breaks in prediction. Rather, the aim was to appropriately mitigate the effects of numerous high-variance artifactual segments that had potential to cause poor performance by identifying these sections and modeling through them. Since the purpose of this research was not to demonstrate a new artifact detection technique, high-variance segments were manually marked with the assumption that a real-time tool would be capable of marking similar segments. The high-variance segments were then temporarily ignored during normalization so that these sections would not negatively impact the normalization procedure. Due to the convolutional nature of the STFT, it was important to also ignore a window of +/- 1 second on either side of an artifactual segment in order to capture all points that were affected by a high-variance artifact during computation of PSD values.

To normalize the PSD values, an empirical Cumulative Distribution Function (CDF) was created for these values at each time step for each electrode/frequency combination within a given participant's block for the MATB task. An empirical CDF is a non-parametric normalization method which maps values ordered from lowest to highest into corresponding percentiles based on position. The computed CDF values were then used to transform the PSD values and the results were scaled to span the range $[-1, 1]$ rather than $[0, 1]$, as this was advantageous for neural network training.

Next, the dataset was repopulated with the previously-ignored high-variance artifact segments. Several possibilities were examined regarding reinsertion of these segments into the already normalized data. Empirically it was determined that best algorithmic performance could be achieved by replacing these high-variance EEG feature segments with zeros, and, for all architectures except CNNs, the data stream was augmented by concatenating a boolean bad-data indicator variable which was set to 1 in these segments and zero elsewhere. Due to the way parameters are shared

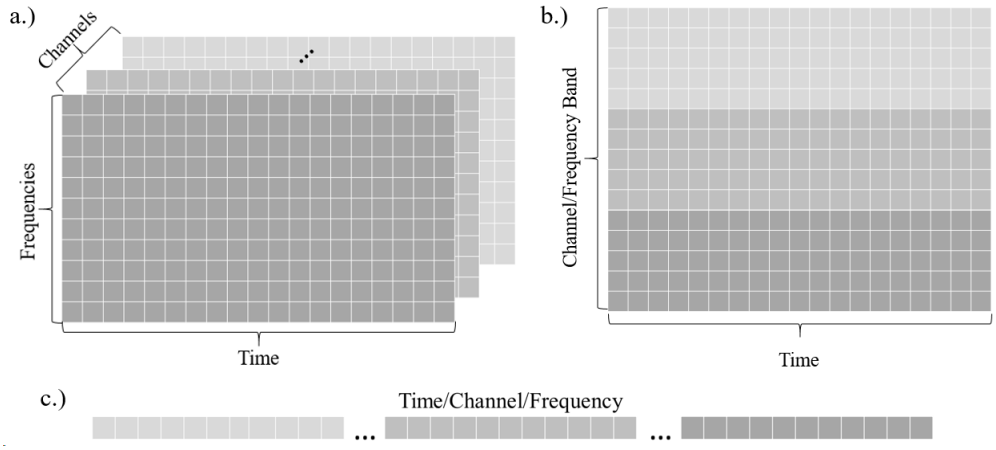


Figure 6. Input shapes for each network type are illustrated: a.) The CNN input shape was expanded about the channel (electrode) dimension as the number of filters increased. b.) The LSTM input flattened the electrode and frequency dimensions into a single combined dimension. c.) All dimensions were flattened into a single input vector for the ANN.

in a CNN, adding an indicator variable would have been detrimental, so this technique was not used for that architecture. An alternative normalization process would have been to apply the empirical CDF to all the data, including the high-variance segments. This was attempted, but these segments consistently produced saturated results of -1 or 1 which degraded overall system performance. By setting the high variance segments to 0 and the boolean indicator variable to 1, temporally stateful models are able to learn the best way to propagate the state through these segments.

Finally, rolling 5, 10, 20, and 30-second windows with a step size of one second between windows (4, 9, 19, and 29 seconds of overlap respectively) were created to prepare features for input to convolutional or recurrent networks. For the feedforward ANN, all features in the sequence were flattened and provided as a single vector to the network. Figure 6 depicts the different input shapes to the various types of networks.

5.2.3 Model Architectures.

Numerous deep neural network architectures were built to evaluate their effectiveness in a multi-task environment. Here, the design of seven neural network ar-

chitectures is briefly described. Detailed depictions of each architecture are available in Appendix C. Several architectures are derivatives of others, so they are described starting with baselines and building up in complexity. To avoid repetitiveness, all network specifications only describe the hidden layers and will forgo detailing the input layer shapes and sigmoid output layers used for workload classification.

Two neural network architectures were used to provide neural network performance baselines for the more complex architectures. A single-hidden-layer ANN was trained while varying the number of hidden nodes to examine validation performance for models containing between 1 and 6 million parameters for each sequence length. On average, models with 64 nodes performed the best on the validation data across sequence lengths. Hence, 64-node models were chosen as the final architecture. Appropriately-tuned ℓ_1 regularization (ℓ_1 rate = 0.00000125) proved to be very important due to the large number of features in the input vector–101,790 features for the 30 second sequence. A two-layer LSTM (2L-LSTM) was used as a second baseline. This architecture was used in a previous within-subject MATB study [78]. The hidden layers consisted of a sequence-to-sequence LSTM layer with 50 memory cells followed by a many-to-one LSTM layer with 10 memory cells.

The convolutional architecture (CNN) consisted of a series of 3x3 kernels with Rectified Linear Unit (ReLU) pre-activations, 2x2 max-pooling layers, dropout, and batch normalization layers. The architecture convolved and pooled across time and frequency dimensions in an attempt to have the CNN learn meaningful temporal and frequential representations of PSD. Following the final convolutional layer, a 2d global average pooling layer was used to reduce computational complexity by reducing the overall number of parameters prior to a fully-connected layer. This layer fed into a fully connected layer which found correlations between average time-frequency feature maps and sent them to the output layer.

The remaining architectures are all either variants of a five-layer LSTM or larger models which incorporate a five-layer LSTM into the overall architecture. Several variants of each of these architectures were evaluated based on validation performance including 50, 110, and 200 unit-per-layer LSTMs. The 110-unit-per-layer variant was selected and trained for all final models. All LSTM models used 20% dropout on the input sequence. The first five-layer LSTM model was simply connected to the output layer with no further modification. The second variant, a BDLSTM, examined the effect of bi-directionality on the model by making each layer a bi-directional LSTM without any other architectural modifications. The final variant, the Bidirectional ResNet LSTM (BDRLSTM), added residual connections around the middle three layers using an activation scaling factor of 0.3 to improve stability during training, as recommended by Szegedy et al. [155]. While Szegedy et al. [155] found that incorporating residual connections into their convolutional architecture caused training time to decrease drastically, we found a slightly increased number of training epochs were required to reach maximum accuracy using the BDRLSTM compared to the BDLSTM.

The final architecture is categorized as a MPCRNN. This is the most complex design of the seven. It combines a wide multi-path, residual, convolutional network with a bi-directional, residual LSTM. The architecture finds multi-scale time-frequency relationships by simultaneously convolving or pooling PSD values across a variety of convolution and pooling paths while maintaining temporal and frequential structure. The notion of electrode channel is abstracted as the depth of the network increases.

A feature present throughout the convolutional sections of the MPCRNN are 1×1 convolutions. Lin et al. [107] described how using 1×1 filters with a different depth than the preceding layer acts as a form of cross-channel parametric pooling. Szegedy et al. [156] used this idea to reduce the dimensionality of the inputs to the

larger filters and to create compressed representations of the previous layer with the additional benefit of adding a nonlinearity. Our architecture borrows this idea in two ways. First, each multi-path module has a path that includes only 1×1 convolutions which means that all the diversity of scale from the previous module passes through to the next module in a compressed form. This multi-scale processing allows for unique and differently structured time-frequency representations to be learned. Secondly, each module uses 1×1 convolutions to reduce dimensionality and evenly distribute the computational load among the various paths.

Figure 7 displays a modular view of the architecture, and a full schematic is present in the Supplementary Materials. The overall model begins with a 2d convolutional layer in time and frequency followed by a max-pooling layer which downsamples the frequency space by a factor of 2. This representation is then fed into the first multi-path convolutional module, *Module 1*, which is repeated once with each instance employing residual connections to improve gradient propagation through the deep architecture. *Module 1* begins with batch normalization, ReLU activation, and 20% dropout layers and then branches into 5 separate paths. Three paths have convolutional elements while the other two are pooling paths. All paths with convolutions utilize 1×1 convolutions to reduce the dimensionality of the inputs prior to the larger filters and to create compressed representations of the previous layer with the added benefit of adding a nonlinearity [107, 156]. The purpose of the multi-path structure is to place no undue preconditions on the way frequency or time are represented. The convolutional and pooling model only enforces locality of time and frequency, but lets the network learn the appropriate scales of locality to make the features the most discriminatory. Max-pooling and average-pooling were included as separate paths because for some frequency bands, the maximum PSD may be the best feature, while in others, the average PSD over the entire band may be more useful. Each

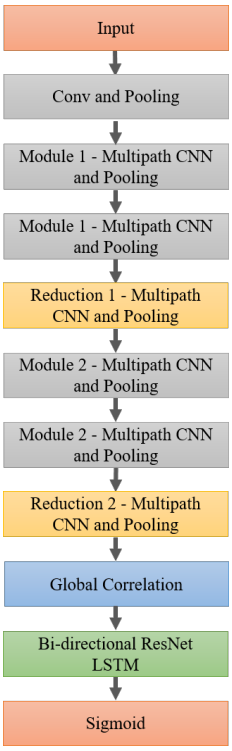


Figure 7. A modular depiction of the MPCRNN.

module also contains a path which includes only 1×1 convolutions which means that all the diversity of scale from the previous module passes through to the next module in a compressed form. At the end of each module, the various paths are concatenated, adjusted to the appropriate dimensionality via a 1×1 convolutional layer, and go through an activation scaling layer [155] prior to being added to the residually-connected features. *Module 2* is very similar to *Module 1* except that it handles different shapes since it is further downstream.

After each set of modules comes a multi-path frequency reduction module to downsample in frequency and expand in the channel dimension. After the series of multi-path and reduction modules, a 1×1 convolutional layer is used to reduce dimensionality prior to passing through a convolutional layer just prior to the BDRLSTM. This layer is unique because the kernel is the full width of the feature vector similar to the first layer of the architecture proposed by Lawhern et al. [102]. This finds

global correlations across combinations of abstracted-electrode/convolved-frequency-band representations for each time period. After a reshaping of the tensor, the features are ready to be processed by the same BDRLSTM to account for temporal context, as previously described. An additional benefit of using residual connections in this architecture is that they allow gradients to propagate via identity connections to any residually-connected layer in the network, thus bypassing many layers when needed. It is postulated that this is important for architectures which connect large recurrent networks on the end of convolutional architectures as it creates more direct paths to lower layers, thus improving the learning process.

5.2.4 Neural Network Training.

After preprocessing was completed, the seven architectures were initialized, trained, and tested using a variety of techniques. Several zero-data cross-participant training methods were compared: a cross-validated group method, a validation-set group method, and two variants of cross-validated ensemble methods. Four sequence lengths ranging from 5 to 30 seconds of PSD features were evaluated for each model to understand the effect of sequence length and network architecture design on zero-data cross-participant workload classification.

While several variations in model-training were used in the course of this study, numerous network training considerations remained the same. Describing the model development procedures and training aspects that were consistent across techniques is a logical place to begin.

All neural networks were trained using the Keras [37] and Tensorflow [1] frameworks. Model development was initially conducted manually using a patience-based early-stopping with validation set approach to tune model parameters such as regularization rate, number of hidden nodes, dropout rate, and architecture selection. This

was to reduce model development time. Upon selection of final architectures, optimal-stopping methods were used to determine the number of training epochs based on validation data. Each network was trained using mini-batch gradient descent with a batch size of 128 observations. The Adam optimizer was used due to its ability to handle non-stationary targets and noisy data [96]. Learning rate values ranged from 0.0001 to 0.000001, depending on the model being trained, to ensure smooth training. All models used a binary cross-entropy loss function. Several forms of regularization were applied to each model including dropout [150] and various levels of ℓ_1 or ℓ_2 regularization. Next, the four cross-participant training methods are presented. These training techniques were used for all network architectures and data sequence lengths.

5.2.4.1 Cross-Validated Group Method.

Cross-validated training of each group model began by iteratively sequestering one participant's data as the hold out test set. Next, 7-fold cross-validation was performed using each of the remaining 7 participants' data to tune number of training epochs. During cross-validation, models for each fold were trained for 30 epochs and were saved at each epoch during training. Validation accuracy was then averaged across folds at each epoch. The epoch with the highest average validation accuracy was selected as the best stopping epoch. After determining the best number of epochs to train based on cross-validation, the selected 7 participants' data was used to train a final group model to evaluate the hold-out test set (the 8th participant). This procedure was repeated until each participant had been used as the hold-out test participant once.

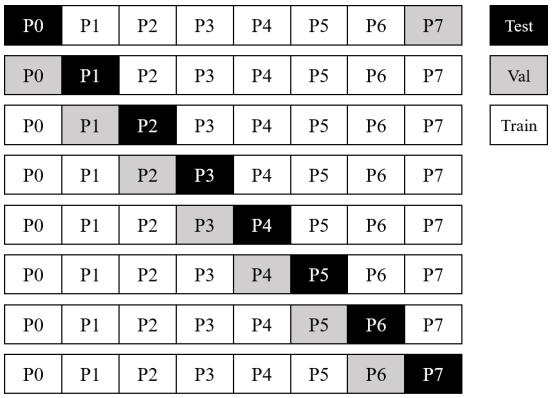


Figure 8. Training, validation, and test sets for the optimal-stopping validation-set group method are colored by use for each participant. For example, the fourth row indicates the model was trained using data from participants [0, 1, 4, 5, 6, 7], validated using participant 2's data, and evaluated using participant 3 as the hold-out test set.

5.2.4.2 Validation-Set Group Method.

Architectures were also trained using the optimal-stopping group method as depicted in Figure 8. Models were trained using six participants' data, validated using a seventh, and tested on an eighth participant in a leave-one-participant-out manner. In each case, the model was trained for 30 epochs with the model being saved anytime better validation performance was achieved. This optimal-stopping method differs from a patience-based approach often used to train deep networks in that the selected training epoch is the one in which the network's performance on the validation set was the highest over all epochs. This method contrasts with commonly-used patience-based early stopping, where an epoch counter is reset when a better performance is achieved. In the patience-based method, training continues as long as the epoch counter fails to exceed a pre-determined value. If a new minimum validation error is not achieved within that epoch count, training is stopped and the model associated with the best prior validation set performance is selected to train the final model. This final model is then used to make predictions on the associated hold-out-test participant's data.

Using optimal stopping instead of patience-based stopping can never result in a

worse validation-set accuracy. The optimal stopping method often yields a slightly better performing model than the patience-based approach, at the expense of an increase in training time. However, this additional computational expenditure was necessary to allow direct performance comparison with the cross-validated method which also used the optimally performing epoch found during cross-validation. All other network hyperparameters were consistent with those in the cross-validated group method.

5.2.4.3 7-Classifier Ensemble-of-Individuals Method.

A simple ensemble of single-participant models served as the 7-classifier ensemble method. Individual model development began by splitting each participant's data into the four blocks described in Section 5.2.1. Next, the number of epochs to train each of the 7 architectures, outlined in Section 5.2.3, was tuned for each participant using 4-fold cross-validation. Final individual models were then trained using all data for each participant. A maximum of 240 epochs was used in training each model to ensure an equivalent number of backward passes were available to fit the network as in the group method. A visual depiction of this process is shown in Figure 9 parts a.) and b.) respectively. The result was 8 final models (1 per participant) for each network architecture.

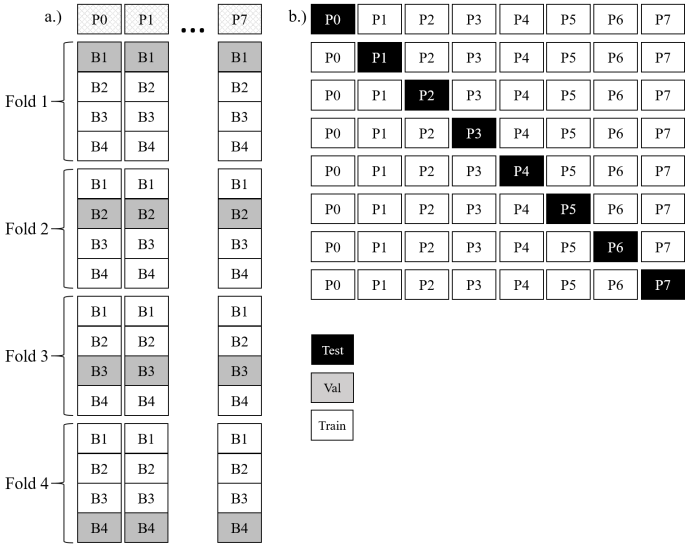


Figure 9. Ensemble Training: a.) 4-fold cross-validation across blocks was performed to tune hyperparameters for each within-participant model. Each of the models from these folds was also used as members of an ensemble of by-participant ensembles rather than continuing with part b. b.) Next, using each participant separately as the hold-out test set, the remaining individual's were used to train individual models using the hyperparameters derived from their individual model tuning in part a. Participants are not connected as they are in Figure 8 to highlight that they are each trained individually.

Throughout the training and cross-validation process, only one participant's data was available in each model. Because of this, any combination of participant models can be chosen to form an ensemble to make predictions on another participant not in the ensemble. An additional benefit of the ensemble method is that it is computationally more efficient than the cross-validated group method, requiring less than 2/3 of the computational resources and time as the cross-validated group method. By tuning the learning rate differently, a further reduction in resources by a factor of 8 could be realized without impacting classifier accuracy. While this demonstration was performed, the decision to maintain consistency between as many parameters as possible for the across-training-method evaluations was desirable and results were reported as described. Final predictions for the 7-classifier ensemble consisted of a simple average of all participant's individual probabilities for each class, with the

highest average probability being the predicted class.

5.2.4.4 28-Classifier Ensemble Method.

A second ensemble was formed using each of the four models per participant illustrated in Figure 3a. Rather than using each set of four models to determine a cross-validated stopping point and training with all the data for an individual participant, the overall ensemble just includes each of the four models across 7 participants for a total of 28 models in the ensemble. Since no tuning was accomplished using any other models, there are never interactions between the hold-out test set data (any single selected participant), and the 28 models associated with the remaining participants. Again, a simple average of class probabilities across the models in the ensemble was used to make predictions on the test participant. This was performed iteratively using each participant's data as the test set once. Testing this model against the others also allows for conclusions to be drawn regarding whether using a cross-validated stopping technique or an ensemble of by-participant-ensembles is more effective for handling distributional diversity attributable to individual differences.

This method required less computational resources than any other cross-validated method, reducing the 7-classifier ensemble computational requirement by approximately 20%. A final benefit to using ensemble training methods is that they are easy to update and improve as data from new individuals are obtained. This is because a new model is simply created and appended to the ensemble whenever new data is gathered. Conversely with the group training method, the entire model using all collected data needs to be retrained every time new data is added. This aspect of ensembles is especially attractive for real-world settings or long-term studies.

5.2.5 Statistical Evaluation Strategy.

Several ANOVAs with post-hoc Tukey Honest Significant Difference (HSD) tests were conducted to understand how network architecture, sequence length, and training method affect both mean classification accuracy and variance of cross-participant classification accuracy. Considering cross-participant variance of classification accuracy is required to appropriately characterize the effect of cross-participant distributional differences rather than merely examining overall accuracy. If a particular methodology reduces this variance while simultaneously improving mean accuracy, it could suggest a path to reduce the impact of cross-participant differences in future experimental designs. Unless otherwise stated, the results of the baseline CNN models were omitted during statistical analysis due to their poor outlying performance. Throughout the statistical testing, an $\alpha = 0.05$ was used as the threshold of significance.

Statistical evaluation began with two 3-factor ANOVAs used primarily to understand the effects of training methodology in the context of a given network architecture and sequence length. The independent variables in each ANOVA were: network architecture with 6 levels, sequence length with 4 levels, and training method with 4 levels. Main and first-order interaction effects were examined. A Tukey HSD test followed each ANOVA to identify which levels statistically differed and to determine direction and magnitude of the differences. A single outlying datapoint associated with LSTM model performance was omitted during the ANOVA and post-hoc tests in order to satisfy model assumptions. This datapoint is further discussed in Section 5.3.1.

Because there were significant interaction effects associated with some of the training methods, two follow-up 3-factor ANOVAs were conducted. These ANOVAs were used to examine the effects of sequence length and architecture in the context of

the two recommended training methods: the cross-validated group model and the 7-classifier ensemble method. The independent variables in these ANOVAs were: network architecture with 6 levels, sequence length with 4 levels, and training method with 2 levels. While only main effects were examined for the mean-accuracy follow-up ANOVA, the interaction of sequence-length and training method was considered for the variance of cross-participant classification accuracy ANOVA. A Tukey HSD test followed each ANOVA to identify which levels statistically differed and to determine direction and magnitude of the differences.

5.3 Results and Discussion

Now the various aspects of zero-data cross-participant models are decomposed, key factors influencing model performance are dissected in order to characterize the domain, and recommendations for future work are made. Overall results are shown in Table 15 and Figure 10. The following meaningful findings are discussed in detail in the ensuing sections and should inform future EEG-based analyses:

1. The cross-validated group training method and 7-class ensemble method performed similarly across shorter sequence lengths, but diverged at longer lengths due to a reduction in training observations associated with the ensemble method. Despite this, these methods are clearly preferable to the remaining methods and are suggested for future cross-participant EEG modeling depending upon application considerations, quantity of within-participant data, and computational constraints.

2. Increasing sequence length improves cross-participant mean accuracy, but also increases cross-participant variance. This indicates that a reduction in distributional dissimilarities between individuals cannot be achieved by increasing

Table 15. Mean classification accuracy across participants for a given training method, sequence length, and architecture. Number of parameters per model is displayed below each neural network architecture. Measured computational costs normalized to the cross-validated group model are displayed below each cross-participant training method. The best performing architecture for each training method and sequence length is shown in bold.

Training Method {Normalized Computational Cost}	Sequence Length	MPCRNN (6.2M)	BDRLSTM (4.2M)	BDLSTM (4.2M)	LSTM (1.9M)	2L-LSTM (0.7M)	CNN (1.8M)	ANN (1.1-6.5M)
Cross-Validated Group Model {1.0}	5	**0.791**	0.768	0.760	0.766	0.771	0.635	0.749
	10	**0.820**	0.785	0.777	0.778	0.793	0.657	0.777
	20	**0.850**	0.839	0.839	0.811	0.820	0.732	0.834
	30	**0.868**	0.852	0.853	0.849	0.834	0.711	0.862
Optimal-Stopping Val-Set Group Model {0.12}	5	**0.781**	0.748	0.748	0.682	0.754	0.627	0.743
	10	**0.819**	0.786	0.785	0.689	0.782	0.599	0.774
	20	**0.834**	0.825	0.825	0.802	0.809	0.684	0.828
	30	0.850	**0.860**	0.846	0.838	0.824	0.691	0.855
7-Classifier Ensemble Model {0.65}	5	**0.791**	0.775	0.771	0.757	0.773	0.731	0.767
	10	**0.822**	0.806	0.805	0.789	0.797	0.757	0.798
	20	**0.842**	0.814	0.803	0.833	0.798	0.757	0.834
	30	**0.865**	0.833	0.812	0.808	0.808	0.732	0.838
28-Classifier Ensemble Model {0.52}	5	0.768	**0.780**	0.778	0.765	0.773	0.701	0.769
	10	0.800	0.804	**0.806**	0.792	0.801	0.690	0.802
	20	0.809	0.807	0.816	0.826	0.799	0.731	**0.833**
	30	0.837	0.825	0.818	0.809	0.810	0.694	**0.841**

sequence length. To alleviate the problem of individual differences, a method which improves accuracy and decreases cross-participant variance is needed.

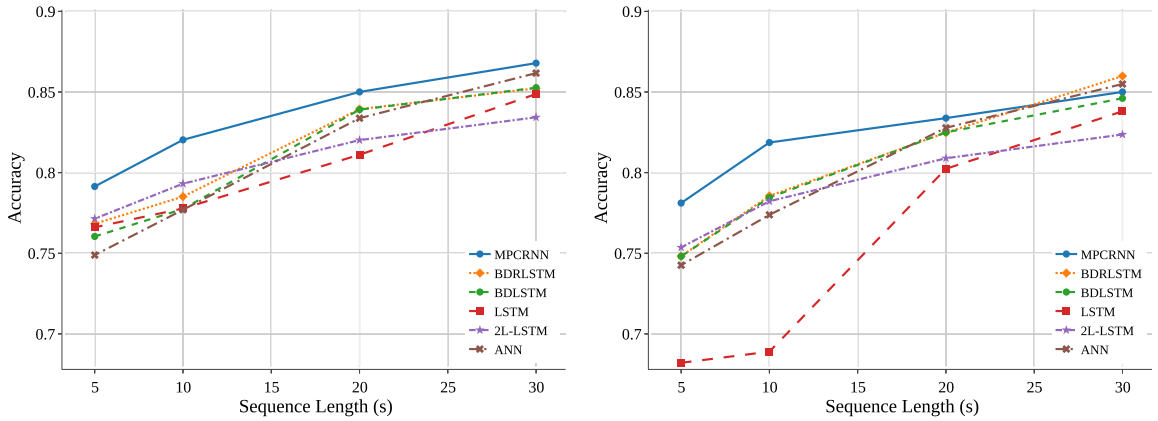
3. Compared to all other tested model architectures, the MPCRNN architecture both improved cross-participant mean accuracy and decreased cross-participant variance. This demonstrates that using domain-specific knowledge to inform deep neural network architecture design can reduce the impact of individual differences on model performance.

