DEVELOPING TRANSFERABLE DEEP MODELS FOR MOBILE HEALTH

A Dissertation
Presented to
The Academic Faculty

By

Supriya Nagesh

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology

December 2022

DEVELOPING TRANSFERABLE DEEP MODELS FOR MOBILE HEALTH

Thesis committee:

Dr. James M. Rehg, Advisor
School of Interactive Computing
Georgia Institute of Technology

Dr. Omer Inan, Co-advisor
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. David Anderson
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Judy Hoffman
School of Interactive Computing
Georgia Institute of Technology

Dr. Inbal Nahum-Shani
Institute for Social Research
University of Michigan

Dr. Santosh Kumar
Department of Computer Science
University of Memphis

Date approved: September 23, 2022

This thesis is dedicated to my family

ACKNOWLEDGMENTS

This thesis has been possible due to the support and encouragement provided by several people over the course of my PhD. First, I would like to express my gratitude to my advisor Jim Rehg. His curiosity, passion, and enthusiasm for research were infectious and I came out of every meeting with renewed energy. He will always be a source of motivation in my career.

I would like to thank my dissertation committee members - Omer Inan, Billie Nahum-Shani, Judy Hoffman, Santosh Kumar, and David Anderson. The discussion I had with each of them was valuable in shaping different parts of this thesis.

I spent a wonderful summer as an intern at Amazon and got to work on an exciting problem at the intersection of healthcare and machine learning. I would like to thank Nina Mishra and Yonatan Naamad for giving me this opportunity.

Next, I would like to thank my lab members. Particularly Alex Moreno, Burak Aydemir, and Max Xu for their support inside and outside of grad school. My time at Georgia Tech was made memorable by my friends, who were my family and support system in Atlanta.

This thesis has been possible mainly due to the support and many sacrifices made by my family in India. I am grateful to them for bearing with my infrequent calls and visits home. I would like to thank my parents for trusting me and giving me opportunities that they could never dream of. I'd like to thank my grandmother for her encouragement, constant support and affection and my brother for being a source of motivation and a role model over the course of my education.

One of the best outcomes of my time at Tech has been meeting Rakshith. I am grateful to him for supporting me through the highs and lows of grad school, lending me an ear while I complained about my coursework, and bearing with me in general while I put off all plans until 'after my defense'. I am excited for our next phase of life together.

TABLE OF CONTENTS

Acknowledgments iv

List of Tables ix

List of Figures xi

List of Acronyms xvi

Summary xvii

Chapter 1: Introduction 1

1.1 Thesis statement 5

1.2 Overview 5

1.2.1 Transformers for EMA non-response prediction under covariate-space shift 5

1.2.2 Domain adaptation through self-supervised learning for pulsative physiological signals 6

1.2.3 CFVAE: Counterfactual VAE for explainable ranking of patients for home hospital care 7

1.3 Contributions 9

Chapter 2: Transformers for Ecological Momentary Assessment non-response prediction under covariate-space shift 11

2.1 Introduction 11

2.2 Contributions 14

2.3 Related work 15

2.3.1 Analyzing and predicting EMA non-response 16

2.3.2 Deep models for EMA data 18

2.3.3 Transformers for Electronic Health Records Data 18

2.4 Study protocol and dataset 19

2.5 Methodology 21

2.5.1 Non-response problem framing 21

2.5.2 Dataset specific feature construction: 22

2.5.3 Valence feature construction 24

2.5.4 Transformer model 24

2.5.5 Self-supervised pre-training of EMA transformer 27

2.5.6 Input representation for zero shot transfer to other EMA datasets . . 31

2.6 Experimental results 31

2.6.1 Predicting non-response to the next EMA using the current EMA response 32

2.6.2 Predicting non-response to the next EMA using a sequence of EMA responses 32

2.6.3 Role of temporal/positional encoding: 36

2.6.4 Predictive performance of valence features 36

2.6.5 Generalization to other EMA studies 37

2.6.6 Feature ablation results 39

2.7 Discussion 40

2.7.1 Importance of temporal history in non-response prediction 40

2.7.2 Temporal/positional encoding for EMA input to transformers 41

2.7.3 Self-supervised pre-training of EMA transformer 41

2.7.4 Interpreting the learned attention weights 42

2.7.5 Generalization to other EMA datasets 44

2.8 Conclusion 44

Chapter 3: Domain adaptation through self-supervised learning for pulsative physiological signals 46

3.1 Introduction 46

3.2 Contributions 49

3.3 Related work 50

3.4 Methods 51

3.4.1 Arrhythmia classification from ECG 51

3.4.2 Respiration rate detection from PPG 52

3.4.3 Domain adaptation through self-supervised tasks 52

3.4.4 Finding an invariant input-representation 56

3.5 Experiments and results 56

3.5.1 Classification results without domain adaptation 56

3.5.2 Using domain adaptation on ECG and PPG 57

3.6 Discussion 61

3.6.1 Invariant input representation 61

3.6.2 Performance of self-supervised domain adaptation when trained on source data only 61

Chapter 4: Explainable patient ranking for home hospital care 64

4.1 Introduction 64

4.2 Contributions 67

4.3 Related work 67

4.3.1 Explainability via Counterfactual Generation 69

4.3.2 Patient ranking 70

4.3.3 Predicting Time to Intervention 70

4.4 Methods 71

4.4.1 CF VAE: Counterfactual Variational Autoencoder 71

4.4.2 Multitask learning of ranking and acute intervention prediction . . . 75

4.5 Experiments and Results 76

4.5.1 Dataset and pre-processing 78

4.5.2 Baseline methods for generating counterfactuals 78

4.5.3 Multitask learning of ranking and acute intervention prediction . . . 79

4.5.4 Visualization of counterfactuals 81

4.5.5 Evaluation of counterfactuals 83

4.5.6 Impact of the sparsity term on the counterfactuals 84

4.6 Conclusion and future work 85

Chapter 5: Conclusions and future work 91

References 93

LIST OF TABLES

2.1 Features used in our analysis - we include both raw features and summary features. The raw features directly obtained from the EMA response. We compute additional summary features from the history of the raw features. We perform an ablation without using 'Summary features' in Table Table 2.3 to assess the importance of history. 23