Results are presented by first explaining the effect of training methodology. Then a down-select to the two recommended training methods is accomplished in order to more clearly understand the effects of sequence length and network architecture on model accuracy and cross-participant variance.

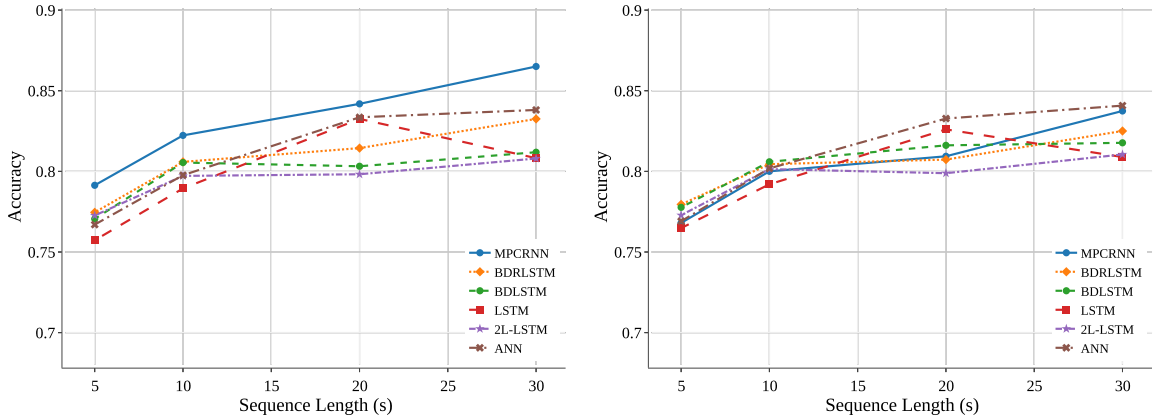
5.3.1 Effect of Training Method.

Table 15 and Figure 10 depict cross-participant zero-data classification accuracy for each model as a function of PSD sequence length and model training method. The most prominent trend was associated with sequence length: as sequence length increases, accuracy improves. However, gains in accuracy associated with longer sequences were influenced by the chosen training method. Results of the mean classification accuracy ANOVA showed that the interaction of training method and sequence length had a significant effect on cross-participant mean accuracy (Table 16, $p < 0.0001$). Table 17 highlights significant differences associated with the complex interaction between training method and sequence length. For 5 and 10 second sequence lengths, the validation set group model significantly underperformed all other training methods (all $p <= 0.0142$). At a sequence length of 20 seconds, the cross-validated group model resulted in improved accuracy compared to the validation group model ($p = 0.0427$) and the 28-classifier model ($p = 0.0090$). Finally, at the 30 second sequence length, both group models outperformed the ensemble models (all $p <= 0.0392$).

The complex interaction of training method and sequence length stems from the strengths and weaknesses of each training method. Ensemble models generally produced diminished gains in accuracy compared to the group models as sequence length increased. This effect is evident in the inferior performance for the ensemble methods at the 30 second sequence length as shown in Table 15. Compared to the group models, as sequence length increases, the amount of data used for validation of the ensembles consumes a larger fraction of the total training data available. This reduction becomes most critical when fitting complex models using the 28-classifier ensemble, which has only 2/3 the training data available compared to the 7-classifier ensemble, and appreciably less data than the group models. The reduction in perfor-



(a) Cross-Validated Group Model (b) Optimal-Stopping Validation-Set Group Model



(c) 7-Class Ensemble Model (d) 28-Class Ensemble Model

Figure 10. Mean classification accuracy for all participants as a function of sequence length, architecture, and training technique. Mean classification accuracy tends to improve as a function of sequence length for all classifiers and training methods. Error bars are omitted for clarity.

mance highlights an important consideration: determining the appropriate mixture of within-participant to cross-participant data to optimize predictive performance of the selected training method and architecture. If conducting an experiment with a large number of participants, it is suggested to gather twice the data from the first few participants. This way, models can be fit to those participant's data and it will be possible to examine the effect of varying the amount of training data available on a model's performance prior to collecting data from the rest of the individuals. Then,

Table 16. Results of the 3-factor ANOVA evaluating the effects of architecture, sequence length, training method, and first-order interactions on mean classification accuracy. Significant results are displayed in bold.

Factor	DF	Sum of Squares	F Ratio	Prob >F
Architecture	5	0.0090	10.0929	**<.0001**
Sequence Length	1	0.0709	398.2276	**<.0001**
Training Method	3	0.0021	3.9591	**0.012**
Sequence Length*Training Method	3	0.0073	13.7292	**<.0001**
Architecture*Sequence Length	5	0.0029	3.2372	**0.0116**
Architecture*Training Method	15	0.0039	1.4701	0.1449

Table 17. Tukey HSD pairwise comparisons for the interaction of training method and sequence length at specified sequence lengths. Significant results are shown in bold.

	5 sec				10 sec				20 sec				30 sec			
Comparison	Est	StdErr	t ratio	Pr>—t—	Est	StdErr	t ratio	Pr>—t—	Est	StdErr	t ratio	Pr>—t—	Est	StdErr	t ratio	Pr>—t—
28-Class 7-Class	-0.001	0.006	-0.192	0.9975	-0.002	0.005	-0.421	0.9746	-0.004	0.004	-0.854	0.8284	-0.005	0.007	-0.762	0.8713
28-Class CV Grp	0.011	0.006	1.800	0.2832	0.003	0.005	0.565	0.9421	-0.014	0.004	-3.282	**0.0090**	-0.030	0.007	-4.420	**0.0002**
28-Class Val Set	0.030	0.006	4.886	**<.0001**	0.019	0.005	4.034	**0.0009**	-0.002	0.004	-0.555	0.9448	-0.024	0.007	-3.500	**0.0047**
7-Class CV Grp	0.012	0.006	1.992	0.2022	0.005	0.005	0.986	0.7579	-0.010	0.004	-2.428	0.0823	-0.025	0.007	-3.658	**0.0029**
7-Class Val Set	0.031	0.006	5.074	**<.0001**	0.021	0.005	4.447	**0.0002**	0.001	0.004	0.293	0.9912	-0.018	0.007	-2.738	**0.0392**
CV Grp Val Set	0.019	0.006	3.120	**0.0142**	0.016	0.005	3.481	**0.0050**	0.011	0.004	2.704	**0.0427**	0.006	0.007	0.920	0.7941

an informed decision about whether to reduce data collection for the remainder of the participants can be made. Alternatively, this type of analysis can be completed using applicable preexisting data if available.

A significant interaction between training method and sequence length was also present in the cross-participant variance of classification accuracy ANOVA (Table 18, $p < 0.0001$). Visual inspection of Figure 11 corresponds with the ANOVA results from Table 18 and clearly shows the interaction effects. An interaction plot of training method and sequence length for variance is shown in Figure 12. It illustrates the transition of statistically significant differences in variance as sequence length increases. Based on pairwise Tukey HSD tests at each sequence length level, the ensemble classifiers result in significantly less variance at the 5 second sequence length than the group models (all $p <= 0.0069$), while the opposite is true as sequence length increases to 30 seconds (all $p <= 0.0005$). In the ensemble training methods, the strong trend of increasing inter-participant variance with increasing sequence length is again likely caused by the reduction in number of training sequences associated

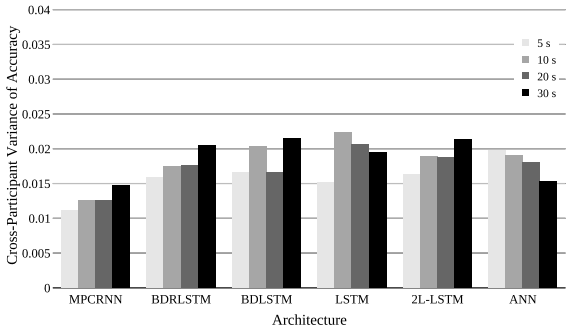
with an increase in sequence length. For ensemble models, this reduction is a higher proportion for each model than when all participants are grouped together. The result suggests to not only gather data from more participants, but also to gather more data for each individual if planning to use an ensemble method.

While the interaction of algorithm and training method was not significant, it should be noted that this was influenced by the removal of the 10 second sequence length LSTM point for the optimal-stopping validation-set group model whose accuracy was exceptionally low in comparison to all other models. Removal of this point could mask an interaction, but was required in order to satisfy ANOVA assumptions.

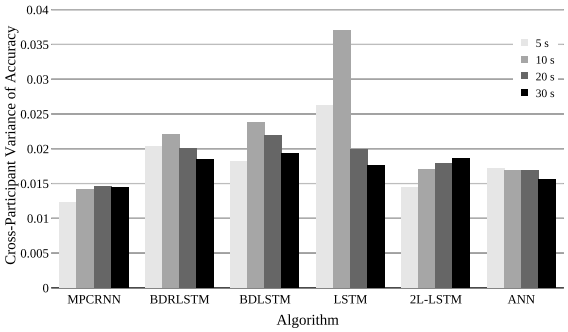
The statistical results do not tell the whole story regarding training technique. Digging a little deeper, it was discovered that the validation-set approach was likely far more prone to overfitting than even these results indicate. Over a fifth (21.4%) of the training runs resulted in a maximum number of epochs condition being reached using the validation-set method. Post-hoc analysis of each of the cases that reached maximum epoch conditions revealed that 70.8% of the corresponding test accuracies were in severe overfitting conditions with performance rapidly decreasing compared to the perpetuating modest improvements of the remaining 29.2% of cases. While computational constraints placed an upper bound of 30 epochs on training cases, had the validation-set approach been continued beyond the 30 epoch maximum, the performance gap between cross-validated and validation-set approaches undoubtedly would

Table 18. Results of the 3-factor ANOVA evaluating the effects of architecture, sequence length, training method, and first-order interactions on cross-participant variance of classification accuracy. Significant results are displayed in bold.

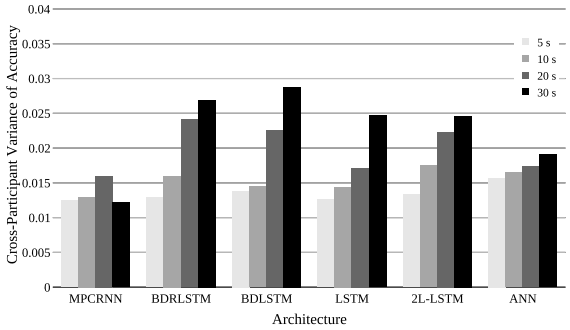
Source	DF	Sum of Squares	F Ratio	Prob >F
Architecture	5	0.00033601	16.3957	**<.0001**
Sequence Length	1	0.00042976	104.8501	**<.0001**
Training Method	3	0.00001822	1.4816	0.2283
Sequence Length*Training Method	3	0.00037315	30.3464	**<.0001**
Architecture*Sequence Length	5	0.00011529	5.6258	**0.0002**
Architecture*Training Method	15	0.00010311	1.6771	0.0796



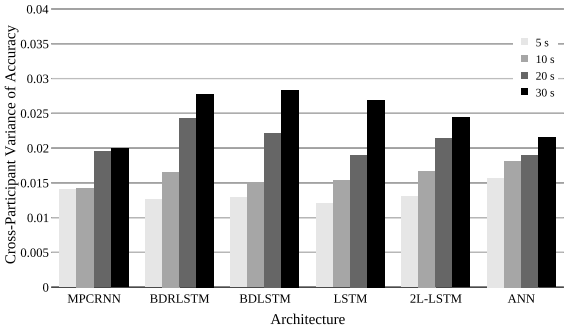
(a) Cross-Validated Group Model



(b) Optimal-Stopping Validation-Set Group Model



(c) 7-Class Ensemble Model



(d) 28-Class Ensemble Model

Figure 11. The plots of cross-participant variance of classification accuracy per architecture and sequence length show how much variance there was across-participants for a given training technique.

have widened since the cross-validated approach would have moderated these extremes through averaging. Additionally, using a cross-validated group-training technique resulted in improved accuracy compared to the optimal-stopping validation-set group training method for 26 out of 28 network architecture/sequence length combinations. While not examined in this chapter's analysis, future work could explore different methods for combining the number of training epochs from each fold during cross-validation, since taking the mean may not be the optimal method.

Finally, computational cost and experimental design considerations may guide a researcher to use a different technique than the cross-validated group method. Table 15 shows the computational cost difference between different cross-participant methods. The 7-classifier ensemble method was 35% more computationally efficient

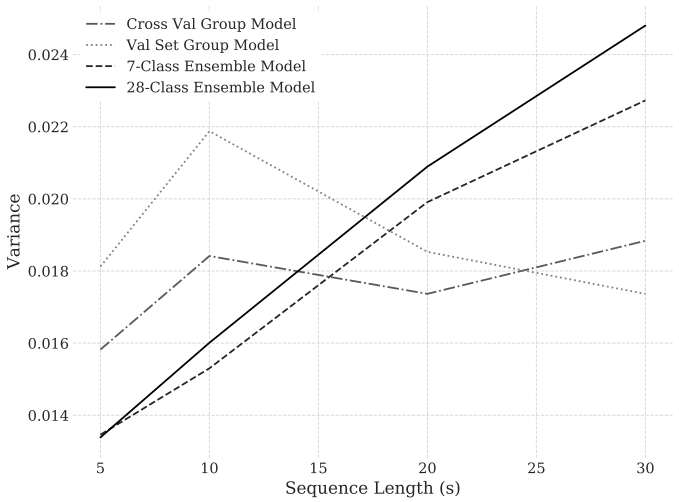


Figure 12. Interaction of sequence length and training method for cross-participant variance of accuracy.

than the cross-validated group method and resulted in very similar performance to the first two training methods for the MPCRNN. Ensemble training methods are recommended when the set of individuals being modeled will change over time. If arrival of new users is expected, the 7-classifier ensemble (1 classifier per individual) is desirable because each model in the ensemble only requires data from a single individual for the training process, making integration of the new individual's data into the existing ensemble an $O(1)$ process. By comparison, the cross-validated group training method requires that with each new participant the entire model be retrained using data from all participants: adding an individual model to an existing model with N participants is $O(N)$.

Due to the complex interactions associated with training method, for the remainder of this chapter's analysis only the two recommended training methods were considered: the cross-validated group method and the 7-classifier ensemble method. This greatly decreased the number of comparisons for the remaining factor analyses enabling a focused comparison of effects between these two recommended training methods.

5.3.2 Effect of Sequence Length.

Figure 10 depicts the clear and statistically significant trend indicating that when holding other factors constant, increases in sequence length lead to improvements in model accuracy. This trend highlights the tradeoff between temporal specificity and predictive accuracy across all classifiers as sequence length increases. A longer sequence length means there is a decreased ability to identify and temporally locate workload transitions in real-time, which is a goal of future research. In many applications, a reduction in classification accuracy associated with a reduced sequence length may be preferable to the loss of temporal specificity and the delay in predicting a change in the human's cognitive workload.

The 3-factor ANOVA results modeling effects on mean accuracy are shown in Table 19. The associated model explained a large proportion of the variance with an $R^2 = 0.823$. Parameter estimates of the model indicated that sequence length was the most significant factor ($p < 0.0001$) affecting mean classification accuracy with a 0.28% expected improvement for each additional second added to the sequence length.

Despite better predictive accuracy associated with longer sequence lengths, the variance among participants is not reduced. Based on a 3-factor cross-participant variance of classification accuracy ANOVA with first-order interaction terms for sequence length and training method (Table 20, $R^2 = 0.700$), all pertinent model parameters are positive, indicating that even after accounting for interaction effects, variance grows with increasing sequence length. This increase in variance occurs be-

Table 19. Results of the 3-factor ANOVA evaluating the main effects of architecture, sequence length, and two training methods (cross-validated group method and 7-classifier ensemble method) on mean accuracy. Significant results are displayed in bold.

Factor	DF	Sum of Squares	F Ratio	Prob >F
Architecture	5	0.0058	5.4780	**0.0006**
Sequence Length	1	0.0335	157.0921	**<.0001**
Training Method	1	0.0003	1.1776	0.2843

cause the additional data present in longer sequence lengths allows participants whose models perform well at shorter sequence lengths to improve their performance, while simultaneously failing to enhance generalization performance on those participants whose models performed poorly at shorter sequence lengths. The distributional dissimilarity between participants is causing the divergence and indicates that increasing sequence length will not alleviate the problem of individual differences. Rather, it is postulated that an increased number of study participants and/or a novel set of features that can account for differences in cortical parcellation between participants, or which can capture participant-specific differences in task strategy, may be required to meaningfully improve cross-participant stationarity.

Table 20. Results of the 3-factor ANOVA evaluating the main effects of architecture, sequence length, and two training methods (cross-validated group method and 7-classifier ensemble method), including the interaction effect of training method and sequence length, on cross-participant variance of classification accuracy. Significant results are displayed in bold.

Factor	DF	Sum of Squares	F Ratio	Prob >F
Architecture	5	0.0002	7.2180	**<.0001**
Sequence Length	1	0.0002	38.5229	**<.0001**
Training Method	1	0.0000	0.1122	0.7395
Sequence Length*Training Method	1	0.0001	15.6379	**0.0003**

An important nuance regarding the treatment of sequence length in this chapter compared to traditional treatments of increasing window length should be discussed. Previous results have indicated that longer windows for computing PSD tend to improve predictive performance [13, 4, 30]. However, these efforts used either average spectral power over increasingly long temporal periods, or merely increased the window length when computing PSD. This differs from the treatment of sequence length as a temporally ordered sequence of 2-second PSD computations with a 50% overlap, resulting in a 1 Hz update rate, as done in this study. This distinction is important

because it means that there is potentially less of a tradeoff in temporal specificity compared to previous methods. However, since workload transitions were not present in the data used for this chapter's analysis, it is not possible to determine if improved temporal specificity is present even at longer sequence lengths. This is left for future work.

5.3.3 Effect of Model Architecture.

The effect of model architecture was evaluated using the paired-down 3-factor ANOVAs (Tables 19 and 20) and associated Tukey HSD tests when appropriate. We first present results of the cross-participant variance of classification accuracy ANOVA followed by the mean classification accuracy ANOVA.

Table 20 indicates that architecture had a significant effect on cross-participant variance of classification accuracy ($p < 0.0001$). Significant results of an all-pairs Tukey HSD test comparing variance are shown in Table 21. The MPCRNN architecture produced models with less cross-participant variance than every other architecture (all $p < 0.0090$) while all other architecture pairs had insignificant differences.

Table 21. Post-hoc Tukey HSD results between architectures, for the cross-participant variance of classification accuracy ANOVA. Only significant results are shown (α=0.05).

Arch 1	Arch 2	Diff	Std Err	t Ratio	Prob>|t|	95% Conf Int
MPCRNN	BDLSTM	-0.0062	0.0012	5.02	**0.0002**	-0.0100 to -0.0025
MPCRNN	2L-LSTM	-0.0061	0.0012	4.87	**0.0003**	-0.0098 to -0.0023
MPCRNN	BDRLSTM	-0.0058	0.0012	4.69	**0.0004**	-0.0096 to -0.0021
MPCRNN	LSTM	-0.0052	0.0012	4.21	**0.0019**	-0.0090 to -0.0015
MPCRNN	ANN	-0.0046	0.0012	3.66	**0.0090**	-0.0083 to -0.0008

Table 19 showed architecture had a significant effect on mean classification accuracy ($p = 0.0006$). Table 22 presents significant architecture pairs of a post-hoc Tukey HSD test. This analysis revealed that the MPCRNN statistically outperformed all

other architectures in cross-participant model accuracy (all $p < 0.0443$). Classification accuracy was improved on average by 2.2% to 3.2% absolute depending the particular pair.

Table 22. Post-hoc Tukey HSD results between architectures, for the mean classification accuracy ANOVA. Only significant results are shown (α=0.05).

Arch 1	Arch 2	Diff	Std Err	t Ratio	Prob>|t|	95% Conf Int
MPCRNN	LSTM	0.032	0.0073	4.44	**0.0009**	0.0105 to 0.0542
MPCRNN	2L-LSTM	0.032	0.0073	4.37	**0.0011**	0.0100 to 0.0538
MPCRNN	BDLSTM	0.029	0.0073	3.93	**0.0041**	0.0068 to 0.0506
MPCRNN	ANN	0.024	0.0073	3.30	**0.0233**	0.0022 to 0.0469
MPCRNN	BDRLSTM	0.022	0.0073	3.04	**0.0443**	0.0004 to 0.0441

Due to the relationships between the MPCRNN, BDRLSTM, BDLSTM, and LSTM, analysis of these results suggest that there is a mean accuracy performance advantage associated with the addition of bidirectional context and residual connections to the 5-layer LSTM. However, the further addition of multi-path convolutional modules to the BDRLSTM resulted in an architecture which demonstrated superiority over all others. Notice that the only architecture which resulted in improved accuracy *and* decreased cross-participant variance compared to any other algorithm was the MPCRNN. This was significant because it means that the improvement in accuracy did not lead to the widening of cross-participant variance, as it did with sequence length. Rather, this architecture improved accuracy and led to a narrower difference between participants than all other tested architectures. This finding suggests that new feature representations can make headway on the challenge of building models that account for cross-participant distributional differences.

The MPCRNN performed better than all other architectures for both measures, indicating that these layers resulted in a learned representation which better matched assumptions about the data, and that those assumptions led to improved cross-

participant generalization. The multi-path convolutional layers enabled the model to find global spatial correlates of brain activity and to learn useful cross-participant frequency representations (rather than using only features related to power in the standard clinical bands), while at the same time maintaining temporal ordering so that the recurrent network could account for temporal context. Additionally, the multi-path structure enabled a diversity of scale and representation that would otherwise not be possible. With each multi-path module, adjacent frequency bands could be pooled using maximum or average pooling, or adjacent bands could be convolved with filters that learn how to best combine information present at different frequencies. The 1x1 pass-through convolutional layers in each module also compressed the content of the current representation and allowed it to pass through to the next module enabling a diversity of scale not present in any other architecture. It is posited that future use of multi-path architectures which enable not only frequential, but also localized spatial filtering, which, in combination with source localization techniques, may enable further improvements towards building a generalized architecture universally useful to multi-participant EEG analysis.

5.4 Conclusions

It is good to consider results in the context of outstanding challenges for a given field. In Section 5.1 the three main challenges associated with assessing operator functional state using psychophysiological features were discussed: temporal non-stationarity, individual differences, and cross-task applicability. The work in this chapter showed that using domain-specific knowledge to inform deep neural network architecture design resulted in a reduced impact on model performance due to individual differences. The cross-participant models produced a global reduction in cross-participant variance compared to within-participant models. Results in this

chapter also showed that performance decrements due to within-participant temporal non-stationarity can be largely overcome by using models which incorporate other individuals' data when sufficient diversity is present in the data.

For zero-data cross-participant modeling, it was found that while increasing sequence length improves model accuracy, it does not improve generalizability since cross-participant variance increases due to cross-participant distributional differences. Furthermore, longer sequences reduce temporal specificity which decreases a model's utility in a real-time environment. The only condition among the experiments across sequence lengths, architectures, and training methods which resulted in improved accuracy and decreased cross-participant variance was the multi-path convolutional recurrent architecture. The combination of multi-path convolutional layers with bidirectional context and residual connections enabled a diversity of scale and representation that better captured generalizable, cross-participant patterns of brain activity. Model training method was less important than other factors and should primarily be chosen based on experimental design and computational cost constraints; ensemble methods may be better if the underlying population being modeled will likely change frequently, but more data might be required per individual in order to offset the smaller amount of data available to train each model.

Results from this study suggest several avenues for future work to further improve cross-participant generalizability. Using models which learn more useful feature representations and gathering more data are the apparent paths for progress. A study employing a large number of participants ($n > 50$) should be performed to understand how well deep networks can integrate and generalize psychophysiological data on a greater scale. It is expected that with greater diversity in the data set due to individual differences, the performance gap between within-participant and cross-participant modeling will further reduce–and with enough individuals available, group

models may outperform individual models.

Unsupervised clustering of source localization across subjects using Independent Component Analysis (ICA) could also be attempted. This type of preprocessing may better align the input feature space with the assumptions of CNNs and also may improve recurrent network performance by pairing signaling from similar brain regions across subjects for each input feature. This would also enable a greater understanding of the underlying neuroscience at the expense of being able to perform real-time processing. Methods to weight ensemble models based on similarity of the input features to the feature distributions of each model in the ensemble could also be developed. This could allow for better cross-participant generalization by only using data from individuals with similar brain activity for a given task.

Future work is also required to improve cross-task applicability. Results from this chapter taken in context of other recent research which showed an improvement in temporal stationarity associated with the use of deep recurrent networks [78], demonstrates progress on two of the three primary assessment challenges using deep neural networks–temporal non-stationarity of features, and cross-participant distributional differences. The progress made on these two challenges suggests that deep neural networks may be able to learn representations that have generalizable features at various depths–features that could be used to improve the third primary challenge, cross-task model applicability. It is expected that transfer learning using deep neural networks may provide a fundamentally different result than transfer learning using other techniques because deep networks compose low level features into hierarchical representations which may capture similarities across domains and allow for modifications of features from one domain to become relevant in the other.

VI. Deep Learning Regression Approaches to Mental Workload Estimation in a Simulated Surveillance Task

6.1 Introduction

The measurement of neural activity to determine operator workload is a burgeoning field with many researchers developing systems to assess an operator's cognitive workload state [26]. Despite numerous researchers using the electroencephalograph (EEG) to assess workload, very few have used a regression model to predict quantitative workload values. Rather, most have opted to predict mental workload using classification, with two or three partitions of workload. While classification models provide utility in many domains, it is desirable to produce a richer continuous information stream of predicted operator workload when appropriate target data are available. In this work, several regression approaches for estimating mental workload using EEG signals are evaluated in a complex, realistic, simulation environment. Target workload values were derived from a video-based analysis of each participant performing a simulated Remotely Piloted Aircraft (RPA) Intelligence, Surveillance, Reconnaissance (ISR) task on a second-by-second basis [24]. These high-frequency workload targets enabled deep learning regression approaches which account for temporal context to be contrasted with models that make Independent Identically Distributed (IID) assumptions in a complex, multi-task, simulated environment with arbitrary mission workload transitions.

Regression-based approaches to psychophysiological workload estimation have been extremely sparse. They are typically not adopted for operator workload estimation using psychophysiological features for several reasons. First, workload ratings are usually recorded based on subjective questionnaires administered only after task completion and represent either an average or maximum value experienced during the

execution period [135]. In this case, temporal specificity is lacking and it is sensible to group the entire trial into an approximate class for prediction. Furthermore, while highly correlated [16], the specificity of responses to these subjective questionnaires varies somewhat on test-retest measures which means it may be more reasonable to group the ratings into classes. A final reason classification methods are used is that often experimental procedures are designed or tuned to elicit class-based responses. However, there are drawbacks to classification approaches. Classification methods are often treated as nominal rather than ordered categories, which implies equivalent costs between errors. With regard to workload, this poses two problems: 1) Experienced workload is at least ordinal if not continuous which suggests that either a weighted classification or regression approach should be used as the number of discernible levels increase, 2) In multi-task environments, a new combination of tasks may be experienced during the test scenario compared to all training scenarios, yet may be within the range of workload experienced during training. In this case, classification methods cannot discern this new value of workload, but regression techniques may be successful since they output a continuous value. A final consideration is that in many environments, near real-time estimates are required for time-series data. Regression solutions that are stable and predict reasonable transitions between states may be preferable to classification approaches which can suffer from stability problems with regard to time-series data since they always have to predict a particular class.

A problem until recently was that a method for producing high-quality, high-frequency sequences of operator workload target values was not present in literature. This challenge was addressed by Borghetti, et al. [24]. By using a discrete event simulation model built with Improved Performance Research Integration Tool (IMPRINT), target workload values were produced for supervised training [24].

IMPRINT is a discrete event simulation that enables researchers to assign visual, auditory, cognitive, psychomotor (VACP) values to particular tasks being performed, whose sum represents an estimate of workload [121]. IMPRINT models can be considered a quantitative instantiation of multiple resource theory [134]. These IMPRINT models were built based on video of the operators performing the task. Their output generates a time series of workload values–workload profiles for the task. In this work, five models are trained on EEG data collected during an operationally relevant, simulated RPA ISR task involving searching, identifying, and tracking high value targets while responding to radio calls: naive, feedforward Artificial Neural Network (ANN), random forest, bidirectional Long Short-Term Memory (LSTM), and a Siamese-triplet network. Model performance was then compared both quantitatively and qualitatively for six participants each performing 16 trials.

Specifically, we answer the question, "Does modeling temporal context improve workload estimation in a realistic, multi-task environment with numerous workload transitions?" In doing so, Siamese-triplet networks are introduced for EEG analysis, a mathematical formulation for setting a variable margin for Siamese-triplet network use in regression environments is motivated and developed, and several feature saliency visualization techniques are devised which lead to both psychophysiological and algorithmic findings. This study provides the following contributions and findings to the field:

1. Deep learning methods which incorporate temporal context perform better in an environment with many workload transitions compared to methods which do not consider temporal context in terms of Root Mean Squared Error (RMSE) and correlation. This is an extension from previous work which demonstrated improved classification accuracy in a complex, multi-task environment without workload transitions [78].

2. Siamese-triplet network performance is characterized for EEG analysis and did not significantly differ from best-in-class bidirectional LSTM performance in terms of RMSE and correlation.

3. By analyzing margin considerations, a theoretical argument for using a variable margin for Siamese-triplet networks applied in regression settings is presented and a mathematical formulation is developed which can reduce the need for a hyperparameter search.

4. A feature saliency visualization technique is introduced which enables rapid visual determination of relative feature saliency for neural networks with grouped feature types.

5. Skewness and kurtosis of PSD distributions are salient features in an environment with workload transitions for some participants. Based on previous results, inclusion of these features does not have a negative effect.

6. Extension of a feature saliency technique [132] is used to examine bidirectional LSTMs and indicates that temporal feature saliency of EEG signals is tied to the depth of the path associated with backpropagation when using LSTMs. Differentiable neural attention mechanisms are suggested to eliminate this dependency in future work.

Related work is covered next, followed by a description of the datasets used in this work. Then methodology is explained including data preparation, model development to include introduction of Siamese-triplet networks, and statistical analysis details. Results and discussion follow, concluding with future work.

6.2 Related Work

There have been numerous applications of Recurrent Neural Networks (RNNs) to EEG analysis. For a comprehensive review, see Section 2.5.1. However, relatively few researchers have explored fitting a regression model to workload values based on EEG signals, and there is no scientific literature which describes the application of deep learning regression approaches using EEG to predict mental workload. Therefore, this section reviews only studies where shallow regression techniques were applied to EEG analysis.

Smith, et al. fit several regression model types (full tree regression, pruned tree regression, and random forest) to IMPRINT-generated workload profiles of two simulated RPA High-Value Target (HVT) reconnaissance mission difficulty levels [147]. Each trial consisted of two tasks: The primary task was to use a computer mouse to control the RPA's camera to track either one HVT (easy mission difficulty level) or two HVTs (hard mission difficulty level) fleeing an outdoor marketplace on motorcycles. Each of these primary tasks had the same secondary task which was to periodically respond to "radio calls" which prompted the participant for simple arithmetic calculations regarding flight parameters.

Smith, et al. evaluated cross-participant and cross-profile applicability of their models built using the aforementioned model types. For each model type, a cross subject model was built by training models based on six participants and evaluating on the seventh for the 1-HVT profile. Their results indicated that random forests significantly outperformed the other two models for cross-participant workload estimation [147]. While their results are not directly comparable to the work in this chapter due to different tasks and research objectives, their analysis indicates that random forests are a good baseline method to compare newly proposed techniques against.

Chaouachi, et al. [34] performed Gaussian process regression to fit a mental workload model to NASA TLX workload values for a variety of laboratory-based tasks. Power Spectral Density (PSD) features from 4 Hz to 48 Hz for each of four active EEG electrode sites were used as input features to fit the models. Based upon the fitted models, a series of learning activities involving math problems of varying difficulty were performed by each of the 17 participants in the study. Significant correlations between task difficulty, subjective workload rating, and predicted workload level were reported [34].

Ke, et al. [95] used a feature selection procedure coupled with Support Vector Machines (SVMs) to perform both within-task and cross-task ordinal regression for the N-back task. They found regression models to be preferable compared to classification for mental workload estimation due to performance and experimental design reasons. Mental workload does not have classes, but rather varies around particular levels instead of stabilizing at a specific value [95]. Ke, et al. [95] discuss that because both mental workload and an individual's interaction with the environment are continuous and dynamic, correlation is likely a more valuable evaluation criteria for mental workload than RMSE, assuming similar RMSEs, especially given the temporal mismatch between perceived and recorded workload changes. An approach similar to Ke et al. is advocated in this chapter, and correlation is examined as well as RMSE.

Operators' subjective self-reported workload ratings have been shown to be correlated with changes in task demands across a spectrum of tasks [121]. Since administration of these subjective scales is intrusive and would distract from the primary task, they are often administered at the end of tasks, or no more than once every few minutes. This poses a problem when trying to collect enough observations to adequately train machine learning models to predict workload at an operationally-relevant time scale–models which can make predictions every few seconds. Conversely,

discrete event simulation models can be updated at operationally-relevant frequencies, but often must be developed manually by painstakingly observing the task (or video recordings of operator behavior) and encoding the specific cognitive activities the operator performed during each second of the task. Clearly, this encoding of operator workload would not be feasible for tasks lasting more than a few minutes. Borghetti, et al. [24] introduced the notion of using a discrete event simulation scaffolding to generate estimated cognitive workload observations at a higher update rate than possible with subjective self reports alone. They produced a discrete event simulation using IMPRINT and validated it based upon a correlational analysis with NASA TLX ratings. This enabled high temporal-fidelity training of an operator functional state assessment system based on psychophysiological features.

Borghetti, et al. [24] used the same dataset as in this work to determine if workload predictions made by training a random forest using within-participant data would perform better than a random forest model produced using cross-participant data. Their results indicated a slight improvement in workload prediction using within-participant data. Borghetti, et al. [24] also discuss the difficulties associated with performing regression analysis on highly variable workload data and note that using only RMSE fails to capture the true utility of a machine learning approach since datasets tend to be unbalanced with a majority of points close to the mean. Because of this, little improvement over naive models were present [24]. Rather, they focused analysis on the ability to identify transitions between workload levels and the presence of secondary tasks [24]. In this work, correlation will be examined as a secondary metric in conjunction with RMSE.

Table 23. Surveillance scenario timeline with descriptions of simulation state and corresponding starting times and finishing times for each state from the beginning of the trial.

State	Description	Start	Finish
Trial Start	No HVTs present	0	0
HVT 1	HVT 1 is on screen	9	59
Radio Call 1	Radio Call 1 is heard	30	35
HVT 2	HVT 2 is on screen	69	119
Radio Call 2	Radio Call 2 is heard	90	95
HVT 3	HVT 3 is on screen	129	179
Radio Call 3	Radio Call 3 is heard	150	155
End Segment	Completion of trial segment	181	181
Other tasks	Tasks outside scope of this study	181	270
NASA TLX	NASA TLX ratings collected	270	420

6.3 Materials

Data from this study was gathered from two sources. The first source was an experiment conducted in the Air Force Vigilant Spirit Control Station, a simulated RPA ISR mission environment [83]. This source provided raw psychophysiological data to be used as features for supervised machine learning models. The second source of data were continuous workload profiles generated by Borghetti et al. [24] which provided estimates of cognitive workload for each second of the operator's task.

Six participants completed four trials per day, each 15 minutes in duration, over four test days, resulting in a total of 16 trials. Each trial had two different scenarios: one focused around a surveillance task while the other used a tracking activity as the primary task. The tracking activity is not used in this study, but was used as the dataset for Smith's study [147], and was described in detail by Hoepf, et al [83]. The surveillance portion of the experiment was used by Borghetti, et al. [24], and is the primary task used in this study. The surveillance task required the operator to

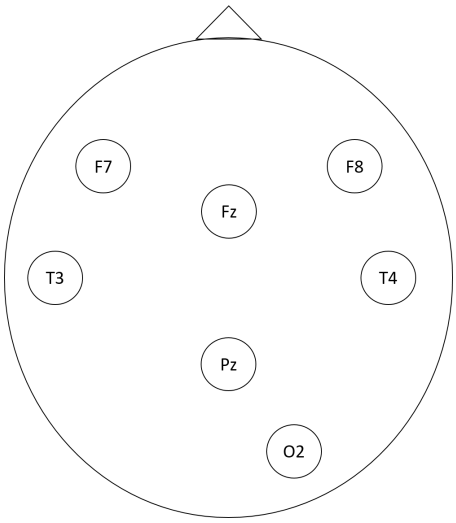


Figure 13. Seven EEG electrode locations used in this study located at F7, F8, T3, T5, Fz, Pz, and O2 sites, as specified by the International 10/20 system.

visually search for, identify, and track HVTs in a simulated outdoor marketplace by panning and zooming a simulated optical camera mounted to a RPA [63]. A secondary task was also present, requiring periodic responses to "radio calls" which prompted the participant to solve simple arithmetic calculations regarding flight parameters. Participants stated their answers to the calculations by using a push-to-talk button and speaking into a microphone. Only 181 seconds of each trial were used due to electromyograph (EMG) artifact presence caused by biomarker collection near the end of each trial. Additionally, NASA Task Load Index (TLX) ratings were conducted at the completion of each trial. The surveillance scenario timeline details when each event occurred in the scenario and is shown in Table 23.

Figure 13 illustrates locations of the seven electrodes which were used to collect EEG signals on a Cleveland Medical Devices BioRadio 150 with sampling at 480 Hz [83]. Vertical Electrooculograph (VEOG) data, other peripheral physiological data, as well as video and audio footage were recorded but not directly used in this study. Workload profiles were previously created using discrete event simulation with target values updated at 1 Hz. Rusnock, et al. [135] described the process for generating

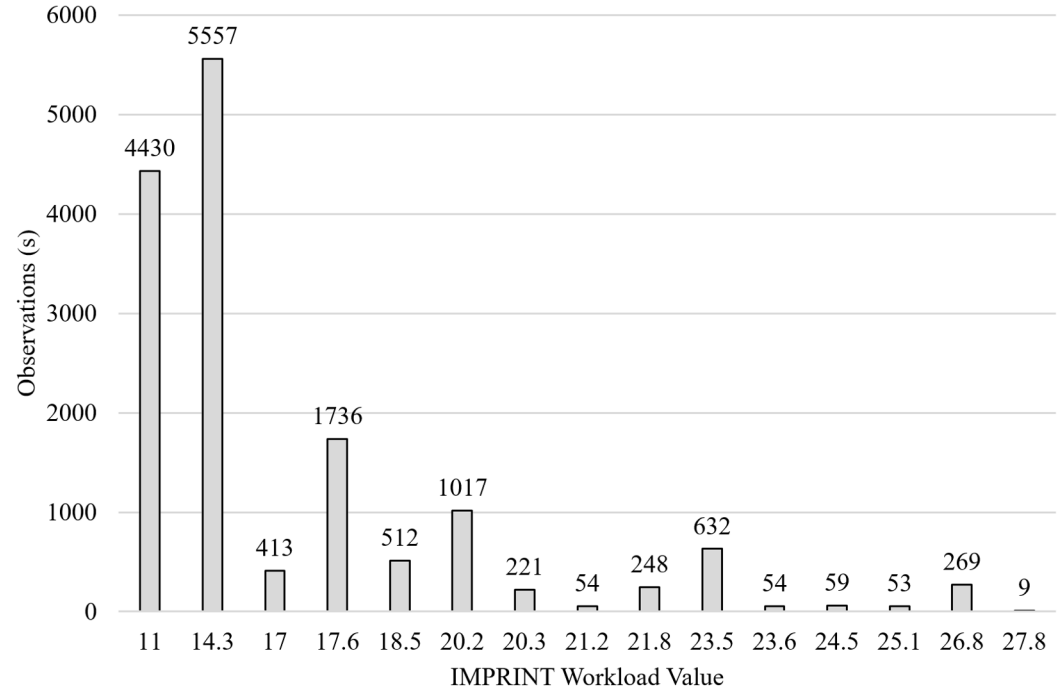


Figure 14. This illustrates the distribution of the target workloads created using the IMPRINT model. The total number of seconds across all trials at each workload value is displayed on the y-axis with the corresponding workload value on the x-axis.

these profiles using the IMPRINT model and validating them against NASA TLX ratings. The time-weighted average of these target values over the trial were strongly correlated with NASA TLX subjective ratings taken at the end of the trial with a mean correlation coefficient of 0.67 [24]. Target values were distributed as shown in Figure 14. This distribution shows that the majority of the time the participant was either tracking a HVT (11), searching for a HVT (14.3), or performing verification of a HVT (17.6) with no secondary task present. All other workload levels were induced by performing the primary task while listening, processing, calculating, or responding to the radio call associated with the secondary task.

6.4 Methods

Statistical properties of baseline-normalized time-varying frequency-band power distributions were previously demonstrated as being very effective features with regard to workload estimation [77, 78] and are used as input features for this study. Generally, mean or median PSD over a baseline period is used to normalize all other time-varying frequency-band power distributions for each channel [42]. The sixty seconds prior to commencement of the start of the trial was established as the baseline period and the median PSD for each channel-band combination was used to normalize distributions of time-varying frequency-band power. The median was selected due to significant variance within mean channel-specific time-varying frequency-band power during the sixty seconds.

Feature generation began by aligning the target workload profiles produced using IMPRINT with the raw EEG data. Procedures previously developed by Hefron, et al. [78] were then utilized, as summarized below. Preprocessing of EEG data began with conversion into the frequency domain and extraction of power in the clinical frequency bands to conduct time-frequency analysis. Oscillatory EEG signals were analyzed in the following bands: delta (3-4Hz), theta (4-8Hz), alpha (8-14Hz), beta (15-30Hz), and gamma (30-55Hz). The power spectral density was determined for 30 points spread out over a logspace from 3Hz to 55Hz by extracting power from complex Morlet wavelets [42]. Each wavelet was 2 seconds in length and the number of wavelet cycles increased logarithmically from 3 to 10 in conjunction with the frequencies. Mean power in each band was determined by averaging each power value for the evaluated frequencies within each of the clinical bands. Power was then aggregated over a two second sliding window with one second of overlap, allowing for a new update each second. A shorter sliding window size of 2 seconds was used in this analysis compared to other researchers due to the dynamic nature of the workload

target values [39, 101, 178]. Too long a window failed to provide sufficient temporal fidelity to discern transitions between states.

The mean, variance, skewness, and kurtosis of the two-second frequency-band baseline-normalized power distributions for each EEG channel were used as final features. This process yielded 140 features for each second and 2,896 observations per participant across the 16 trials. These features were then centered and scaled by trial so that each feature had a mean of zero and variance of one for a given trial. Next, model architecture and training considerations are presented for each type of model with special emphasis on Siamese-triplet networks since they have not been previously applied to EEG research.

6.4.1 Models.

Five categories of regression models were created to predict workload: Naive, random forest, feedforward ANN, bidirectional LSTM, and a recurrent Siamese-triplet model. In development of these models, it was assumed that each trial was independent from one another and that temporal dependencies across trials were random. Model training procedures can be broken down into three groups: naive models, random forest models, and supervised models using neural networks. The 96 naive models–one for each participant/trial combination–simply used the average value of all 15 training trials as the estimate for the workload value in the test trial. Using the average value minimized RMSE of the intercept-only models. Random forest models are discussed next, followed by ANN and bidirectional LSTM models. An extended consideration of Siamese-triplet networks concludes the section.

6.4.1.1 Random Forest Models.

Random forests are an improved version of bagged trees which create an ensemble of decorrelated trees by considering only a subset of randomly sampled features for every split in the tree [28]. Typically out-of-bag error is used to tune and validate random forest models because the extent of bias can be quantified and accounted for resulting in an unbiased estimate of error, whereas the amount of bias is difficult to estimate when using cross-validation [28]. However, due to autocorrelation within trials, out-of-bag error should not be used as an error estimator with EEG data and cross-validation is a better approach. The random forest models were developed using 5-fold cross-validation to select hyperparameters for each participant/trial combination. The folds were specified by withholding the test trial from the cross-validation set and splitting the 15 cross-validation trials into five groups of three trials each. Both the number of trees in the forest and the number of randomly selected predictors to consider for each tree split were optimized using a cross-validated grid search. The number of trees in the search varied from 10 to 200; while five values linearly spaced from 4 to 20 and centered about the recommended value–the square root of the number of features [88]–were evaluated to select the number of random splits. Lowest average RMSE across the five folds was used to select the best parameters for every participant/trial condition. This cross-validation procedure resulted in the evaluation of 9,600 models. The best hyperparameters were selected for each participant/trial combination and the final 96 random forest models were then trained (6 participants, 16 trials each). This procedure ensured optimal tuning to establish a valid baseline for comparison with deep learning techniques.

6.4.1.2 ANN and Bidirectional LSTM Models.

ANN and bidirectional LSTM models were tuned in a similar manner to the random forest model. For each individual, a hold-out test trial was selected and the remaining data was split into five separate sets used for 5-fold cross-validation. Five models were produced by retaining the weights associated with the best validation loss from each fold. Then a simple ensemble of these five models were used to make predictions on the hold-out test set in a manner similar to Hefron, et al. [76]. This procedure was repeated for each individual and trial resulting in 96 ensembles of five models each for both network architectures. Mini-batch gradient descent with 159 observations per batch was optimized using the *Adam* optimizer since it performs well with non-stationary targets and noisy data [96]. Model capacity was tuned to optimize validation set performance and to ensure approximately the same number of parameters were used in both models. Learning rate was decayed from 0.001 to a minimum value of 10^{-6} in 50 epochs for the bidirectional LSTM, and in 200 epochs for the ANN. Minority class oversampling techniques were investigated, but did not improve regression performance.

Bidirectional LSTMs processed 20-second long sequences of the aforementioned 140 features. Each model had two hidden layers each with 50 bidirectional units. The first hidden layer was connected to the second in a sequence-to-sequence manner. This was followed by a final linear output layer. The ANN was provided the same features as the bidirectional LSTM, except that the features were flattened into a 2,800 length vector (20 seconds of 140 input features). This input was connected to a 50-node fully-connected layer, followed by a 100-node fully-connected layer and a linear output layer. Each hidden layer used L_2 regularized Rectified Linear Unit (ReLU) activation functions. Model capacity was tuned to have approximately the same number of parameters as the bidirectional LSTM and overall architectures for

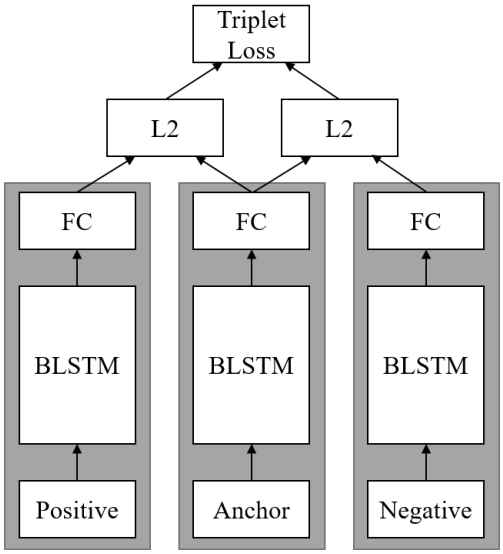


Figure 15. Siamese-triplet network where FC is the fully-connected embedding layer, and L2 is the L_2 distance between the connected embedding layers. Gray subnetworks denote layers where weights are identical.

both networks were determined by manual tuning. Next, background information on Siamese networks is provided followed by specific implementation details.

6.4.1.3 Siamese-Triplet Networks.

Siamese networks have been successfully used for a wide variety of tasks including signature verification [29], person re-identification [5], facial recognition [142, 154, 159], and one shot classification [97]. Siamese networks use a shared subnetwork to contrast examples and learn a useful embedding space. Siamese-triplet networks are a generalization of the Siamese network approach which use three subnetworks with shared weights, rather than two, as well as an associated triplet loss function [169]. A depiction of the Siamese triplet architecture is shown in Figure 15 with the shared subnetworks shaded gray. In this work, a Siamese-triplet network architecture is combined with LSTMs [60, 81] to process temporally-dependent data.

Siamese-triplet networks use their shared subnetworks to learn a useful embedding that clusters similar observations nearby while forcing differing observations to be

further apart in the embedding space. This is the purpose of the triplet loss function which was proposed by Schroff, et al. [142] and was an outgrowth of Weinberger and Saul's [171] work on margin-based distance metric learning for nearest neighbor classifiers. A randomly chosen example from the training set serves as an *anchor*, x_i^a, from class a. Two more observations are randomly chosen from the training set: one from the same class called the positive, x_i^p, and one from a randomly chosen different class called the negative, x_i^n. L_2 distances are calculated between the anchor-positive pair, $d_i(x_i^a, x_i^p)$, and the anchor-negative pair, $d_i(x_i^a, x_i^n)$:

$$d_i(x_i^a, x_i^p) = ||f(x_i^a) - f(x_i^p)||_2 \tag{8}$$

$$d_i(x_i^a, x_i^n) = ||f(x_i^a) - f(x_i^n)||_2 \tag{9}$$

where $f(x_i^a)$ is the embedding for the i^{th} anchor x^a, $f(x_i^p)$ is the embedding for the i^{th} positive x^p, and $f(x_i^n)$ is the embedding for the i^{th} negative x^n. Similar to Hermans, et al. [79], we empirically found Euclidean distance to be more stable during training rather than the more traditional squared Euclidean distance. The triplet loss function, $\mathcal{L}_i$, then minimizes the L_2 distance between pairs from the same IMPRINT class, $d_i(x_i^a, x_i^p)$, and pairs consisting of two different IMPRINT classes, $d_i(x_i^a, x_i^n)$, subject to a margin α:

$$\mathcal{L}_i = [d_i(x_i^a, x_i^p) - d_i(x_i^a, x_i^n) + \alpha]_+ \tag{10}$$

The goal of triplet loss is to form clusters in an embedding space for each class by bringing observations from the same class closer together while simultaneously pushing observations from differing classes further apart by at least a defined margin. In this way, triplets that already are farther apart than the margin, α, in embedding space, do not have an effect on the learned embedding during training.

Hermans, et al. [79] and Shirvastava, et al. [144] discuss several considerations regarding selection of anchor-positive and anchor-negative pairs that formulate a batch. These considerations stem directly from the effect of the margin and necessitate selection of pairs that violate the margin so that learning can occur. Those pairs which violate the margin are termed *hard examples*. Shirvastava, et al. [144] described the traditional way to mine hard examples as iterating over the following two step process:

1. Add new hard examples to the active training set based on some metric of difficulty until the desired number of training examples are accumulated.

2. Train the model until convergence based on the active training set.

The method for mining hard examples using in this chapter is similar to Shirvastava, et al. [144], except that hard-examples are carried forward from the batch that was just trained on to the next batch where they are again checked to determine if they are still hard. New hard examples are also mined to supplement those carried forward while maintaining a batch size of approximately 2048. Training until convergence on the active set often led to overfitting to a subset of hard examples and made training much more variable than adding additional hard examples as others satisfy the margin requirement and fallout of the set after each gradient update. Prior to further discussing the margin, it is important to understand that the Siamese-triplet network only functions as the encoder in an encoder-decoder scheme. A decoder is also trained to learn the transformation from the encoding space to the target output space of regression predictions.

6.4.1.4 Margin Selection.

Because the margin determines which examples will have an effect on the learned embedding, defining the margin is an extremely important aspect when using the

triplet loss function. Three factors that affect the margin are distinguished:

1. The dimensionality of the embedding space, N.

2. The range and distribution of the activation function used just prior to the embedding space, D.

3. The distance between anchor and negative points in the target output space as defined by the decoder loss function, y.

Many use a distance measure constrained to the unit hypersphere. While this is an option, it adds an additional constraint to the separation of classes translating the loss into a cosine similarity function rather than making use of the entire embedding space. On the other hand, tuning the margin becomes more challenging when working with an unconstrained distance measure. Despite this challenge, using theoretical considerations based on the three factors that affect the margin, a suitable value can be derived.

Using a predefined margin is a poor choice if there is significant variation in the target values. Rather, a variable margin is proposed so that the margin is still effective for anchor-negative distances that are relatively close compared to anchor positive distances. Otherwise, large-distance anchor-negative pairs dominate training causing separation challenges for distinct, yet nearby classes. The separation challenges only become problematic when class errors are weighted differently, as in the regression-based approach used in this chapter, and becomes even more critical if using a squared distance loss function since it magnifies scale differences. The following equation can be used to appropriately tune the margin for a given Siamese-triplet network given a finite-range activation function:

$$\alpha_i = \frac{\beta(x_i^a - x_i^n)^2\sqrt{N(max(D) - min(D))}}{(max(x^a) - min(x^a))^2} \tag{11}$$

where β is a hyperparameter used to determine how much separation is desired in the embedding space, x_i^a is an *anchor* from class a, x_i^n is a *negative* from class n, $(max(x^a) - min(x^a))$ represents the range of possible values of anchor classes, N is the dimensionality of the embedding space, and D is the range of the activation function just prior to the embedding space. Empirically, $\beta = 0.5$ was found to be an effective default value as it strives to separate classes by at least half of the maximum Euclidean distance across the embedding space. This value was used for all Siamese network results reported in this chapter. Next, specific details regarding the models trained for this study are discussed.

6.4.1.5 Siamese-triplet Models.

A Siamese-triplet architecture was developed which translated the 20-second sequence lengths of 140 features into a 3 dimensional embedding space by using 200 bidirectional LSTM units connected to 3 *tanh* nodes for the encoder. The decoder consisted of two densely connected layers consisting of 8 ReLU activations each and a final linear output layer. Multiple encoder and decoder architectures were examined during training including numerous high-dimensional embeddings; however, by reducing the embedding space to three dimensions, it was possible to observe clustering which proved helpful and insightful while training the network. Siamese-triplet network training used the same 5-fold cross-validation and ensemble scheme as used with the ANN and bidirectional LSTM models. The encoder was trained for 40 epochs. For each epoch of encoder training, hard examples were mined, as discussed in Section 6.4.1.3, and the decoder was trained for 200 epochs. The best overall combination of encoder and decoder, based on validation accuracy, were stored as training progressed.

An ancillary benefit of using a Siamese approach was that the method for hard

example mining implicitly solved the target-distribution imbalance problem. By selecting only the hard examples for training, the algorithm naturally balanced the required training examples to only those with challenging differences that would have an effect on the model. Uniformly sampling examples across all training classes and then picking the hard combinations ensured each observation for a particular class had equal opportunity to be encountered as either a hard positive or hard negative that forced a margin violation. As training progressed and the cluster boundaries changed, new examples from different classes resulted in margin violations. Regardless, each class had equal opportunity during each epoch to contribute to the active training set.

6.4.2 Analysis Strategy.

The results from each algorithms' 96 models were overall IMPRINT workload prediction values from 21 seconds through 179 seconds for the specified test trial. These were used to calculate RMSE and correlation values for each test trial. Algorithmic performance was then assessed using two complementary techniques. Statistical evaluation of results began by using a mixed effects model to explain the dependent variable, predicted RMSE, based on the fixed effects of algorithm while accounting for the random effects associated with participant-trial combination. This enabled appropriate modeling of the covariance structure associated with specific participant-trial combinations, but not due to algorithmic effects. Post-hoc Tukey Honest Significant Difference (HSD) tests were then conducted to determine statistically significant differences in fixed effects at the $\alpha = 0.05$ level of significance, and 95% confidence intervals are reported.

Next, correlation coefficients were compared using all-pairs, exact binomial tests to determine if it is expected for one algorithm to yield higher correlations than the other

Table 24. Average RMSE for each algorithm by participant and average RMSE for each algorithm across all trials and participants.

Algorithm / Participant	1	2	3	4	5	6	Overall
Bidirectional LSTM	3.58	3.42	3.77	3.45	4.05	3.82	3.68
Siamese-Triplet	3.61	3.51	3.78	3.57	3.98	3.91	3.73
RF	3.80	3.59	3.94	3.66	4.03	3.85	3.81
Naive	3.91	3.71	4.00	3.72	4.19	3.91	3.90
ANN	4.41	3.96	4.44	4.07	4.56	4.45	4.32

Table 25. Mean correlation for each algorithm by participant and mean correlation for each algorithm across all trials and participants.

Algorithm / Particpant	1	2	3	4	5	6	Overall
Bidirectional LSTM	0.40	0.40	0.34	0.40	0.31	0.30	0.36
Siamese-Triplet	0.38	0.34	0.35	0.32	0.32	0.18	0.31
RF	0.26	0.27	0.20	0.20	0.30	0.21	0.24
ANN	0.24	0.26	0.21	0.23	0.26	0.11	0.22

for a given participant-trial combination. Set up of the test involved transforming paired results into a dichotomous variable which specified whether the correlation coefficient of algorithm A was higher or lower than that of algorithm B. Once in this format, tests were performed using a Bonferonni-corrected significance level of $\alpha = 0.0083$ to control the familywise error rate at $\alpha = 0.05$. As Ke et al. [95] pointed out, correlation information can inform how well workload predictions generally track the variation in workload target values. As mentioned in Section 6.1, the use of workload profiles to produce target values makes the assumption that they reflect a participant's true perceived workload. By using correlation as a secondary metric, this assumption can be relaxed somewhat such that the perceived workload merely needs to track with changes in the participant's perceived workload.

6.5 Results

Model RMSE and correlation were recorded for each type of algorithm for each participant-trial combination. By-participant and overall average values for RMSE and correlation are displayed in Tables 24 and 25 respectively. Based on RMSE, bidirectional LSTMs performed the best with Siamese-triplet networks and random forests also performing better than naive models. On the other hand, ANNs tended to perform worse than the naive models. Results of the mixed effects model indicate that algorithm has a statistically significant effect on RMSE ($p < 0.0001$). Post-hoc Tukey HSD results with RMSE mean difference estimates and corresponding 95% confidence intervals are displayed in Table 26. These results show that all model pairs had statistically significant differences in mean RMSE performance (all $p < 0.0282$) with the exception of the bidirectional LSTM and Siamese-triplet pair ($p = 0.5503$). This indicates no statistically significant difference between the two algorithms that account for temporal context (bidirectional LSTM, Siamese-triplet), but that they have significantly better mean RMSE performance than all non-temporally aware models (random forest, naive, ANN). Similar to previously reported results [24], random forest models performed slightly better than naive models ($p = 0.0103$). The ANNs were the clear under-performers with results significantly worse than all other models including the naive approach (all $p < 0.0001$).

In previous work using this dataset, Borghetti, et al. [24] noted the difficulty of outperforming a simple naive model in terms of RMSE due to the fact that 70% of the data is clustered near the mean. While methods described here enable production of statistically significant differences in mean RMSE from baselines, this metric fails to capture correlated variation between the predictions and target values. This is quantitatively measured by the correlation coefficient between predicted and target sequences. Correlation results tell a similar story as RMSE, and show that all mod-

Table 26. Post-hoc Tukey HSD RMSE results between algorithms, for the mixed effects model. The better performing algorithm is listed in the Algorithm 1 column. Significant results are shown in bold ($\alpha = 0.05$).

Algorithm 1	Algorithm 2	Diff	Std Err	t Ratio	Prob>|t|	95% Conf Int
LSTM	ANN	0.595	0.029	20.70	**<.0001**	0.516 to 0.674
Siamese	ANN	0.551	0.029	19.19	**<.0001**	0.473 to 0.630
RF	ANN	0.467	0.029	16.25	**<.0001**	0.388 to 0.546
Naive	ANN	0.373	0.029	12.99	**<.0001**	0.295 to 0.452
LSTM	Naive	0.221	0.029	7.73	**<.0001**	0.143 to 0.300
Siamese	Naive	0.178	0.029	6.21	**<.0001**	0.099 to 0.256
LSTM	RF	0.128	0.029	4.46	**0.0001**	0.049 to 0.206
RF	Naive	0.094	0.029	3.27	**0.0103**	0.015 to 0.172
Siamese	RF	0.084	0.029	2.94	**0.0282**	0.006 to 0.163
LSTM	Siamese	0.044	0.029	1.52	0.5503	-0.035 to 0.122

Table 27. All-pairs binomial results for correlation. The better performing algorithm is listed in the Algorithm 1 column. Significant results are shown in bold ($\alpha = 0.0083$).

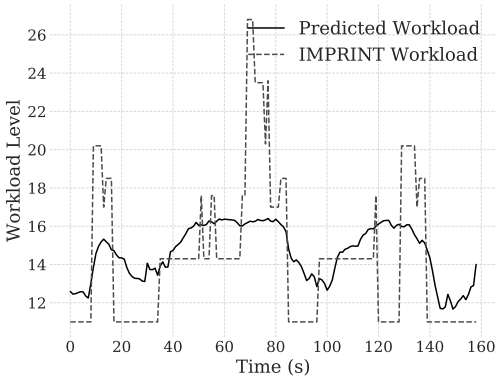
Algorithm 1	Algorithm 2	p-value
LSTM	ANN	**<0.0001**
Siamese	ANN	**<0.0001**
LSTM	RF	**<0.0001**
Siamese	RF	**0.0056**
LSTM	Siamese	0.0184
RF	ANN	0.6100

els, including the under-performing ANNs, are positively correlated with the target values. Results of pairwise binomial tests are shown in Table 27 and indicate that in a majority of instances, higher correlation is expected when using a bidirectional LSTM or Siamese-triplet model compared to a random forest or ANN for a given participant-trial combination (all $p < 0.0056$). No significant differences were found between the bidirectional LSTM or Siamese-triplet models nor between the ANN and random forest models. This shows a distinct separation in terms of correlation performance between temporally-aware and non-temporally-aware models.

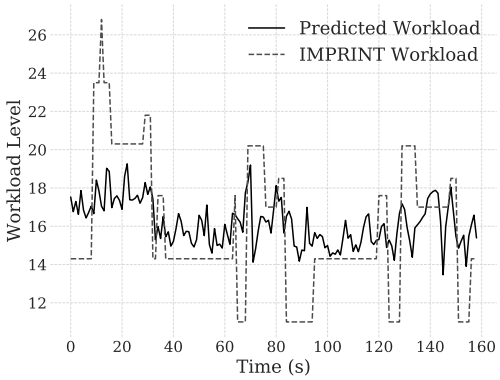
Qualitative performance can be evaluated by visually comparing plots of similarly performing participant-trial combinations for each algorithm. To enable a fair qual-

itative evaluation, plots of approximate best performance and approximate median performance are depicted in Figures 16 and 17 respectively. These plots were chosen for each algorithm by first sorting each algorithms' quantitative results by the ratio of the algorithm's RMSE for a given participant-trial combination to the corresponding trial's naive RMSE. This gives a quantitative measure of improvement over the naive approach since different participant-trial combinations result in different workload profiles. In addition to the RMSE ratio, correlation results are separately sorted by descending maximum correlation. Selection of the best performing plots required the given participant-trial combination to be present in the top five out of 96 values for both RMSE ratio and correlation criteria. Likewise, for the median performance plots, both criteria were required to be within the 10 closest values $[+5, -5]$ of median performance for both criteria.

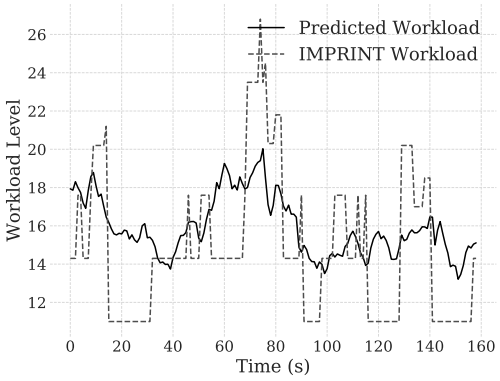
Qualitatively, several performance characteristics are immediately apparent. The first is that none of these algorithms capture the richness of the workload profiles' sharp transitions between states with accompanying stability between transitions. The bidirectional LSTM had the smoothest prediction results with reasonably sharp transitions between states. However, it failed to reach the extremes of workload when multiple tasks were present. The Siamese-triplet networks were slightly less smooth than the bidirectional LSTMs, but were still stable with little noise. Qualitatively, they were able to predict more extreme values without statistically significant RMSE or correlation degradation compared to bidirectional LSTMs. This makes them a strong candidate for future research. Both the random forest and ANN models resulted in noisy predictions. While the ANN was able to make reasonable predictions at the higher workload levels in Figure 16, this quality was not present in the median performance figure. Rather, the capacity for making sharp transitions merely resulted in a higher level of noise at the median performance point for the ANN.



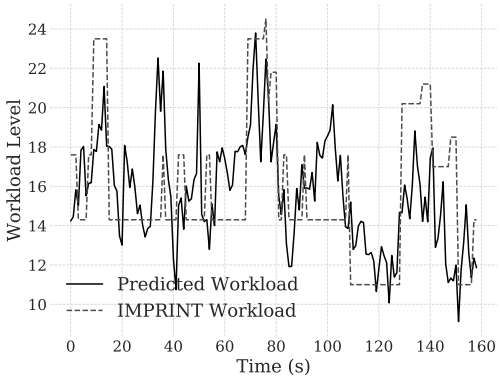
(a) Bidirectional LSTM: Participant 1, Trial 3
Corr: 0.59, RMSE: 3.19, Naive: 3.82, Ratio: 0.83



(b) Random Forest: Participant 5, Trial 9
Corr: 0.50, RMSE: 3.02, Naive: 3.47, Ratio: 0.87



(c) Siamese-triplet: Participant 5, Trial 15
Corr: 0.55, RMSE: 3.36, Naive: 4.01, Ratio: 0.84

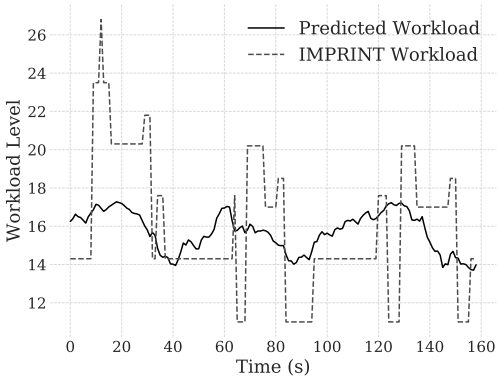


(d) ANN: Participant 5, Trial 2
Corr: 0.49, RMSE: 3.33, Naive: 3.66, Ratio: 0.91

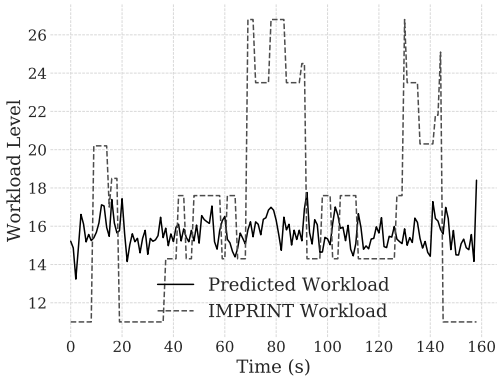
Figure 16. A comparison of approximate best RMSE and correlation performance for each algorithm. Performance values are reported in the following form: Correlation for trial; Algorithm-specific RMSE for trial $RMSE$; Naive RMSE for trial $RMSE_{Naive}$; RMSE reduction ratio compared to baseline ($RMSE/RMSE_{Naive}$).

6.6 Discussion

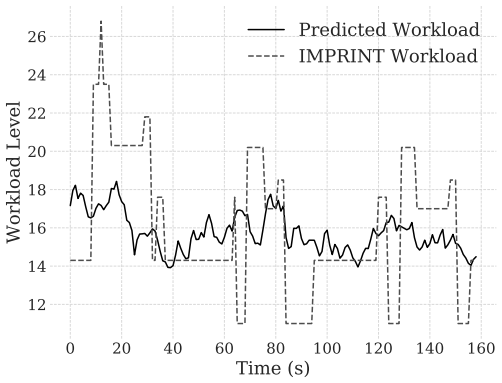
A major challenge associated with using EEG signals for prediction of workload is overcoming the non-stationarity of EEG data [77]. Previous work showed that using temporal context to inform mental workload prediction resulted in better classification performance than methods which did not consider prior mental states in a setting where no workload transitions were observed during the course of a trial [76, 78]. Here,



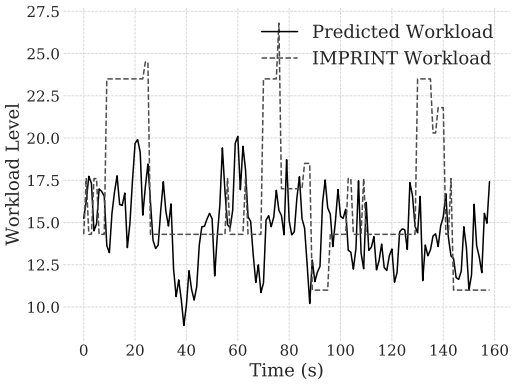
(a) Bidirectional LSTM: Participant 5, Trial 9
Corr: 0.40, RMSE: 3.20, Naive: 3.47, Ratio: 0.93



(b) Random Forest: Participant 3, Trial 2
Corr: 0.26, RMSE: 4.89, Naive: 5.01, Ratio: 0.97



(c) Siamese-triplet: Participant 5, Trial 9
Corr: 0.32, RMSE: 3.30, Naive: 3.47, Ratio: 0.95



(d) ANN: Participant 5, Trial 13
Corr: 0.24, RMSE: 4.54, Naive: 4.13, Ratio: 1.10

Figure 17. A comparison of approximate median performance for both RMSE and correlation for each algorithm. Performance values are reported in the following form: Correlation for trial; Algorithm-specific RMSE for trial $RMSE$; Naive RMSE for trial $RMSE_{Naive}$; RMSE reduction ratio compared to baseline ($RMSE/RMSE_{Naive}$).

this work is extended by showing that algorithms which account for temporal context provide improved predictions in an environment with many workload transitions. This indicates that bidirectional LSTMs do not merely act as an averaging mechanism with regard to brain activity over time.

To provide further evidence for this result, temporal feature salience is examined for bidirectional LSTM models using a simple extension to feature saliency methods

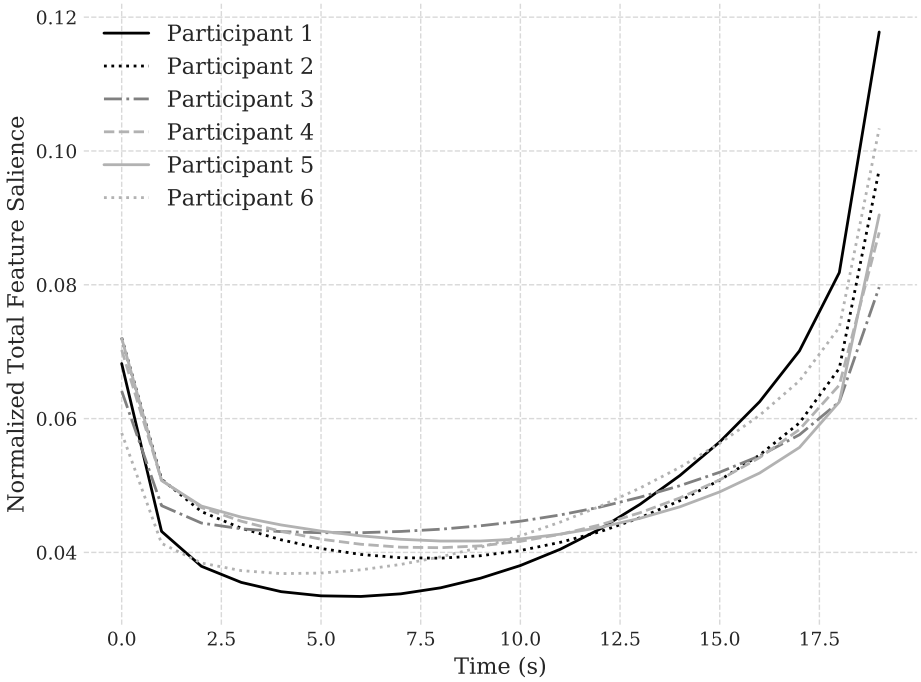


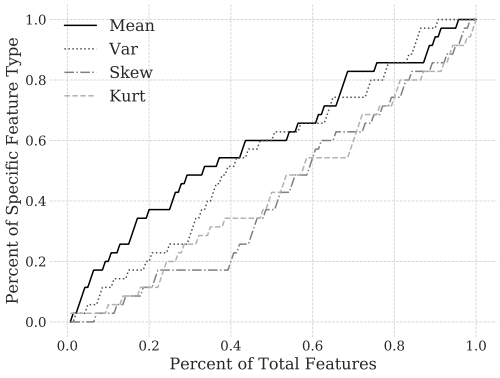
Figure 18. The effects of bi-directionality on feature salience is evident in this plot of by-participant normalized temporal feature salience. The most salient features are found at the beginning and end of the 20 second sequence due to the short backpropagation path present in both temporal locations.

introduced by Ruck, et al. [132]. Ruck et al. [132] derived a method for understanding feature salience in neural networks by computing the derivative of the outputs with respect to the inputs of the training set. For a LSTM, this can still be done, but the inputs have temporal dependence. Because of this dependence, saliency should be examined in multiple ways. The first way is through a plot of temporal salience of all the features. Figure 18 displays by-participant normalized temporal feature salience as a function of average feature salience at any given point in the 20 second sequence length across all training observations. The plot shows that the most salient times in a given sequence are associated with the shortest backpropagation paths for a bidirectional LSTM: the beginning and end of the sequence. This suggests

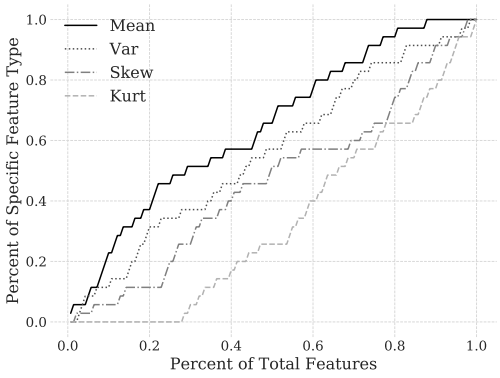
that algorithmic assumptions play a significant role in the learned features in a deep LSTM network. This realization also guides future research direction which should examine the effect of using neural attention mechanisms coupled with the feature saliency techniques presented here. Differentiable neural attention mechanisms can learn what portions of a sequence are most important to predictive performance and focus on these salient subsequences to improve results. This may result in a more nuanced understanding of what features are important and enable better modeling of brain activity and workload transitions.

Another interesting result of previous work was that skewness and kurtosis of PSD distributions were not statistically significant features in multi-task environments without workload transitions despite their inclusion in the best-performing models [78]. Feature saliency was not examined on an individual basis in Hefron, et al. [78], even though models were created for each individual independently of the others. Because of this design, if a particular participant had a majority of highly-salient features associated with higher-order statistical moments, this would be overshadowed by the remaining participants' results. Additionally, it was postulated that skewness and kurtosis may be important features in an environment with workload transitions. Here, feature saliency is examined on an individual-basis in an environment with numerous workload transitions.

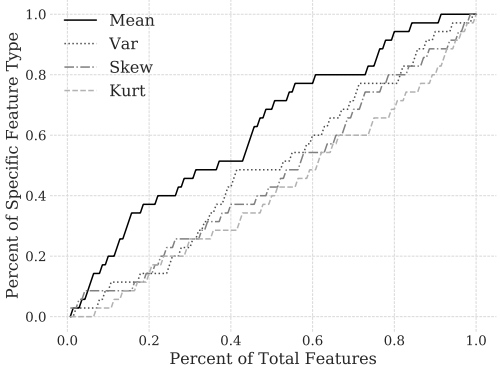
Figure 19 illustrates the results of a by-participant feature salience analysis of the bidirectional LSTM. All 140 features were ordered based on feature salience values determined by taking the partial derivate of the model output with respect to the input features across all trials and timesteps for a given participant. There were 35 features of each type (mean, variance, skewness, and kurtosis). Using forward stepwise selection, these plots illustrate the percentage of features in the salient set from each type (mean, var, skew, kurt) as the next most relevant feature is added starting



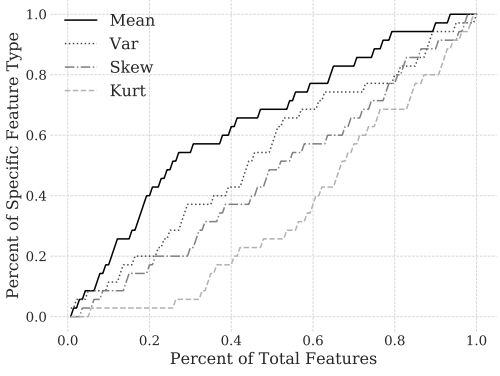
(a) Participant 1



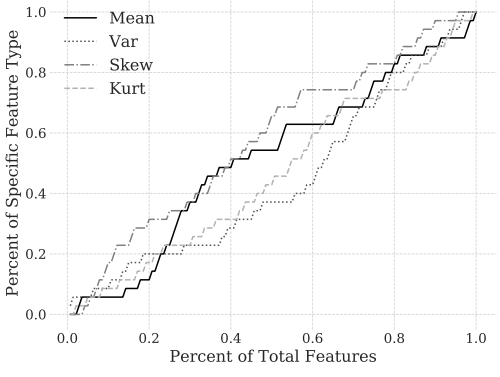
(b) Participant 2



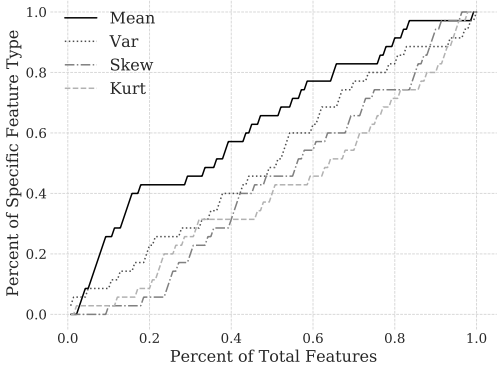
(c) Participant 3



(d) Participant 4



(e) Participant 5



(f) Participant 6

Figure 19. These plots illustrate the percentage of features for a given feature type that have been added to the salient set compared to the total percentage of features processed using forward step-wise selection. The upper right corner of the plot indicates that all features are included, while the bottom left corner indicates no features are included. Curves closest to the upper-left portion of the plot indicate the most salient feature-type.

Table 28. Top 20 most salient average features across participants and timesteps for bidirectional LSTM models.

Rank	Feature
1	Pz_gamma_mean
2	Fz_gamma_mean
3	T3_gamma_var
4	O2_gamma_mean
5	O2_beta_mean
6	Pz_alpha_var
7	T3_gamma_mean
8	F7_alpha_var
9	Pz_beta_mean
10	T4_alpha_mean
11	F7_gamma_var
12	O2_gamma_var
13	T4_theta_mean
14	Fz_alpha_mean
15	O2_gamma_skew
16	Pz_theta_mean
17	T4_delta_mean
18	T4_gamma_skew
19	T4_gamma_mean
20	Pz_alpha_skew

from an empty set. For example, in Figure 19d, 40% of mean features (14 of 35 mean features) had been added to the salient set after processing 20% of the overall features (28 of the 140 total features) with greedy selection. Mean features tended to dominate for all participants with the exception of participant 5, where skewness features were the most important.

Finally, average feature salience across participants was examined by electrode location and frequency band to determine trends. Results using bidirectional LSTM models are tabulated in Table 28 and show that gamma features are most salient. These results are in-line with a growing body of literature which indicates that gamma-band activity is highly salient in non-stimulus locked environments [27, 77, 101, 148, 185]. However, it should be noted that the increased gamma salience may in-part be attributed to increased motor movement which could plausibly be associated with the IMPRINT-derived physical workload components. Future work can explore the impact of this analytical limitation.

6.7 Conclusion and Future Work

In this study, several regression approaches to psychophysiological workload estimation were evaluated. Bidirectional LSTMs and Siamese-triplet networks showed statistically significant improvements in prediction compared to non-temporally aware models in a non-stimulus-locked, realistic simulation environment. This work marks the first usage of a Siamese-triplet network for EEG analysis. Initial results were statistically and qualitatively comparable to best-in-class bidirectional LSTM results. A theoretical method for setting a variable margin for the Siamese-triplet network was also presented.

Future work should empirically compare this technique with other margin-setting methods. Several feature visualization techniques were introduced aimed at understanding EEG feature saliency as functions of time and feature type. These techniques led to the realization that bidirectional LSTMs are likely biasing results based on algorithmic assumptions. Differentiable neural attention mechanisms are suggested to eliminate this dependency in future work. This may result in clearer insights into temporal feature salience enabling improved brain activity and workload transition modeling.

VII. Conclusion and Future Work

This dissertation has focused on improving operator workload estimation in multi-task environments using novel deep learning and signal processing techniques with a target use case of operator functional state assessment in an in-flight environment. In the following section, the major themes of this document will be reviewed through contributions discussed by chapter. Chapters I and II surveyed the problem space and examined two major challenges associated with analysis of electroencephalograph (EEG) data: day-to-day variability, and cross-participant differences. Day-to-day variability is caused by temporal non-stationarity of the feature-to-target mapping and results in single models that are trained across several days to perform worse than models trained and tested within the same day. Cross-participant differences refers to the way individual differences manifest in EEG signals causing variation in the feature-to-target mappings from one individual to another. Then, deep learning was introduced and its application to EEG analysis was explored. While deep learning techniques had been widely applied across a diverse set of EEG and non-EEG use cases, research had not been conducted using deep learning to assess operator workload in multi-task environments; nor had these techniques been used to mitigate the effects of day-to-day variability and individual differences.

7.1 Contributions and Findings

The contributions in this dissertation represent the first application of novel feature generation techniques and deep learning approaches to non-stimulus aligned, multi-task environments. To provide context for the contributions, they are summarized here by chapter. Chapter III began by discussing a new methodology for generating time-frequency EEG features in the context of cross-day non-stationarity.

Multi-day data collected from a Multi-Attribute Task Battery (MATB) workload study was used to evaluate a new feature generation methodology which examined not just the average power, but also the variability of the power distribution in the clinical frequency bands over a 10 second sliding temporal window. High versus low workload levels were predicted for day five of the study based on training three traditional classifiers–Linear Discriminant Analysis (LDA), random forest, and K-Nearest Neighbors (KNN)–on the first four days' results. This work yielded the following contributions and findings:

A1: A new methodology for performing time-frequency EEG analysis used variance of distributions produced from sliding windows of Power Spectral Density (PSD). Including variance as a feature enabled a statistically significant ($p < .0001$) cross-day workload classification accuracy improvement of 5.8% above models using only mean power [77].

A2: Feature saliency analysis of a cross-day workload experiment found temporal gamma oscillations to be the most salient feature across participants, but noted significant variation in feature saliency across individuals [77]. These results added to existing literature presented by Laine, et al. [101] and Bowers, et al. [27].

Chapter IV extended the analysis from Chapter III by systematically examining feature utility and by addressing temporal stationarity challenges associated with EEG data. Several deep Recurrent Neural Network (RNN) models including Long Short-Term Memory (LSTM) architectures, a feedforward Artificial Neural Network (ANN), and Support Vector Machine (SVM) models were trained on data from six participants who each performed several MATB sessions on five separate days spread out over a month-long period. Each participant-specific classifier was trained on

the first four days of data and tested using the fifth's. The overall performance of this approach was accurate enough to warrant evaluation for inclusion in operational systems when sufficient data is available. Specific contributions and findings include:

B1: Using deeply recurrent neural networks to account for temporal dependence in EEG-based workload estimation considerably improved day-to-day feature stationarity resulting in significantly higher accuracy ($p < .0001$) than classifiers which did not consider the temporal dependence encoded within the EEG time-series signal [78]. Average classification accuracy of 93.0% was achieved using a deep LSTM architecture [78]. These results represent a 59% decrease in error compared to the best previously published results for this dataset and a 58% reduction in classification error over baseline methods [78].

B2: The significance of new features: all combinations of mean, variance, skewness, and kurtosis of EEG frequency-domain power distributions were evaluated on a dataset without workload transitions. Mean and variance were statistically significant features, while skewness and kurtosis were not [78].

Chapter V transitioned from within-participant analysis to cross-participant state estimation and used a novel convolutional-recurrent architecture to mitigate performance losses compared to within-participant results for a non-stimulus-locked task environment. A zero-data training technique was used, meaning that a trained model made workload estimates on a new participant who was not represented in the training set. Using data from a separate MATB experiment, a variety of deep neural network models were evaluated in the trade-space of computational efficiency, model accuracy, variance and temporal specificity yielding three important contributions or findings:

C1: The performance of ensembles of individually-trained models is statistically

indistinguishable from group-trained methods at most sequence lengths [76]. These ensembles can be trained for a fraction of the computational cost compared to group-trained methods and enable simpler model updates [76].

C2: While increasing temporal sequence length improves mean accuracy, it is not sufficient to overcome distributional dissimilarities between individuals' EEG data, as it results in statistically significant increases in cross-participant variance [76].

C3: Compared to all other networks evaluated, a novel convolutional-recurrent model using multi-path subnetworks and bi-directional, residual recurrent layers resulted in statistically significant increases in predictive accuracy and decreases in cross-participant variance [76].

The application of deep learning techniques to assess operator functional state has been limited to laboratory based experiments with individually-tuned predefined levels of task difficulty designed to elicit a difference in performance for a given individual. The work in Chapter VI departed from these traditional limited experiments in two ways. First, an operationally realistic Remotely Piloted Aircraft (RPA) Intelligence, Surveillance, Reconnaissance (ISR) simulation environment was used that incorporated multiple tasks and numerous workload transitions. Second, rather than using subjective workload ratings to determine machine learning target values, objective analytical measures were used to increase temporal resolution of task load and to derive target values using a discrete event simulation [24, 135]. With the increased temporal resolution and multiple levels of workload based on combinations of tasks, several deep learning regression-based approaches were developed and evaluated against baselines to better model the continuously interactive environment. The approaches represent the first attempt to use regression-based deep learning for EEG analysis. Develop-

ment of these models and analysis of their results led to the following contributions and findings:

D1: Deep learning methods which incorporate temporal context significantly outperform methods which do not consider temporal context in terms of Root Mean Squared Error (RMSE) (all $p <= 0.0282$) and correlation (all $p < 0.0056$) in a multi-task, realistic simulation environment with many workload transitions. This is an extension from previous work which demonstrated similar performance in a complex, multi-task environment without workload transitions [78].

D2: Siamese-triplet networks are introduced for EEG analysis and perform as well as best-in-class bidirectional LSTMs for workload estimation in terms of RMSE and correlation metrics.

D3: By analyzing margin considerations, a theoretical argument for using a variable margin for Siamese-triplet networks applied in regression settings is presented and a mathematical formulation is developed which can reduce the need for a hyperparameter search.

D4: A feature saliency visualization technique is introduced enabling rapid discrimination of feature saliency for neural networks with grouped feature types.

D5: Skewness and kurtosis of PSD distributions are salient features in multi-task environments with workload transitions for some participants. Based on previous results, inclusion of these features does not have a negative effect on model performance [78].

D6: An extension of a feature saliency technique [132] is used to examine bidirectional LSTMs and indicates that temporal feature saliency of EEG signals is tied to the depth of the path associated with backpropagation when using

LSTMs. Differentiable neural attention mechanisms are suggested to eliminate this dependency in future work.

7.2 The Way Ahead

Several paths for future work were discussed at the end of each chapter. The follow-on work from those studies will be briefly summarized, and then three other potential campaigns of research will be introduced. These paths for future exploration should improve results and enable a better understanding of the underlying brain activity as related to workload. These concepts for future work will expand the utility of deep learning for EEG analysis and examine the effectiveness of multi-modal experimental techniques for the flight environment.

7.2.1 Follow-on Work from Studies A-D.

To further improve cross-participant applicability and day-to-day variability, a study employing a large number of participants ($n > 50$) should be performed to understand how well deep networks can integrate and generalize psychophysiological data on a greater scale. It is expected that with greater diversity in the data set due to individual differences, the performance gap between within-participant and cross-participant modeling will further reduce–and with enough individuals available, group models may outperform individual models. Additionally, it would be interesting to quantitatively evaluate any interaction effects of using cross-participant data in the presence of day-to-day variability. In each study in this dissertation, the effects of cross-participant differences and day-to-day variability were isolated so interaction effects were not investigated.

To further improve cross-participant applicability, unsupervised clustering of source localization across subjects using Independent Component Analysis (ICA) could also

be attempted. This type of preprocessing may better align the input feature space with the assumptions of Convolutional Neural Networks (CNNs) and also may improve recurrent network performance by pairing signaling from similar brain regions across subjects for each input feature. Methods to weight ensemble models based on similarity of the input features to the feature distributions of each model in the ensemble could also be developed. This could allow for better cross-participant generalization by using data only from individuals with similar brain activity for a given task.

This dissertation marked the first usage of a Siamese-triplet network for EEG analysis. A theoretical method for setting a variable margin for the Siamese-triplet network was introduced. Future work should empirically compare this technique with other margin-setting methods. Several feature visualization techniques were presented aimed at understanding EEG feature saliency as functions of time and feature type. These techniques led to the realization that bidirectional LSTMs are likely biasing results based on algorithmic assumptions. Differentiable neural attention mechanisms are suggested to eliminate this dependency in future work. This may result in clearer insights into temporal feature salience enabling improved brain activity and workload transition modeling.

7.2.2 Data Augmentation for EEG.

In Chapter II, it was shown that only Gaussian noise injection, interpolation methods, and window-slicing were examined for EEG data augmentation. Due to the nature of brain activity, it is difficult to perform physically plausible data augmentation. This is largely because a person cannot simply look at the data and determine that a valid transformation was performed, as one can in the image recognition domain. Furthermore, the temporal dynamics and causes of non-stationarity

are difficult to characterize. Physically-plausible data augmentation requires creation of a generative process which can model realistic perturbations of the latent process and produce physically-plausible observations. This type of augmentation works in data-space and requires a more nuanced problem understanding to produce observations than interpolation-based techniques.

One area to avoid is using ICA to separate out latent processes in brain activity and then back-projecting a subset of these into the original feature space. This was attempted and was found to be unsuccessful. In particular, removing individual components and then back-projecting all remaining components did not have a significant effect on workload classification accuracy. Additionally, isolating artifactual signals and then mixing them across trials or individuals had a significantly negative effect on model performance. The effect was more pronounced when these artifactual signals were mixed across workload conditions–high workload ocular artifacts were injected into a low workload trial and vice versa.

Instead of using an ICA approach, recent deep learning research into generative modeling may enable successful methods to be developed for generative augmentation of EEG data. Generative adversarial networks and variational autoencoders present promising paths which could drastically expand existing datasets and improve results without gathering more data.

7.2.3 Transfer Learning and Cross-task Applicability of EEG Signals.

Cross-task generalization is a major challenge for EEG-based models. To date, no cross-task transfer learning EEG study using deep learning techniques has been conducted despite successes in other application domains such as image processing [51, 124, 182]. Deep neural networks build up a hierarchical feature representation with a gradation from general features in the lower levels of the network to more

specific features in the higher levels. Conversely, traditional classifiers use shallow features and do not have levels of feature representations that span from general to specific. Because of these differences in feature types, there is an important distinction between performing transfer learning using traditional machine learning techniques and deep networks. During transfer learning scenarios using traditional classifiers, feature pruning is used to find the best set of features that work in both task environments. On the other hand in a deep neural network, the same set of input features is used across tasks, but varying degrees of fine tuning are performed on the different layers depending on how different the base domain is compared to the transfer domain.

A study focusing on transfer learning using deep neural network models trained with data from one task-domain to make predictions in a different task domain would be useful for several reasons. First, it could yield improved results by allowing transfer of some of the learned representation from one domain to another. Second, it may be possible to acquire a larger amount of relatively low-cost lab data using a surrogate task to tune a high-fidelity model used to make predictions in a high-cost, or high-risk operational task. Finally, a method could be created which characterizes the similarity of brain activity in one task compared to another based on feature similarity in a deep neural network. It is suggested to start this work using single-task environments. Currently no method exists to quantify similarity between two tasks in EEG feature space. Developing one could lead to both a better understanding of how the brain responds to different tasks, and what kinds of tasks would be compatible for transfer learning in the future. To jump-start this effort, a short literature review on transfer learning and cross-task utility is provided in Appendix D.

7.2.4 High-fidelity Flight Simulator Workload Estimation.

After addressing many of the outstanding challenges associated with Operator Functional State Assessment (OFSA) using EEG signals in a laboratory setting, it is desirable to determine how well the models translate to more operationally realistic flight environments since no deep learning studies have been conducted in this area. Evaluating the degree of model improvement that can be attained by combining multiple psychophysiological and behavioral data streams when highly-skilled operators perform complex operational tasks is also needed to improve workload estimation. This can only be accomplished in a flight simulator or in actual flight conditions. Streams of psychophysiological and behavioral data to include EEG, electrocardiogram (ECG), Galvanic Skin Response (GSR), respiration, pilot control input (stick/yoke, rudder pedals, and throttle movement), auditory input and output, and eye tracking data features should be examined using deep learning approaches. Feature salience associated with these deep learning models should be incorporated in this analysis to understand the utility of each different data stream.

Deep learning approaches have yielded improvements on the underlying challenges associated with day-to-day variability, and cross-participant applicability. After understanding how well these approaches generalize across tasks, determining the utility of deep-learning-infused workload assessment tools in high-fidelity simulation environments will be necessary. This will be a vital step towards attaining the level of accuracy required to reap the benefits associated with accurate workload assessment and operator augmentation. It is hoped these techniques will lead to improved training and better wingman utilization in the military flight environment, objective methods for estimating workload during developmental flight test, and ultimately adaptive augmentation that reduces aircraft accident likelihood and improves operator performance.

Appendix A. A Multi-Faceted Approach to Operator Workload Estimation

Psychophysiological measures are clearly high correlates to mental workload levels. Borghini, et al. summarized several neurophysiological measures for assessing operator functional state assessment and the expected response as workload increases [26]. They identified EEG measures, blink rate and duration using EOG, and heart rate as well-tested measures with known responses to fluctuations in operator workload. However, psychophysiological measures do not tell the whole workload story. In this section, the reasons a multi-faceted approach to workload estimation which combines psychophysiological, behavioral, and task requirement expectations should provide the greatest probability of success in operational environments is discussed, with the caveat that first laboratory work needs to be accomplished to improve the probability of success of such an effort.

Gartner and Murphy provided a comprehensive summary of all workload indicators available at the time of their study in 1976 [59]. Surprisingly, the list is still relatively comprehensive today and few updates would be required; however, analysis techniques and sensors have evolved significantly since the time of their study. Gartner and Murphy categorized workload sensing methods based on what types of measures were utilized. The categories included analyzing the task requirements, evaluating task performance, behavioral measures, psychophysiological measures, and finally subjective methods [59]. Some measures were grouped as task performance that will be referred to as behavioral measures in the remainder of this paper due to the way we define operator input-output. Pilot input-output can be separated into three distinct categories largely corresponding to the factors identified by Cohen [41]: motor movement to include control surface movement (stick, rudder pedal, throttle movement) and switch actuation, optical scanning of internal instrumentation and

external visual inputs, and auditory stimuli and required responses.

Evaluating the spectrum of categories described by Gartner, two are difficult to use in an unobtrusive way in the flight environment: task performance, and subjective ratings. Task performance is difficult to measure in a continuous manner and is also not necessarily correlated with operator workload. Cooper and Harper note that it is possible for the operator to obtain very good performance for a wide range of workload conditions including those where little-to-no excess capacity for any other task exists [45]. Furthermore, cognitive effort is not assessed by performance measures [59]. Subjective assessments are often used as truth data in most experiments involving workload. However, their drawback is that they can interfere with the tasks being performed and they are subject to individual rating biases. By eliminating post hoc subjective ratings and performance metrics from an operator functional state assessment system, justification for use of the three remaining sources is needed. Psychophysiological justification will be provided in Chapter II, so further justification is provided for the use of behavioral measurements, task requirements analysis, and how the use of all three methods could provide the clearest picture for operator workload.

While much research has been performed using a single given category, comparatively little research has been conducted combining categories to obtain a holistic assessment. Gartner and Murphy concluded that workload measures were selected primarily due to an investigator's theoretical interest in that measure or due to ease of measure rather than because of their expected overall utility towards measuring workload [59]. They further explain that performing workload analysis in an operationally representative environment is essential to avoid serious inadequacies in results, yet was not prevalent in much of the research at the time of their study [59]. Unfortunately, this remains a shortfall in many of the operator functional state exper-

iments conducted in recent years. Gartner and Murphy continue by stating that the synthesis of multiple workload sensing techniques across the spectrum of categories may offer the most promise towards achieving reliable workload estimates [59].

In the 1993 technical information manual on EEG use as a workload assessment tool for in-flight testing from the Air Force Flight Test Center, Hunn identified workload as a, "Concept which involves primarily mental effort, but also the associated factors of physical actions and environmental variables...[In the flight] environment, there is no simple division of mental versus physical tasks, and consequently, the effort of measuring workload will take a global approach [85]." This is indicative of the interaction effects between physical and mental workload in-flight and lend further credence to a holistic approach.

Parasuraman affirmed this theme when he reported that psychophysiological measures in conjunction with behavioral measures may provide more information than either alone [125]. Again, evidence of this notion is corroborated by Cohen in his study of projected workload for 225 different scenarios for an advanced fighter aircraft in light of multiple resource theory. Cohen found that nearly all pilot variation of estimated workload for 225 different scenarios could be accounted for by three factors: visuo-spatial demands, verbal communications load, and both manual and speech output demands [41].

Further evidence of the utility of behavioral measures is noted by Cooper and Harper when they note that the integral of control displacement, control force, has shown high correlation with subjective workload ratings [45]. They surmise that mental workload must not have been a factor in these circumstances [45]. Whether or not mental workload was a factor, the combination of measures from both aspects of the operator's functional state–physical and mental–should be complementary. While there is a continuum of physical and mental workload experienced by an operator,

overall workload can be dominated by high mental workload, high physical workload, or a combination of both. Another physical factor is how aggressively the pilot is flying. A pilot in a 6g turn will undeniably experience a higher workload than a pilot in a 30 degree bank turn. This also hints at the physical aspect of workload. The more dynamic a flight is with regard to an acceleration profile, the higher the workload rating in general. Schnell and Melzer found similar results to Cooper and Harper when they remarked that flight control input spectral analysis exhibited correlation with pilot workload during their in-flight workload testing in an L-29 [141]. They also noted that tracking of aircraft state is important in order to identify phases of flight and tasks to further assist with workload state identification [141].

The coupling of these features in a real-time workload model is one that has been ill-explored to date and warrants further investigation. Appendix B documents high-fidelity simulator and in-flight workload modeling efforts. While a multi-faceted approach to operator workload in the flight environment is likely required to achieve the desired levels of accuracy for operational utility, there are underlying algorithmic obstacles associated with operator state assessment using psychophysiological components of the system which must first be addressed in the laboratory environment. The greatest barriers to implementation of a system capable of objectively measuring operator workload are associated with the measured data and the algorithms used for assessment. This work focuses largely on the most challenging, yet arguably highest potential psychophysiological signal source–brain activity, as measured by the electroencephalograph (EEG). For EEG in particular, these analytical challenges require attention prior to building a multi-faceted system that performs satisfactorily in an operational environment. This effort begins by developing and applying analytical techniques that do not make typical independent, identically distributed assumptions present in many traditional machine learning techniques, but rather uses a deep

computational graph approach which accounts for the dependencies present in the data and investigates their performance in a controlled laboratory environment. After proving utility in the laboratory, this approach could lead to many benefits once applied to the military flight environment.

Appendix B. Overview of In-flight and High Fidelity Simulator Research

While non-stationarity is one of the major obstacles that must be overcome to implement an in-flight workload sensing system, a significant body of research using high fidelity simulators and in-flight testing has been performed that did not address non-stationarity. Generally, these studies are marked by experienced operators and more realistic, complex-task scenarios than laboratory-based research. The purpose of this section is to examine these high-fidelity simulator and in-flight psychophysiological workload estimation studies to show that no research has been conducted evaluating multi-modal/multi-faceted models or deep learning based models in an operationally-realistic environment. We also document lessons learned which may be of use in future experimental designs such as equipment setup.

An early in-flight recording of EEG occurred in 1987 when Sterman, Mann, and Kaiser conducted laboratory and in-flight testing to evaluate the effects of G-forces on pilot's brain activity using EEG [152]. The in-flight setup required the pilots to wear special vests to house recording equipment as well as use a custom helmet liner that contained EEG electrodes [152]. These devices must not impact ejection seat requirements. In-flight tasks included air-to-air and air-to-ground target acquisition, instrument procedures, and all other tasks required for a typical flight. Data was collected on four pilots during the study. Sterman, et al. determined that spatial disorientation is clearly detectable by EEG. They also found unique patterns of brain activity during competent task performance under G-loading in a T-38. Furthermore, they determined that these unique patterns normalized as performance degraded in an elevated-G environment. At frequencies below 8 Hz, total power was increased when operating under elevated-G conditions [152]. These findings indicate that flight dynamics have an effect on operator functional state assessment and must be ac-

counted for in an accurate workload model. It would be wise to capture aircraft G-loading as a time-synchronized feature. Furthermore, this highlights the need for in-flight validations of laboratory and simulator-derived models since the combination of flight dynamics and task realism cannot be adequately replicated anywhere else.

Wilson, Fullenkamp, and Davis conducted the first in-flight study examining evoked potentials in EEG recordings in 1990 while pilots performed air-to-ground training missions in an F-4 [174]. They pointed out that several shortfalls of operator workload subjective measures include the inability to continuously measure workload, interference with task performance, and operator bias [174]. Four electrodes were used to gather data and test points with muscle or gross movement artifacts were rejected. They found that it is possible to successfully record in-flight evoked potentials, but that muscle and gross movement artifacts pose challenges and only seven out of ten pilots had acceptable recordings [174].

Sterman, Mann, and Kaiser conducted in-flight handling qualities and cognitive workload testing in the Calspan NT-33 at USAF Test Pilot School while recording EEG data [151]. The experiment consisted of four 90 minute test flights in which air-to-air tracking tasks were performed with differing flight control laws to modulate task difficulty. The goal of the experiment was to identify an objective index of workload. The experimental setup was limited in that only 4 EEG channels could be recorded (F3, T4, P3, and P4). Eye open and eye closed baselines were performed both on the ground and in the air. Cooper-Harper ratings, performance measures, and cognitive workload ratings were assigned for each task. Post processing of the EEG data included FFTs, 2-second epoch spectral magnitude values were calculated, and ANOVA tests were performed [151]. Limitations of this study included limited EEG electrode sites and limited computational processing power for analysis. Consistent with recent studies, Sterman found decreased power in parietal alpha band to be

associated with higher workload conditions [151].

The in-flight physiological test program was completed in 1993 at Edwards AFB and was an early attempt at using EEG physiological measurements to gauge aircrew workload [85]. One important finding was that the use of event related potentials (ERPs) for in-flight testing suffers from external stimuli which can mask or simulate responses due to an inability to control the environment compared to the laboratory setting [85]. Hunn accentuates several other in-flight considerations including the utility of performing eye-tracking to identify what the pilot is looking at and the use of a camera system to capture movements related to aircraft control because they may induce muscle artifacts. He continued to explain that artifacts are a major concern in the cockpit environment and can be caused by muscle and eye movements as well as buildup of static electrical potentials or electromagnetic interference in the aircraft [85]. Hunn noted that these artifacts are identifiable and it is likely techniques for handling them can be devised. Two more conclusions Hunn draws are worth capturing for any future flight related operator functional state testing:

1. Simulator and flight tasks should be as similar as possible and the same aircrew should be utilized for both phases of a test.

2. Include reporting requirements that include interviews, subjective rating scale usage in-flight, and a written post-flight narrative [85].

Schnell, Keller, and Poolman developed an integrated hardware and software system, termed the Cognitive Avionics Tool Set (CATS), for capturing and processing psychophysiological data in-flight including EEG, ECG, respiration, pulse oximetry, galvanic skin response, and eye tracking. The device architecture is described in detail providing an excellent example of a fully-integrated real-time data acquisition system for psychophysiological measures [140]. Schnell and Melzer performed an

in-flight workload evaluation using the CATS system in an L-29 simulating a close-air-support (CAS), air-to-ground attack scenario [141]. Results from their test indicate that flight control input spectral analysis exhibits correlation with pilot workload and that tracking of aircraft state is important to be able to identify phases of flights and tasks [141]. The authors also noted that heart rate variability correlated with peak workload at approximately a 20 second lag [141]. Meaningful workload estimation results were not reported for the in-flight study.

Callan, et al. performed a single subject simulator and in-flight experiment to determine if a dry-electrode EEG system with wireless capability was suitable for in-flight applications [31]. They performed testing in a motion-based simulator and in an aerobatic, open cockpit biplane. A least-squares probabilistic classifier was used for workload classification. The motion-based simulator and in-flight testing both produced numerous head and body-movement artifacts which the authors said made classification considerably more challenging. While large artifacts were produced during dynamic portions of the flight profiles, performing preprocessing including Kalman filtering followed by ICA was found to significantly improve classification accuracy from 66.1% to 79.2%. Due to a significant amount of artifacts associated with head movement in-flight, it was suggested that future flight testing include a head-mounted accelerometer to try to isolate and remove head movement related artifacts. An additional note on the prevalence of artifacts is that results from a previous study indicated that dry EEG electrodes may be more sensitive to artifacts compared to electrodes which use gel [108]. Callan reported another finding relevant to future simulator studies: Using ICA and Kalman filtering did not significantly improve classification for the motion based simulator with a marginal increase in classification accuracy from 71.6% to 73.1%. It was postulated that the abrupt movements of the body and head due to the jerkiness of the simulator may have caused these methods to be ineffective

[31]. Interestingly, the order of operations for ICA and Kalman filtering was found to significantly effect the overall classification accuracy. It was suggested to perform ICA first followed by Kalman filtering because removal of the artifacts enabled the filtering to be more effective. Overall, the dry electrode based system was found to be suitable for in-flight research [31].

Astolfi, et. al performed a study which simultaneously recorded pilot and copilot EEG signals during various phases of flight in a high-fidelity aircraft simulator to estimate cognitive connectivity between the pilots [10]. They found the highest levels of cognitive synchronization during segments where shared tasks were performed such as during takeoff and landing. During these segments of high workload, they noted the pilot flying exhibited increased PSD in frontal theta bands with a decrease in parietal alpha bands similar to results found by other researchers [10].

Di Stasi, et al. conducted in-flight recordings of military helicopter pilots during various phases of flight with the objective of correlating modulation of EEG power spectrum characteristics with high and low operator cognitive workload [50]. This is one of very few recent publications where an in-flight workload evaluation was conducted using EEG. Eight Spanish Air Force rotary wing pilots participated in the study. Di Stasi, et al. used three separate questionnaires for each pilot prior to flight: the Stanford Sleepiness Scale, the Borg Rating of Perceived Exertion, and the Morningness-Eveningness Questionnaire to control for sleepiness, fatigue, and time-of-day alertness predisposition respectively. Flights were divided into four stages: takeoff, two air-work segments including slow-flight, stall series, and constant-rate turns, and a landing phase. Band-pass filtered measurements between .3 and 30Hz were captured at a 256Hz sampling rate. Four electrodes were analyzed–F3, F4, C3, and C4. The authors state that this combination of electrodes is optimal to avoid errors due to vibration, EMI, and operator movement [50]. VEOG and HEOG

were used in combination with expert opinion to detect and correct eye artifacts [50]. The four stages of flight were treated as trials and 4 second windows were created. EEGLAB was used to perform an FFT on the data and average power in the delta, theta, alpha, and beta bands were used as features [50]. Of all possible combinations of flight stage, frequency band, and channel, only the combination of flight stage and frequency band exhibited significant differences at the $\alpha = .05$ level of significance using within-subject ANOVA hypothesis tests [50]. Pilots exhibited increased power in the alpha band during takeoff and landing phases of flight compared to the air-work phases [50]. No significant difference was found in theta band power, but beta band exhibited statistically significant increased power during the takeoff phase [50]. These results are in stark contrast to Astolfi and other researchers who reported alpha suppression during demanding tasks and significant theta PSD increases. The source of this divergence is unknown. Di Stasi, et al. postulate that the increased perceived risk level associated with takeoff and landing may explain the increases in power observed during these phases of flight since increases in power are associated with arousal level [50]. An additional finding was that in general EEG power decreased as the flight progressed with the exception of landing. This was attributed to the operators becoming more comfortable with the tasks [50].

The reader may notice that none of the non-stationarity challenges associated with EEG analysis have been explicitly addressed during high-fidelity simulator or in-flight research. However, by not examining these challenges outside of a laboratory environment, many important aspects applicable to workload are left out of the equation. This is likely due to the expensive nature of flight test. Most of the time, very low-density EEG recordings were conducted. Only one of the in-flight or simulator studies performed workload classification, while all others merely examined trends or performed statistical testing. After addressing challenges associated with

non-stationarity in the laboratory setting, it is imperative that the techniques described in Chapters III - VI be applied to a more realistic environment to learn where shortfalls exist and to make progress on multi-sensor fusion approaches.

Appendix C. Cross-participant model architectures

In this section, depictions of each network architecture are provided with the tensor shapes at input and output of each layer. On the left side of the figures, are the layer types with important parameters about the layer annotated below. For *Dense* and *LSTM* layers, the number of hidden units/memory cells are indicated in below the layer type. *Conv2D* layers depict the shape of the kernel and the number of kernels as a 3-element tuple followed by a 2-element tuple describing the stride in each dimension. Pooling layers use the same notation with the exclusion of the number of kernels since there is no depth to these layers. *Dropout* layers depict the level of dropout used. As a note, all LSTM layers use 20% dropout on the input features which is not depicted in the diagrams. On the right hand side, input and output tensor shapes are shown based on 20 second input sequence lengths for illustrative purposes. Finally, the *None* dimension in each graph corresponds to the batch size and is depicted as *None* because it can be variably selected without affecting the architecture structure.

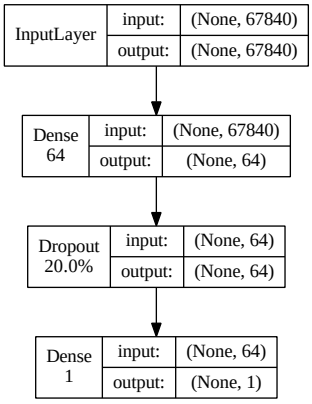


Figure 20. ANN architecture

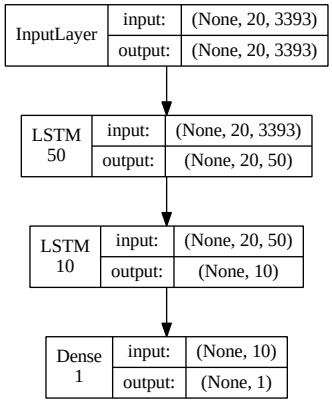


Figure 21. Two-layer LSTM architecture

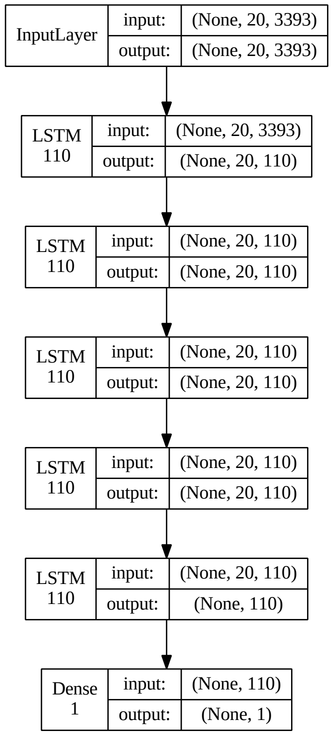


Figure 22. LSTM architecture

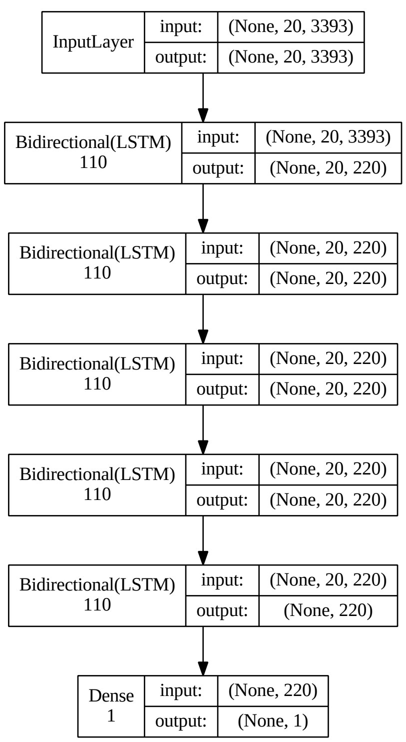


Figure 23. BDLSTM architecture

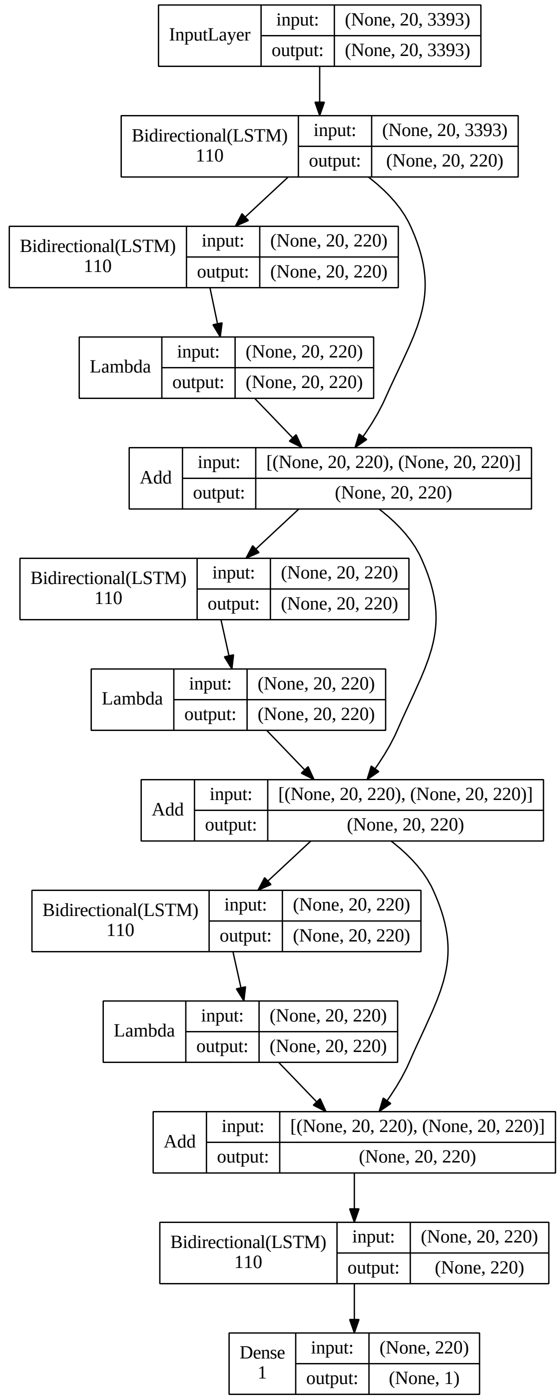


Figure 24. BDRLSTM architecture

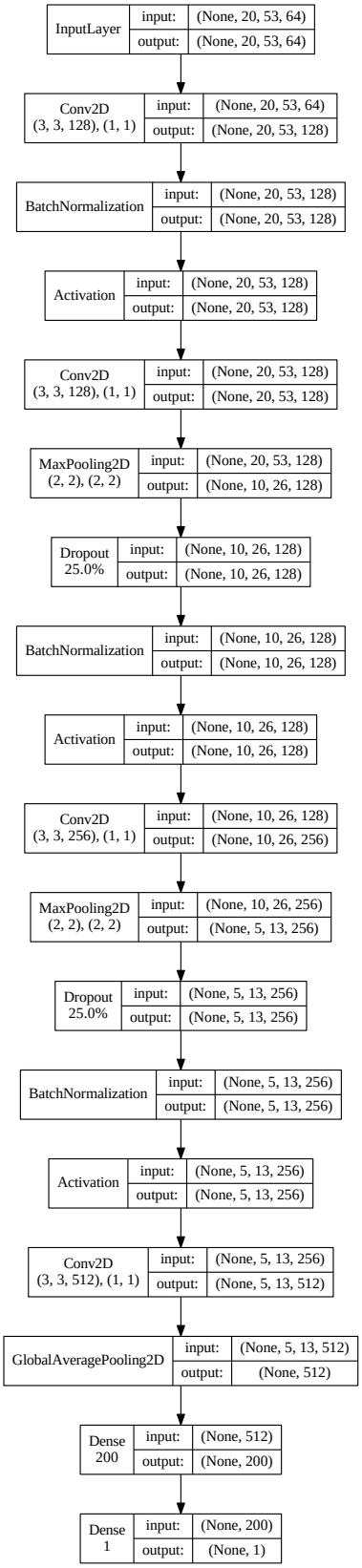


Figure 25. CNN architecture

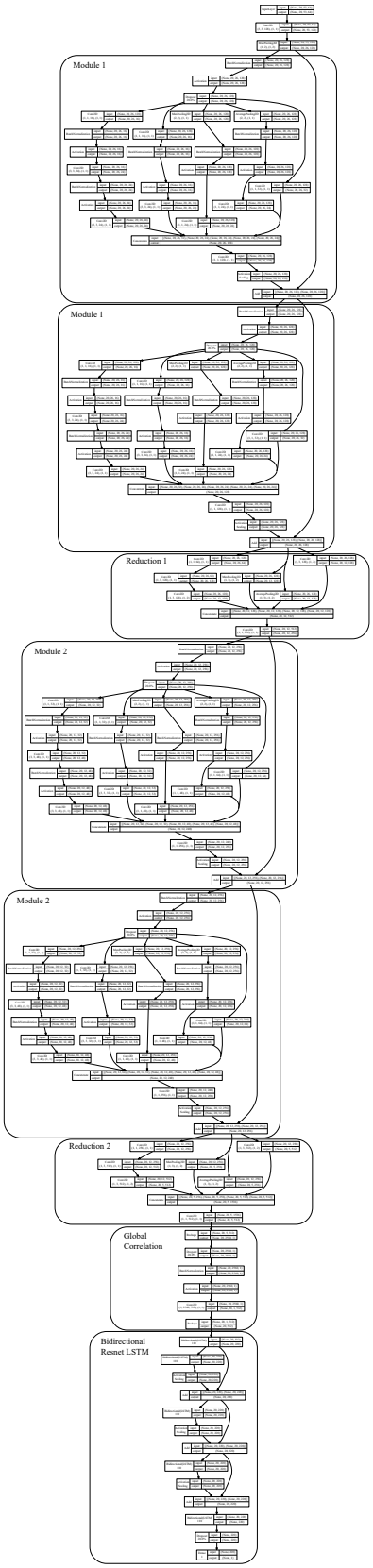


Figure 26. MPCRNN architecture

Appendix D. Transfer Learning and Cross-task Utility

Mental workload varies significantly depending on what task is being performed. Talking on a cell phone in rush hour traffic on the 405 in Los Angeles while children scream in the back seat of the car clearly represents a higher workload situation than the similar task of talking on a cell phone while lounging in a chair at home. The environment shapes the task. However, is it possible to take the environment into account to understand where a task falls on the workload continuum and to apply data from one task to another? Hart and Staveland in their 1988 report on the development of the NASA-TLX (Task Load Index) state, "the sources of workload are numerous and vary across tasks [73]." They continue by explaining that these sources of workload can be identified for a given task and that this information can be used to predict workload ratings [73]. That is the focus of this section–given a particular environment, is it possible to use data from other tasks to improve classification accuracy? Transfer learning is first discussed followed by an examination of studies that evaluate cross-task classification in a variety of ways specific to the EEG domain.

4.1 Transfer Learning

Transfer learning refers to the process of using a trained machine learning model from a source domain and applying it to a target domain with possible adaptation to the new domain [17]. The object is to exploit similarities between learned representations or features from similar tasks and or data [18]. Many different approaches have been applied to generalize models across domains using traditional machine learning methods such as SVMs or random forests, but representation learning has a distinct advantage over such approaches. Representation learning, in the form of deep neural networks, is particularly well-suited to address the transfer learning situation because the features learned in the lower-levels of the network tend to be more general, and

as depth increases, the higher-level features become compositions of the lower layer features which enable greater discrimination between classes and become increasingly specific to the task [20]. Conversely, transfer learning using a non-representation learning algorithm, such as a SVM, does not have a gradation from general features to specific features, but rather focuses on using a method to identify shallow features that are common across domains and pruning those that are domain specific [115]. Here we will examine relevant transfer learning literature from the deep learning community.

Traditionally, transfer learning techniques varied based on two main factors, the size of the target dataset and the number of parameters in the target network [182]. If the target dataset is small and the number of parameters in the overall network are large, then leaving the transfered layers frozen may be preferable compared to performing fine-tuning at all levels; otherwise, overfitting to the target distribution is likely [182]. Conversely, if the target dataset is large, or the number of parameters is small, then overfitting is unlikely and fine-tuning of all transfered layers may be appropriate [182]. It was also determined that even for distantly-related tasks, deep neural network performance can improve by employing transfer learning with fine-tuning at all layers [182]. Transfer learning acts more as an initialization technique which puts the network into a state that can be better optimized in these cases [182]. A final note regarding learned representations is that the same input features should be used, whenever possible, during transfer learning, even though different targets may be present between tasks [17].

Transfer learning for deep neural networks is typically implemented using the following steps [182, 51, 17, 115]:

1. Train a deep neural network using a base domain dataset.

2. Copy a number of layers over to a new network and add several adaptation

layers, or re-initialize the latter layers of the network.

3. Train the upper parts of the new network on a target domain that has a different task than the base domain by freezing the transfered layers that were not re-initialized. This step prevents the randomly initialized upper layers from negatively impacting the lower layers' learned representations while converging on an optimal solution.

4. Perform fine-tuning on all layers of the target domain network by unfreezing the lower layers to allow for better domain adaptation (provided there are sufficient labeled observations in the target domain).

It is possible to perform steps 3 and 4 simultaneously by using a higher learning rate for any layers whose weights are being trained from a newly initialized state. Long, et al. used a learning rate that was 10 times greater than fine-tuned layers if training a newly initialized layer [115].

Now we briefly discuss some lessons learned from the image processing domain since some of the most important transfer learning insights and results have been produced in this field. Oquab, et al. demonstrated that mid-level features in a convolutional neural network can be of great utility for transfer learning if there is enough similarity between base and target distributions [124]. They used Krizhevsky's AlexNet architecture and trained on a base domain of ImageNet images. Then, transfer learning was used to improve performance on other smaller datasets such as the Pascal VOC dataset [55] and the Caltech256 dataset [68]. After training the base network (5 convolutional layers followed by 3 fully-connected layers), they removed the final fully-connected layer and added in two adaptation fully-connected layers with random initializations. Their best results were obtained by freezing the five convolutional layers' weights, but training the remaining 4 fully-connected layers (2 original

and 2 newly added) [124]. Their results showed that even without adjusting the convolutional layers, on sufficiently similar task domains, the mid-level features are transferable to new domains and transfer learning can outperform the best networks trained only using the target domain data.

In similar work, Donahue, et al. evaluated the adaptability of features at various depths of Krizhevsky's AlexNet [51]. First, the base network was trained on the ImageNet database. Then, the weights were frozen and various depths were used (first 5, 6, or 7 layers) to perform forward passes to produce new features for a variety of target datasets. These new features were then used as input to either an SVM or a logistic regression classifier which were trained to learn the new task. Their results showed that across numerous target datasets, using learned features from the output of the first fully-connected layer after the convolutional layers in the base domain resulted in significantly better results than training only in the target domain with no transfer learning [51]. This demonstrated that mid-level feature transfer is useful between differing image recognition tasks and can be especially useful when the target domain has a limited amount of data.

Both Oquab and Donahue were interested in mid-level feature representations and did not examine the full spectrum of general to task specific features in transfer learning applications. Yosinski, et al. formally quantified this transition from general to specific features with increasing network depth and determined some best practices for transfer learning in the image-classification domain [182]. These best practices should apply in other contexts including cross-task EEG-based classification. Like Oquab and Donahue, Yosinski used AlexNet as a tool to study transfer learning. Task specificity versus generality was evaluated by examining transfer learning performance of networks trained with a varying number of layers frozen. Base and target datasets were created using randomly assigned non-overlapping ImageNet classes–half

the classes were assigned to the base domain while the other half were assigned to the target domain. Both datasets had a large number of examples, 645,000 images each. The base and target networks each had 8 layers corresponding to the original AlexNet design. Both networks were fully trained using their respective datasets. The target domain's trained network was used as a control. They then evaluated all combinations of possible transfer learning by freezing all successive layers from the base domain (freeze layer 1, freeze layers 1 and 2, freeze layers 1, 2 and 3...), and transferring these to the target domain with randomly initialized higher layers. They further evaluated whether keeping these layers frozen during training or performing fine tuning improved performance and whether the transfer learning network outperformed the control network for a given number of training iterations. Several very important findings resulted from this thorough evaluation [182]:

1. There are two types of performance degradation that occur during transfer learning situations: layer-wise co-adaptation occurs when two layers co-evolve during training and only one of the neighboring layers is transferred; and poor performance caused by too much feature specificity from the base domain. The former problem manifests itself in the middle layers of a transfered network, while the latter problem dominates performance when transferring too many layers without performing fine-tuning. The co-adaptation problem may be alleviated by dropout and batch normalization.

2. Performance gains over and above those from the control network were present whenever transfer learning was used with fine tuning and the two domains were similar. The best results were obtained by transferring all layers of the old network and fine-tuning all layers. This resulted in better performance than just training on the target dataset for the same number of iterations–a surprising result given the large number of examples in both datasets. This shows that

transfer learning is applicable beyond the typical use case where only a small target dataset is available.

3. By examining the decrease in accuracy from a base domain to a target domain as the number of layers of frozen weights increases, it is possible to clearly quantify how similar the two domains are. If the transfer learning performance remains high as more frozen layers are transfered, then the two domains are more similar. Conversely, if there is a marked decrease in performance when only the lower frozen layers are transfered, then the two domains are quite dissimilar and do not even share general feature representations.

4.2 Cross-task EEG Modeling

Currently, all cross-task modeling and transfer learning work in the EEG analysis domain has focused on using traditional algorithms and feature reduction approaches. It is one of our objectives to apply transfer learning with deep neural networks to this domain, but first we will review existing approaches.

Baldwin and Penaranda came the closest to using a representation learning method that could have benefited from a transfer learning technique. They set out to determine whether a single hidden-layer ANN using 20 hidden nodes and trained to classify workload level based on one task could be used to classify workload in another task for a novice operator performing different memory tasks [11]. This was essentially an exploratory transfer learning attempt as they did not use any data from the second domain to fine-tune the weights from the base domain. Furthermore, the lack of depth in the model meant that their neural network approach would not theoretically perform any better than other shallow classification methods. The memory tasks–reading span, Sternberg, and visuospatial n-back tasks–were performed by each of 15 participants. Each task had distinct easy and hard difficulty levels. Within-task clas-

sification resulted in 85-87% accuracies while cross-task classification averaged 44.8% accuracy which was slightly worse than a random guess. While all three of these tasks were memory tasks, they clearly evoked different psychophysiological responses in each participant. The researchers purposefully chose working memory tasks that were known to use different working memory processes and it is hypothesized that other more similar working memory tasks may have greater cross-task applicability [11]. The results from this study indicate that workload measured on tasks that may superficially appear related, may in truth require very different processing centers and produce unique workload signatures. However, a deep model may still find similarities at a more general level that could be exploited if a mapping from one domain to another could be learned.

Other literature has shown positive results regarding cross-task classification [61]. Gevins, et al. conducted a study which examined workload associated with working memory tasks using eight participants [61]. Two versions of a N-back matching task were performed by eight participants. The task can be thought of as a continuous slide show with transitions to the next slide every 4.5 seconds. A single letter was displayed on each slide in a particular location. When the next slide appeared, either the same letter or a different letter would show up in either the same location or a different location. The verbal task required the participant to identify if the same letter was present on two slides, regardless of the location of the letter. The second task was a spatial task which required the individual to determine if the location of the letter on a particular slide was the same as another slide. Difficulty was varied by changing how many previous slides the individual had to recall to determine if it was a match. Low difficulty was defined as matching with the previous slide. Moderate difficulty required recall from two slides back, while high difficulty was defined as the third slide in the past. EEG signals from 27 electrodes were recorded as well

as EOG. A single-hidden-layer feed-forward neural network was used to train and classify the data. Gevins, et al. examined cross-task generalization by training by-participant networks on a particular task and testing on the other task using hand-selected features and an iterative algorithm to further prune those features. They achieved an average of 94% classification accuracy for the two-class problem (high and low workload). These results are in stark contrast to those found by Baldwin and Penaranda, indicating that the two tasks in this experiment likely used very similar working memory processes compared to the other study. This illustrates that there is a continuum of task differentiation which must be considered when trying to determine if a particular model will be applicable to a different task and that it will affect the degree of domain adaptation which may be required.

In a separate study, Gevins examined the differences in signaling based on two working memory load tasks–spatial and verbal N-back tasks. While this was not a cross-task classification study, Gevins identified several trends in rhythmic activity between high and low workload tasks. Increases in frontal theta activity and reduction in slow parietocentral alpha amplitude were found to be strongly correlated with increased workload and appeared to be insensitive to task type while fast alpha signals were sensitive to spatial versus verbal tasks [62]. Furthermore, increased asymmetric attenuation of hemispheric alpha activity was observed during difficult spatial tasks compared to difficult verbal tasks [62]. These results reinforce the notion that signaling will generally be different based on task type, but that some signals may be task-invariant between particular tasks depending on the degree of similarity between the tasks.

Wilson and Russell performed an air traffic controller workload study which analyzed seven air traffic controllers' workload ratings from realistic air traffic control tasks of varying difficulty [177]. What set their experimental design apart from other

cross-task models discussed here is that their tasks were complex and operationally relevant. Four levels of workload were specified: low, medium, high, and overload. Two distinct tasks were used. Psychophysiological features were recorded and analyzed to classify workload using a classifier based on stepwise discriminant analysis and a separate ANN classifier. Both classifiers had classification accuracies that were approximately the same for each classification task. The authors used an operational setup in an ATC simulator that mimicked real life conditions. Participants provided NASA-TLX ratings which were used as target data during training and classification evaluation. A total of 85 EEG features were produced by taking the log power of the preceding 10 second interval. In addition to the EEG features, respiration rate, heart rate, and EOG log power were used as input to the classifiers. Two tasks were performed. The first task was a complexity task which varied difficulty based on the complexity of the air traffic presentation. The second task varied difficulty based on volume of traffic rather than complexity. While these tasks are related, different skills are required to handle each situation. Perhaps the most important finding from this study was that shallow classifiers based on a given complex task appear to perform very poorly when used to evaluate workload from another related, but still distinct task. In this case, Wilson and Russell evaluated workload for volume-based conditions using classifiers based on complexity and vice versa. The classification accuracies for these cross-task profiles averaged between 26% and 37% correct which was approximately equivalent to the random chance level of 33% for the three class problem of low, medium, or high workload. These results suggest that using models based on training from a different task in operational settings will likely not perform well without some form of domain adaptation. Rather, identifying a hierarchy of features and using transfer learning may be a more relevant undertaking for complex real-world tasks.

Ke, et al. used a feature selection procedure coupled with a SVM to perform cross-task classification between verbal and spatial N-back tasks [95]. Ke used a recursive backwards selection procedure that greedily eliminated one feature at a time using 2/3 of the cross-task data while holding out the other 1/3 of data to evaluate final performance [95]. Starting with 180 PSD features spanning 6 frequency bands and 30 EEG channels, participant-specific SVM models were trained on data from one task and tested on the other using all leave-one-feature out combinations. The model that performed the best at each level was used to identify the feature to remove. This same approach was used to identify feature combinations in a seperate cross-task experiment where verbal and spatial N-back tests were grouped together as one task and the alternative task was the MATB [94]. Ke's results in both studies showed that statistically significant improvements in correlation and Mean Squared Error (MSE) were present when using the recursive feature elimination algorithm compared to using all features in the cross-task train-test environment [94, 95]. However, the impact of the results were difficult to interpret because a regression model was used rather than a classification scheme, and correlation between the predicted value and ordinally-assigned class numbers was reported. Furthermore, it was clear that their cross-task performance was still inferior to within-task performance. Another drawback to their approach was that it was very computationally expensive. Finally, this framework for feature identification only enables a salient set of features to be found common to the specific tasks the model is trained and validated on rather than learning something more generally transferable to other task domains.

The important takeaway from Ke's work is that they showed a feature selection procedure improved cross-task performance in a transfer learning scenario using the same two tasks we propose to use in our transfer learning experiment detailed in Chapter III. This provides some assurance that the two tasks are not so dissimilar

as to render transfer learning completely ineffective. Our goal is different than Ke's because we intend to improve performance compared to the within-task results, and to do so in a more general way by using domain adaptation and deep neural networks.

In summary, workload relevant features appear to be different based on the task being performed. In general, using a classifier trained on one task to make predictions based on another does not work well. However, evaluation of cross-task applicability for given domains has only been conducted using shallow features. No research has focused on using deep neural networks for cross-task transfer learning or domain adaptation, nor has research been conducted to distinguish the degree of task similarity between two tasks. These are critical shortfalls in the existing EEG analysis literature since a deep neural network can enable a rich hierarchy of feature representations ranging from very general features at low levels in the network to specific features just prior to the classification level. This spectrum of features makes transfer learning and domain adaptation more successful using deep neural networks and has been shown to increase cross-task classification accuracy beyond those obtained using within-task only training in other domains such as image classification.

Bibliography

1. Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. Tensorflow: A system for large-scale machine learning. In *OSDI* (2016), vol. 16, pp. 265–283. 129

2. Abarbanel, H. D., Brown, R., and Kennel, M. Lyapunov exponents in chaotic systems: their importance and their evaluation using observed data. *International Journal of Modern Physics B 5*, 09 (1991), 1347–1375. 44

3. Abdel-Hamid, O., Deng, L., and Yu, D. Exploring convolutional neural network structures and optimization techniques for speech recognition. In *Interspeech* (2013), pp. 3366–3370. 51

4. Aghajani, H., Garbey, M., and Omurtag, A. Measuring mental workload with eeg+fnirs. *Frontiers in Human Neuroscience 11* (2017), 359. 145

5. Ahmed, E., Jones, M., and Marks, T. K. An improved deep learning architecture for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 3908–3916. 165

6. Allender, L., Kelley, T., Archer, S., and Adkins, R. Imprint: The transition and further development of a soldier-system analysis tool. *Manprint Quarterly 5*, 1 (1997), 1–7. 109

7. An, J., and Cho, S. Hand motion identification of grasp-and-lift task from electroencephalography recordings using recurrent neural networks. In *Big Data and Smart Computing (BigComp), 2016 International Conference on* (2016), IEEE, pp. 427–429. 15, 46

8. An, X., Kuang, D., Guo, X., Zhao, Y., and He, L. A deep learning method for classification of eeg data based on motor imagery. In *International Conference on Intelligent Computing* (2014), Springer, pp. 203–210. 15, 56, 57

9. Andrzejak, R. G., Lehnertz, K., Mormann, F., Rieke, C., David, P., and Elger, C. E. Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state. *Physical Review E 64*, 6 (2001), 061907. 92

10. Astolfi, L., Toppi, J., Borghini, G., Vecchiato, G., Isabella, R., De Vico Fallani, F., Cincotti, F., Salinari, S., Mattia, D., He, B., Caltagirone, C., and Babiloni, F. Study of the functional hyperconnectivity between couples of pilots during flight simulation: An EEG hyperscanning study. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS* (2011), 2338–2341. 15, 201

11. Baldwin, C. L., and Penaranda, B. Adaptive training using an artificial neural network and EEG metrics for within- and cross-task workload classification. *NeuroImage 59*, 1 (2012), 48–56. 15, 212, 213

12. Bashivan, P., Bidelman, G. M., and Yeasin, M. Spectrotemporal dynamics of the eeg during working memory encoding and maintenance predicts individual behavioral capacity. *European Journal of Neuroscience 40*, 12 (2014), 3774–3784. 112

13. Bashivan, P., Rish, I., and Heisig, S. Mental state recognition via wearable eeg. *arXiv preprint arXiv:1602.00985* (2016). 145

14. Bashivan, P., Rish, I., Yeasin, M., and Codella, N. Learning representations from eeg with deep recurrent-convolutional neural networks. In *Proceedings of the International Conference on Learning Representations* (2016). 4, 15, 38, 40, 41, 53, 90, 112, 113

15. Bashivan, P., Yeasin, M., and Bidelman, G. M. Single trial prediction of normal and excessive cognitive load through eeg feature fusion. In *2015 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)* (2015), IEEE, pp. 1–5. 15, 59, 60

16. Battiste, V., and Bortolussi, M. Transport pilot workload: A comparison of two subjective techniques. In *Proceedings of the Human Factors Society Annual Meeting* (1988), vol. 32, SAGE Publications Sage CA: Los Angeles, CA, pp. 150–154. 152

17. Bengio, I. G. Y., and Courville, A. *Deep Learning*. 2016. Book in preparation for MIT Press. 30, 31, 32, 33, 36, 37, 38, 45, 87, 95, 96, 97, 207, 208

18. Bengio, Y., Courville, A., and Vincent, P. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence 35*, 8 (2013), 1798–1828. 29, 30, 207

19. Bengio, Y., et al. Learning deep architectures for ai. *Foundations and trends® in Machine Learning 2*, 1 (2009), 1–127. 37

20. Bengio, Y., et al. Deep learning of representations for unsupervised and transfer learning. *ICML Unsupervised and Transfer Learning 27* (2012), 17–36. 208

21. Bigdely-Shamlo, N., Mullen, T., Kothe, C., Su, K.-M., and Robbins, K. A. The prep pipeline: standardized preprocessing for large-scale eeg analysis. *Frontiers in Neuroinformatics 9* (2015), 16. 122

22. Binz, M., Otte, S., and Zell, A. On the Applicability of Recurrent Neural Networks for Pattern Recognition in Electroencephalography Signals. In

Workshop New Challenges in Neural Computation 2015 (2015), p. 85. 15, 43, 91

23. Blankertz, B., Dornhege, G., Krauledat, M., Müller, K.-R., and Curio, G. The non-invasive berlin brain–computer interface: fast acquisition of effective performance in untrained subjects. *NeuroImage 37*, 2 (2007), 539–550. 43, 91

24. Borghetti, B. J., Giametta, J. J., and Rusnock, C. F. Assessing continuous operator workload with a hybrid scaffolded neuroergonomic modeling approach. *Human factors 59*, 1 (2017), 134–146. 151, 152, 157, 158, 160, 172, 185

25. Borghini, G., Arico, P., Astolfi, L., Toppi, J., Cincotti, F., Mattia, D., Cherubino, P., Vecchiato, G., Maglione, A. G., Graziani, I., and Babiloni, F. Frontal EEG theta changes assess the training improvements of novices in flight simulation tasks. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, 1 (2013), 6619–6622. 15, 16

26. Borghini, G., Astolfi, L., Vecchiato, G., Mattia, D., and Babiloni, F. Measuring neurophysiological signals in aircraft pilots and car drivers for the assessment of mental workload, fatigue and drowsiness. *Neuroscience & Biobehavioral Reviews 44* (2014), 58–75. 6, 16, 68, 76, 151, 192

27. Bowers, M. A., Christensen, J. C., and Eggemeier, F. T. The Effects of Workload Transitions in a Multitasking Environment. *Proceedings of the Human Factors and Ergonomics Society 58th Annual Meeting2* (2014), 220–224. 9, 68, 75, 180, 183

28. Breiman, L. Random Forests. *Machine Learning 45*, 5 (1999), 1–35. 163

29. Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., and Shah, R. Signature verification using a "siamese" time delay neural network. In *Advances in Neural Information Processing Systems* (1994), pp. 737–744. 165

30. Brouwer, A.-M., Hogervorst, M. A., Van Erp, J. B., Heffelaar, T., Zimmerman, P. H., and Oostenveld, R. Estimating workload using eeg spectral power and erps in the n-back task. *Journal of neural engineering 9*, 4 (2012), 045008. 145

31. Callan, D. E., Durantin, G., and Terzibas, C. Classification of single-trial auditory events using dry-wireless EEG during real and motion simulated flight. *Frontiers in systems neuroscience 9*, February (2015), 11. 5, 15, 200, 201

32. Casson, A. J. Artificial neural network classification of operator workload with an assessment of time variation and noise-enhancement to increase performance. *Frontiers in Neuroscience 8* (2014), 372. 15, 17, 19, 39, 84, 85

33. Chandra, S., Verma, K. L., Sharma, G., Mittal, A., and Jha, D. Eeg based cognitive workload classification during nasa matb-ii multitasking. *International Journal of Cognitive Research in Science, Engineering and Education (IJCRSEE) 3*, 1 (2015), 35–41. 15

34. Chaouachi, M., Jraidi, I., and Frasson, C. Modeling mental workload using eeg features for intelligent systems. In *International Conference on User Modeling, Adaptation, and Personalization* (2011), Springer, pp. 50–61. 156

35. Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research 16* (2002), 321–357. 38

36. Chen, J., Hu, B., Xu, L., Moore, P., and Su, Y. Feature-level fusion of multimodal physiological signals for emotion recognition. In *Bioinformatics and Biomedicine (BIBM), 2015 IEEE International Conference on* (2015), IEEE, pp. 395–399. 42

37. Chollet, F. Keras. `https://github.com/fchollet/keras`, 2015. 95, 129

38. Christensen, J. C., and Estepp, J. R. Coadaptive aiding and automation enhance operator performance. *Human Factors* (2013), 965–975. 79, 80

39. Christensen, J. C., Estepp, J. R., Wilson, G. F., and Russell, C. A. The effects of day-to-day variability of physiological data on operator functional state classification. *NeuroImage 59*, 1 (2012), 57–63. 14, 15, 16, 17, 18, 19, 66, 67, 83, 84, 85, 98, 162

40. Clevert, D.-A., Unterthiner, T., and Hochreiter, S. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289* (2016). 54

41. Cohen, D., Wherry, R., and Glenn, F. A critical analysis of workload predictions generated by multiple resource theory during early crewstation design. Tech. rep., 1994. 192, 194

42. Cohen, M. X. *Analyzing neural time series data: theory and practice*. 2014. 16, 17, 23, 70, 93, 161

43. Cohen, T. S., and Welling, M. Group equivariant convolutional networks. *arXiv preprint arXiv:1602.07576* (2016). 32

44. Comstock, J. R., and Arnegard, R. J. The multi-attribute task battery for human operator workload and strategic behavior research. *NASA Technical Memorandum 104174*, January (1992). 68, 83, 120

45. Cooper, G. E., and Harper, R. P., J. The use of pilot rating in the evaluation of aircraft handling qualities. 2, 193, 194

46. Cui, Z., Chen, W., and Chen, Y. Multi-scale convolutional neural networks for time series classification. *arXiv preprint arXiv:1603.06995* (2016). 39

47. Cummings, M. L., Clare, A., and Hart, C. The role of human-automation consensus in multiple unmanned vehicle scheduling. *Human Factors: The Journal of the Human Factors and Ergonomics Society* (2010). 79

48. Davidson, P., Jones, R., and Peiris, M. Detecting behavioral microsleeps using eeg and lstm recurrent neural networks. In *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the* (2006), IEEE, pp. 5754–5757. 15, 45

49. Davidson, P. R., Jones, R. D., and Peiris, M. T. Eeg-based lapse detection with high temporal resolution. *IEEE Transactions on Biomedical Engineering 54*, 5 (2007), 832–839. 91

50. Di Stasi, L. L., Diaz-Piedra, C., Suarez, J., Mccamy, M. B., Martinez-Conde, S., Roca-Dorda, J., and Catena, A. Task complexity modulates pilot electroencephalographic activity during real flights. *Psychophysiology 52*, 7 (2015), 951–956. 5, 201, 202

51. Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., and Darrell, T. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML* (2014), vol. 32, pp. 647–655. 189, 208, 210

52. Elman, J. L. Finding structure in time. *Cognitive science 14*, 2 (1990), 179–211. 43, 110

53. Estepp, J. R., and Christensen, J. C. Physiological cognitive state assessment: Applications for designing effective human-machine systems. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS* (2011), 6538–6541. 65

54. Estepp, J. R., Klosterman, S. L., and Christensen, J. C. An assessment of non-stationarity in physiological cognitive state assessment using artificial neural networks. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS 2011* (2011), 6552–6555. 19, 65, 68, 83

55. Everingham, M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. The pascal visual object classes (voc) challenge. *International journal of computer vision 88*, 2 (2010), 303–338. 209

56. Fazli, S., Popescu, F., Danoczy, M., Blankertz, B., Muller, K. R., and Grozea, C. Subject-independent mental state classification in single trials. *Neural Networks 22*, 9 (2009), 1305–1312. 15, 28, 114, 115

57. Firpi, H. A., Vogelstein, R. J., and Collection, A. D. Particle Swarm Optimization-based Feature Selection for Cognitive State Detection. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS 2011* (2011), 6556–6559. 67

58. Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., and Bouchachia, A. A survey on concept drift adaptation. *ACM computing surveys (CSUR) 46*, 4 (2014), 44. 14

59. Gartner, W. B., and Murphy, M. Pilot workload and fatigue: A critical survey of concepts and assessment techniques. Tech. rep., 1976. 192, 193, 194

60. Gers, F. A., Schmidhuber, J., and Cummins, F. Learning to forget: Continual prediction with lstm. 87, 119, 165

61. Gevins, a., Smith, M. E., Leong, H., McEvoy, L., Whitfield, S., Du, R., and Rush, G. Monitoring working memory load during computer-based tasks with EEG pattern recognition methods. *Human factors 40*, 1 (mar 1998), 79–91. 15, 20, 26, 109, 213

62. Gevins, A., Smith, M. E., McEvoy, L., and Yu, D. High-resolution EEG mapping of cortical activation related to working memory: effects of task difficulty, type of processing, and practice. *Cerebral cortex (New York, N.Y. : 1991) 7*, 4 (jun 1997), 374–85. 15, 16, 68, 76, 214

63. Giametta, J. J. CROSS-SUBJECT CONTINUOUS ANALYTIC WORKLOAD PROFILING USING STOCHASTIC DISCRETE EVENT SIMULATION. Master's thesis, 2015. 159

64. Glorot, X., and Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In *AISTATS* (2010), vol. 9, pp. 249–256. 51

65. Graves, A. Neural networks. In *Supervised Sequence Labelling with Recurrent Neural Networks*. Springer, 2012, pp. 15–35. 4, 81, 86, 119

66. Graves, A., Mohamed, A.-r., and Hinton, G. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing* (2013), IEEE, pp. 6645–6649. 4, 86, 87

67. Graves, A., and Schmidhuber, J. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks 18*, 5 (2005), 602–610. 113

68. Griffin, G., Holub, A., and Perona, P. Caltech-256 object category dataset. 209

69. Güler, N. F., Übeyli, E. D., and Güler, I. Recurrent neural networks employing lyapunov exponents for eeg signals classification. *Expert systems with applications 29*, 3 (2005), 506–514. 15, 43, 44, 91, 110

70. Hajinoroozi, M., Jung, T.-P., Lin, C.-T., and Huang, Y. Feature extraction with deep belief networks for driver's cognitive states prediction from eeg data. In *Signal and Information Processing (ChinaSIP), 2015 IEEE China Summit and International Conference on* (2015), IEEE, pp. 812–815. 15, 60

71. Hajinoroozi, M., Mao, Z., Jung, T.-P., Lin, C.-T., and Huang, Y. Eeg-based prediction of driver's cognitive performance by deep convolutional neural network. *Signal Processing: Image Communication 47* (2016), 549–555. 15, 55, 56, 111, 112

72. Harrivel, A. R., Liles, C. A., Stephens, C. L., Ellis, K. K., Prinzel, L. J., and Pope, A. T. Psychophysiological sensing and state classification for attention management in commercial aviation. *American Institute of Aeronautics and Astronautics, SciTech* (2016). 1

73. Hart, S. G., and Staveland, L. E. Development of nasa-tlx (task load index): Results of empirical and theoretical research. *Advances in psychology 52* (1988), 139–183. 2, 207

74. He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 770–778. 36, 116, 117

75. He, K., Zhang, X., Ren, S., and Sun, J. Identity mappings in deep residual networks. In *European Conference on Computer Vision* (2016), Springer, pp. 630–645. 36, 116, 117, 118

76. Hefron, R., Borghetti, B., Schubert, K. C., Christensen, J., and Estepp, J. Cross-participant eeg-based assessment of cognitive workload using multi-path convolutional recurrent neural networks. *Sensors (Basel, Switzerland) 18*, 5 (2018). 11, 15, 164, 175, 185

77. Hefron, R. G., and Borghetti, B. J. A new feature for cross-day psychophysiological workload estimation. In *Machine Learning and Applications (ICMLA), 2016 15th IEEE International Conference on* (2016), IEEE, pp. 785–790. 8, 9, 15, 30, 84, 93, 161, 175, 180, 183

78. Hefron, R. G., Borghetti, B. J., Christensen, J. C., and Schubert Kabban, C. M. Deep long short-term memory structures model temporal dependencies improving cognitive workload estimation. *Pattern Recognition Letters* (2017). 9, 10, 15, 119, 125, 150, 153, 161, 175, 178, 184, 186

79. Hermans, A., Beyer, L., and Leibe, B. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737* (2017). 166, 167

80. Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. R. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580* (2012). 31, 96, 97

81. Hochreiter, S., and Schmidhuber, J. Long short-term memory. *Neural computation 9*, 8 (1997), 1735–1780. 82, 87, 119, 165

82. Hockey, G. R. J. *Operator functional state: the assessment and prediction of human performance degradation in complex tasks*, vol. 355. IOS Press, 2003. 1

83. Hoepf, M., Middendorf, M., Epling, S., and Galster, S. Physiological Indicators of Workload in a Remotely Piloted Aircraft Simulation (AFRL-RH-WP-TR-2015-0092). 158, 159

84. Huey, B. M., Wickens, C. D., et al. *Workload transition: Implications for individual and team performance*. National Academies Press, 1993. 81

85. Hunn, B. P. The Use of EEG As a Workload Assessment Tool in Flight Test. *DTIC* (1993). 15, 194, 199

86. Ioffe, S., and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167* (2015). 31

87. Jahns, D. W. Operator workload: What is it and how should it be measured. *KD Cross and JJ McGrath (Ed. s) Crew System Design. Santa Barbara, California: Anacapa Sciences* (1973). 16, 23, 65, 83

88. James, G., Witten, D., Hastie, T., and Tibshirani, R. *An introduction to statistical learning*, vol. 112. Springer, 2013. 65, 71, 73, 98, 163

89. Jas, M., Engemann, D. A., Bekhti, Y., Raimondo, F., and Gramfort, A. Autoreject: Automated artifact rejection for meg and eeg data. *arXiv preprint arXiv:1612.08194* (2016). 15

90. Jia, X., Li, K., Li, X., and Zhang, A. A novel semi-supervised deep learning framework for affective state recognition on eeg signals. In *Bioinformatics and Bioengineering (BIBE), 2014 IEEE International Conference on* (2014), IEEE, pp. 30–37. 15, 57, 58

91. Jirayucharoensak, S., Pan-Ngum, S., and Israsena, P. Eeg-based emotion recognition using deep learning network with principal component based covariate shift adaptation. *The Scientific World Journal 2014* (2014). 15, 58

92. Józefowicz, R., Zaremba, W., and Sutskever, I. An empirical exploration of recurrent network architectures. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015* (2015), pp. 2342–2350. 46

93. Kalunga, E., Chevallier, S., and Barthélemy, Q. Data augmentation in riemannian space for brain-computer interfaces. In *ICML Workshop on Statistics, Machine Learning and Neuroscience (Stamlins 2015)* (2015). 15, 39

94. Ke, Y., Qi, H., He, F., Liu, S., Zhao, X., Zhou, P., Zhang, L., and Ming, D. An eeg-based mental workload estimator trained on working memory task can work well under simulated multi-attribute task. *Frontiers in human neuroscience 8* (2014). 15, 216

95. Ke, Y., Qi, H., Zhang, L., Chen, S., Jiao, X., Zhou, P., Zhao, X., Wan, B., and Ming, D. Towards an effective cross-task mental workload recognition model using electroencephalography based on feature selection and support vector machine regression. *International Journal of Psychophysiology 98*, 2 (2015), 157–166. 15, 156, 171, 216

96. Kingma, D., and Ba, J. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations* (2015). 95, 130, 164

97. Koch, G., Zemel, R., and Salakhutdinov, R. Siamese neural networks for one-shot image recognition. In *ICML Deep Learning Workshop* (2015), vol. 2. 165

98. Koelstra, S., Muhl, C., Soleymani, M., Lee, J.-S., Yazdani, A., Ebrahimi, T., Pun, T., Nijholt, A., and Patras, I. Deap: A database for emotion analysis; using physiological signals. *IEEE Transactions on Affective Computing 3*, 1 (2012), 18–31. 41, 42, 57, 58

99. Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (2012), pp. 1097–1105. 35, 38, 51, 86

100. Kumar, S. P., Sriraam, N., Benakop, P., and Jinaga, B. Entropies based detection of epileptic seizures with artificial neural network classifiers. *Expert Systems with Applications 37*, 4 (2010), 3284–3291. 15, 91

101. Laine, T. I., Bauer, K. W., Lanning, J. W., Russell, C. a., and Wilson, G. F. Selection of input features across subjects for classifying crewmember workload using artificial neural networks. *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans. 32*, 6 (2002), 691–704. 3, 9, 15, 25, 26, 68, 75, 108, 162, 180, 183

102. Lawhern, V. J., Solon, A. J., Waytowich, N. R., Gordon, S. M., Hung, C. P., and Lance, B. J. Eegnet: A compact convolutional network for eeg-based brain-computer interfaces. *arXiv preprint arXiv:1611.08024* (2016). 15, 53, 54, 111, 118, 128

103. LeCun, Y., Bengio, Y., and Hinton, G. Deep learning. *Nature 521*, 7553 (2015), 436–444. 4, 7, 30, 31, 32

104. LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE 86*, 11 (1998), 2278–2324. 49

105. Lee, J. D., and See, K. A. Trust in automation: Designing for appropriate reliance. *Human Factors: The Journal of the Human Factors and Ergonomics Society 46*, 1 (2004), 50–80. 80

106. Li, X., Song, D., Zhang, P., Yu, G., Hou, Y., and Hu, B. Emotion recognition from multi-channel eeg data through convolutional recurrent neural network. In *Bioinformatics and Biomedicine (BIBM), 2016 IEEE International Conference on* (2016), IEEE, pp. 352–359. 15, 38, 41, 42

107. Lin, M., Chen, Q., and Yan, S. Network in network. *arXiv preprint arXiv:1312.4400* (2013). 33, 118, 126, 127

108. Lin, Y., Wang, Y., Wei, C., and Jung, T. Assessing the quality of steady-state visual-evoked potentials for moving humans using a mobile electroencephalogram headset. *Frontiers in human neuroscience 8* (2014). 200

109. Lin, Y.-P., Hsu, S.-H., and Jung, T.-P. Exploring day-to-day variability in the relations between emotion and EEG signals. In *Augmented Cognition, LNCS* (2015), pp. 461–469. 15, 19, 20

110. Lipton, Z. C., Berkowitz, J., and Elkan, C. A critical review of recurrent neural networks for sequence learning. *arXiv preprint arXiv:1506.00019* (2015). 86

111. Liu, W., Zheng, W.-L., and Lu, B.-L. Emotion recognition using multimodal deep learning. In *International Conference on Neural Information Processing* (2016), Springer, pp. 521–529. 61

112. Liu, Y.-T., Lin, Y.-Y., Wu, S.-L., Chuang, C.-H., and Lin, C.-T. Brain dynamics in predicting driving fatigue using a recurrent self-evolving fuzzy neural network. *IEEE transactions on neural networks and learning systems 27*, 2 (2016), 347–360. 15, 47, 48, 55, 110, 111

113. Liu, Y.-T., Wu, S.-L., Chou, K.-P., Lin, Y.-Y., Lu, J., Zhang, G., Lin, W.-C., and Lin, C.-T. Driving fatigue prediction with pre-event electroencephalography (eeg) via a recurrent fuzzy neural network. In *Fuzzy Systems (FUZZ-IEEE), 2016 IEEE International Conference on* (2016), IEEE, pp. 2488–2494. 15, 48

114. Liyanage, S. R., Guan, C., Zhang, H., Ang, K. K., Xu, J., and Lee, T. H. Dynamically weighted ensemble classification for non-stationary EEG processing. *Journal of neural engineering 10* (2013), 036007. 15, 16, 22, 83

115. Long, M., Cao, Y., Wang, J., and Jordan, M. I. Learning transferable features with deep adaptation networks. In *ICML* (2015), pp. 97–105. 208, 209

116. Lotte, F., Congedo, M., Anatole, L., Lotte, F., Congedo, M., and Anatole, L. A review of classification algorithms for EEG-based braincomputer interfaces. *Journal of neural engineering 4*, 2 (2007), R1–R13. 66

117. Mash, R., Borghetti, B., and Pecarina, J. Improved aircraft recognition for aerial refueling through data augmentation in convolutional neural networks. In *International Symposium on Visual Computing* (2016), Springer, pp. 113–122. 35, 38

118. Mazumder, A., Rakshit, A., and Tibarewala, D. A back-propagation through time based recurrent neural network approach for classification of cognitive eeg states. In *Engineering and Technology (ICETECH), 2015 IEEE International Conference on* (2015), IEEE, pp. 1–5. 15, 46

119. McDonald, N. J., and Soussou, W. in the Cognitive State Assessment Competition 2011. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS* (2011), 6542–6546. 76

120. Mirowski, P. W., LeCun, Y., Madhavan, D., and Kuzniecky, R. Comparing svm and convolutional networks for epileptic seizure prediction from intracranial eeg. In *Machine Learning for Signal Processing, 2008. MLSP 2008. IEEE Workshop on* (2008), IEEE, pp. 244–249. 15, 49, 50

121. Mitchell, D. K. Mental Workload and ARL Workload Modeling Tools. *Army Research Laboratory*, April (2000), 35. 153, 156

122. Ng, A. Y. Feature selection, l 1 vs. l 2 regularization, and rotational invariance. In *Proceedings of the twenty-first international conference on Machine learning* (2004), ACM, p. 8. 30, 50

123. Noel, J. B., Bauer, K. W., and Lanning, J. W. Improving pilot mental workload classification through feature exploitation and combination: A feasibility study. *Computers and Operations Research 32*, 10 (2005), 2713–2730. 15, 21, 24, 108

124. Oquab, M., Bottou, L., Laptev, I., and Sivic, J. Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2014), pp. 1717–1724. 189, 209, 210

125. Parasuraman, R., Bahri, T., Deaton, J. E., MOrrison, J. G., and Barnes, M. Theory and design of adaptive automation in aviation systems. Tech. rep., Warminster, PA, 1992. 6, 61, 64, 80, 85, 194

126. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research 12*, Oct (2011), 2825–2830. 98

127. Pham, V., Bluche, T., Kermorvant, C., and Louradour, J. Dropout improves recurrent neural networks for handwriting recognition. In *Frontiers in Handwriting Recognition (ICFHR), 2014 14th International Conference on* (2014), IEEE, pp. 285–290. 97

128. Popovic, D., Stikic, M., and Rosenthal, T. Sensitive, Diagnostic and Multifaceted Mental Workload Classifier (PHYSIOPRINT). In *Foundations of Augmented Cognition* (2015), vol. 9183, pp. 101–111. 15, 27

129. Raichle, M. E., and Snyder, A. Z. A default mode of brain function: a brief history of an evolving idea. *Neuroimage 37*, 4 (2007), 1083–1090. 80

130. Ren, Y., and Wu, Y. Convolutional deep belief networks for feature extraction of eeg signal. In *2014 International Joint Conference on Neural Networks (IJCNN)* (2014), IEEE, pp. 2850–2853. 15, 58

131. Rouse, W. B., Cody, W. R., and Boff, K. R. The human factors of system design: Understanding and enhancing the role of human factors engineering. *International Journal of Human Factors in Manufacturing 1*, 1 (jan 1991), 87–104. 61, 80, 85

132. Ruck, D. W., Rogers, S. K., and Kabrisky, M. Feature selection using a multilayer perceptron. *Journal of Neural Network Computing 2*, 2 (1990), 40–48. 154, 177, 186

133. Ruffini, G., Ibañez, D., Castellano, M., Dunne, S., and Soria-Frisch, A. Eeg-driven rnn classification for prognosis of neurodegeneration in at-risk patients. In *International Conference on Artificial Neural Networks* (2016), Springer, pp. 306–313. 15, 45, 46

134. Rusnock, C. F., and Borghetti, B. J. Workload profiles: A continuous measure of mental workload. *International Journal of Industrial Ergonomics* (2016). 153

135. Rusnock, C. F., Borghetti, B. J., and McQuaid, I. Objective-analytical measures of workloadthe third pillar of workload triangulation? In *Foundations of Augmented Cognition*, D. D. Schmorrow and C. M. Fidopiastis, Eds., vol. 9183 of *Lecture Notes in Computer Science*. Springer International Publishing, 2015, pp. 124–135. 152, 159, 185

136. Saha, A., Konar, A., Chatterjee, A., Ralescu, A., and Nagar, A. K. Eeg analysis for olfactory perceptual-ability measurement using a recurrent neural classifier. *IEEE Transactions on Human-Machine Systems 44*, 6 (2014), 717–730. 15

137. Sanei, S., and Chambers, J. A. *EEG signal processing*. John Wiley & Sons, 2013. 3

138. Schapire, R. E. The Strength of Weak Learnability. *Machine Learning 5*, 2 (1990), 197–227. 72

139. Schmidhuber, J. Deep learning in neural networks: An overview. *Neural networks 61* (2015), 85–117. 36, 119

140. Schnell, T., Keller, M., and Poolman, P. Neurophysiological workload assessment in flight. *AIAA/IEEE Digital Avionics Systems Conference - Proceedings* (2008), 1–14. 5, 199

141. Schnell, T., Melzer, J. E., and Robbins, S. J. The Cognitive Pilot Helmet: enabling pilot-aware smart avionics. *Proceedings of SPIE 7326*, April (2009), 73260A–73260A–9. 15, 195, 200

142. Schroff, F., Kalenichenko, D., and Philbin, J. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2015), pp. 815–823. 165, 166

143. Shoeb, A. H. *Application of machine learning to epileptic seizure onset detection and treatment*. PhD thesis, Massachusetts Institute of Technology, 2009. 43

144. Shrivastava, A., Gupta, A., and Girshick, R. Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 761–769. 167

145. Simard, P. Y., Steinkraus, D., Platt, J. C., et al. Best practices for convolutional neural networks applied to visual document analysis. In *ICDAR* (2003), vol. 3, Citeseer, pp. 958–962. 38

146. Simonyan, K., and Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014). 35, 36

147. Smith, A. M., Borghetti, B. J., and Rusnock, C. F. Improving model cross-applicability for operator workload estimation. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* (2015), vol. 59, SAGE Publications Sage CA: Los Angeles, CA, pp. 681–685. 109, 155, 158

148. Soleymani, M., Asghari-Esfeden, S., Pantic, M., and Fu, Y. Continuous emotion detection using eeg signals and facial expressions. In *Multimedia and Expo (ICME), 2014 IEEE International Conference on* (2014), IEEE, pp. 1–6. 9, 15, 48, 49, 180

149. Srinivasan, V., Eswaran, C., and Sriraam, N. Approximate entropy-based epileptic eeg detection using artificial neural networks. *IEEE Transactions on information Technology in Biomedicine 11*, 3 (2007), 288–295. 15, 91

150. Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research 15*, 1 (2014), 1929–1958. 30, 96, 130

151. Sterman, M. B., Mann, C. A., and Kaiser, D. A. Quantitative EEG Patterns of Differential In-flight Workload. Tech. rep., 1992. 15, 198, 199

152. Sterman, M. E., Schummer, G. J., Dushenko, T. W., and Smith, J. C. Electroencephalographic Correlates of Pilot Performance: Simulation and In-flight Studies. *Electrical and magnetic activity of the central nervous system: research and clinical applications in aerospace medicine, AGARD CP-432* (1987), 31.1–31.16. 15, 197

153. Stober, S., Cameron, D. J., and Grahn, J. A. Using convolutional neural networks to recognize rhythm stimuli from electroencephalography recordings. In *Advances in neural information processing systems* (2014), pp. 1449–1457. 15, 54, 55

154. Sun, Y., Chen, Y., Wang, X., and Tang, X. Deep learning face representation by joint identification-verification. In *Advances in neural information processing systems* (2014), pp. 1988–1996. 165

155. Szegedy, C., Ioffe, S., Vanhoucke, V., and Alemi, A. A. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI* (2017), vol. 4, p. 12. 117, 118, 126, 128

156. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 1–9. 36, 116, 126, 127

157. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 2818–2826. 36, 116, 118

158. Tabar, Y. R., and Halici, U. A novel deep learning approach for classification of eeg motor imagery signals. *Journal of Neural Engineering 14*, 1 (2016), 016003. 15, 50, 52

159. Taigman, Y., Yang, M., Ranzato, M., and Wolf, L. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2014), pp. 1701–1708. 165

160. Tang, Y. Deep learning using linear support vector machines. *arXiv preprint arXiv:1306.0239* (2013). 55

161. Tang, Z., Li, C., and Sun, S. Single-trial eeg classification of motor imagery using deep convolutional neural networks. *Optik-International Journal for Light and Electron Optics 130* (2017), 11–18. 15, 51

162. Taya, F., Sun, Y., Borghini, G., Aricò, P., Babiloni, F., Bezerianos, A., and Thakor, N. V. Training-induced changes in information transfer efficiency of the brain network: A functional connectome approach. In *2015 7th International IEEE/EMBS Conference on Neural Engineering (NER)* (2015), IEEE, pp. 1028–1031. 15, 16

163. Theano Development Team. Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints abs/1605.02688* (May 2016). 95

164. Thodoroff, P., Pineau, J., and Lim, A. Learning robust features using deep learning for automatic seizure detection. In *Machine Learning for Healthcare Conference* (2016), pp. 178–190. 15, 42, 43, 113

165. Turner, J., Page, A., Mohsenin, T., and Oates, T. Deep belief networks used on high resolution multichannel electroencephalography data for seizure detection. In *2014 AAAI Spring Symposium Series* (2014). 15, 60, 61

166. Übeyli, E. D. Analysis of eeg signals by implementing eigenvector methods/recurrent neural networks. *Digital Signal Processing 19*, 1 (2009), 134–143. 15, 44, 91, 110

167. Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. Show and tell: A neural image caption generator. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 3156–3164. 4, 86

168. Walker, I., Deisenroth, M., and Faisal, A. Deep convolutional neural networks for brain computer interface using motor imagery. *Imperial College of Science, Technology and Medicine Department of Computing* (2015). 15, 53

169. Wang, X., and Gupta, A. Unsupervised learning of visual representations using videos. *arXiv preprint arXiv:1505.00687* (2015). 165

170. Wang, Z., Hope, R. M., Wang, Z., Ji, Q., and Gray, W. D. Cross-subject workload classification with a hierarchical bayes model. *NeuroImage 59*, 1 (2012), 64–69. 108

171. Weinberger, K. Q., and Saul, L. K. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research 10*, Feb (2009), 207–244. 166

172. Wilson, G. In-flight psychophysiological monitoring. *Progress in ambulatory monitoring* (2001), 435–454. 79

173. Wilson, G. F. Psychophysiological test methods and procedures. *Handbook of human factors testing and evaluation* (2002), 127–156. 79

174. Wilson, G. F., Fullenkamp, P., and Davis, I. Evoked potential, cardiac, blink, and respiration measures of pilot workload in air-to-ground missions. *Aviation Space and Environmental Medicine 65*, 2 (1994), 100–105. 15, 198

175. Wilson, G. F., Reis, G. A., and Tripp, L. D. EEG correlates of G-induced loss of consciousness. *Aviation, space, and environmental medicine 76*, 1 (jan 2005), 19–27. 6

176. Wilson, G. F., Russell, C., Monnin, J., Estepp, J., and Christensen, J. How does day-to-day variability in psychophysiological data affect classifier accuracy? In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* (2010), vol. 54, SAGE Publications Sage CA: Los Angeles, CA, pp. 264–268. 15, 83

177. Wilson, G. F., and Russell, C. A. Operator functional state classification using multiple psychophysiological features in an air traffic control task. *Human Factors 45*, 3 (2003), 381–389. 15, 214

178. Wilson, G. F., and Russell, C. a. Real-time assessment of mental workload using psychophysiological measures and artificial neural networks. *Human factors 45*, 4 (2003), 635–643. 6, 15, 23, 24, 80, 108, 162

179. Wilson, G. F., and Russell, C. A. Performance enhancement in an uninhabited air vehicle task using psychophysiologically determined adaptive aiding. *Human Factors 49*, 6 (2007), 1005–1018. 15, 24, 25, 79

180. Yang, H., Sakhavi, S., Ang, K. K., and Guan, C. On the use of convolutional neural networks and augmented csp features for multi-class motor imagery of eeg signals classification. In *Engineering in Medicine and Biology Society (EMBC), 2015 37th Annual International Conference of the IEEE* (2015), IEEE, pp. 2620–2623. 15, 50

181. Yin, Z., Wang, Y., Liu, L., Zhang, W., and Zhang, J. Cross-subject eeg feature selection for emotion recognition using transfer recursive feature elimination. *Frontiers in neurorobotics 11* (2017). 108

182. Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. How transferable are features in deep neural networks? In *Advances in neural information processing systems* (2014), pp. 3320–3328. 189, 208, 210, 211

183. Zaremba, W., Sutskever, I., and Vinyals, O. Recurrent neural network regularization. *arXiv preprint arXiv:1409.2329* (2014). 97

184. Zhang, J., Yin, Z., and Wang, R. Recognition of mental workload levels under complex human–machine collaboration by using physiological features and adaptive support vector machines. *IEEE Transactions on Human-Machine Systems 45*, 2 (2015), 200–214. 108

185. Zheng, W.-L., and Lu, B.-L. Investigating critical frequency bands and channels for eeg-based emotion recognition with deep neural networks. *IEEE Transactions on Autonomous Mental Development 7*, 3 (2015), 162–175. 9, 15, 59, 180

186. Zheng, W.-L., Zhu, J.-Y., Peng, Y., and Lu, B.-L. Eeg-based emotion classification using deep belief networks. In *Multimedia and Expo (ICME), 2014 IEEE International Conference on* (2014), IEEE, pp. 1–6. 15, 60

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
13-09-2018	Doctoral Dissertation	Sept 2015 — Sept 2018

4. TITLE AND SUBTITLE

Breaking down the barriers to operator workload estimation: Advancing algorithmic handling of temporal non-stationarity and cross-participant differences for EEG analysis using deep learning

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Hefron, Ryan G, Maj

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT-ENG-DS-18-S-012

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Intentionally Left Blank

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A:
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This research focuses on two barriers to using EEG data for workload assessment: day-to-day variability, and cross-participant applicability. Several signal processing techniques and deep learning approaches are evaluated in multi-task environments. These methods account for temporal, spatial, and frequential data dependencies. Variance of frequency-domain power distributions for cross-day workload classification is statistically significant. Skewness and kurtosis are not significant in an environment absent workload transitions, but are salient with transitions present. LSTMs improve day-to-day feature stationarity, decreasing error by 59% compared to previous best results. A multi-path convolutional recurrent model using bi-directional, residual recurrent layers significantly increases predictive accuracy and decreases cross-participant variance. Deep learning regression approaches are applied to a multi-task environment with workload transitions. Accounting for temporal dependence significantly reduces error and increases correlation compared to baselines. Visualization techniques for LSTM feature saliency are developed to understand EEG analysis model biases.

15. SUBJECT TERMS

EEG, electroencephalograph, deep learning, RNN, LSTM, CNN, Siamese network, Siamese-triplet, psychophysiological, workload, non-stationary environment, day-to-day variability, cross-participant, individual differences

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Dr. Brett J. Borghetti, AFIT/ENG
U	U	U	UU	252	19b. TELEPHONE NUMBER *(include area code)* (937) 255-3636, x4612; brett.borghetti@afit.edu