2.2 Cross validation AUC for predicting non-response to next EMA prompt using a sequence of $T = 15$ EMAs with different pre-training tasks. The different pre-training tasks here are: 1. Imputing one emotion item at a time, 2. Imputing all the emotion items, 3. Imputing five emotion items. We do not see a large change in performance with a change in the pre-training task. 33

2.3 Average 5 fold cross validation AUC for predicting next EMA response using features from the current EMA (sequence length T = 1). We use two sets of features: 1) raw features only, 2) raw features and summary features. Using summary features improves performance significantly, indicating that history is important in predicting response. 34

2.4 Average 5 fold cross validation AUC for predicting non-response to next EMA using a sequence of T EMAs. The deep models LSTM and Transformer perform significantly better than Logistic regression. Among the deep models, the EMA transformer with pre-training performs the best. The transformer performance is significantly better than LSTM when using longer sequences ($T = 15, 25$). All of these results are 5 fold cross validation results on the main dataset. Note that here we use dataset specific features for prediction. 35

2.5 Average 5 fold cross validation AUC for predicting non-response to next EMA using a sequence of T EMAs in the following settings: a) Using all the emotion items in each dataset, b) Using the valence feature representation, c) Transfer performance (e.g., model trained on dataset B and tested on dataset A), d) Transfer performance after performing domain adaptation using DeepCoral [22]. We see that using the valence features results in a slight degradation in performance. However, valence features give us the ability to use the trained model on other datasets. We see that domain adaptation provides a gain in the transfer performance. 38

4.1 Comparison of different counterfactual generation methods: CEM [144], REVISE [51], DICE [52], NG-CF [145], CF VAE - this work. 69

4.2 Multitask model results from three experiments: 1. Ranking and acute intervention prediction using history of vitals, interventions and demographics, 2. Results *without using* the history of the target intervention as an input feature (to rule out any data leakage), 3. Results when we perform ranking only on the patients who require acute intervention within the next 24 hours. *Error bars generated by running the experiment with 10 random seeds.* Here, ACC = Accuracy, AUC = Area under ROC curve, Intv = intervention. 80

4.3 Comparison of [52, 51] and CF VAE. $l_{method} > l_{CFVAE}$ is the fraction of test points where the log likelihood of a CF from prior work exceeded that of our own. We see that in the case of Ventilator, only 2% of the test CF from DICE had a higher likelihood score than our CF. The physician evaluated our counterfactual for plausibility and relevance. The physician's score is the fraction of CF that were deemed both *plausible* and *relevant*. Note that we don't have physician score for REVISE because of its poor validity percentage. 90

LIST OF FIGURES

2.1 (a) EMA notification on the study phone (b) Survey question: angry (c) Survey question: relaxed. 21

2.2 Sliding window approach used to create our feature vector sequence and classification label. EMAs are prompted irregularly in time, feature vectors from a sequence of T EMAs are used to predict compliance to the $(T+1)^{\text{th}}$ EMA. 22

2.3 EMA Transformer model architecture. This includes a transformer encoder followed by a linear layer. 28

2.4 Pre-training the EMA transformer model with an EMA imputation task. The pre-trained model is then fine-tuned for the non-response prediction task. This pre-training strategy is similar to BERT where the model is pre-trained in a self-supervised manner on a large text corpus and the model is fine-tuned for downstream tasks. 29

2.5 Self-supervised EMA masked imputation tasks. Here we assume that there are K emotion items in each EMA. The input sequence here is depicting *only* the responses to emotion items. We do not mask out other features in the input sequence. (a) The first task shown is to impute one emotion item at a time in 15% of the sequence positions that are randomly masked. For example, the value of the emotion 'Happy' can be masked off in some positions of the sequence and the task is to impute this value correctly. Note that we explore imputing each emotion item one at a time as a pre-training task and evaluate the downstream non-response prediction performance. *There are intermediate tasks possible such as masking out 3 emotion items, 4 emotion items, etc.* (b) The second task is to impute all emotion items in 15% of the sequence. 30

2.6 Performance of transformer with different temporal/positional encoding (a) On the pre-training task of imputing EMA responses (here we impute the features one at a time); (b) On the downstream response prediction task with sequence length $T = 15$. The box plot shows performance across different cross-validation folds. *Concatenation* performs the best in all these cases and *temporal concatenation* performs the best in the prediction task. . 36

2.7 Feature ablation study: the performance of EMA transformer is evaluated in 4 settings: 1. using all features, 2. removing positive emotions, 3. removing negative emotions, 4. removing non-EMA features. We see the largest drop in performance when the positive emotion items are removed, indicating that the positive emotions have the most predictive power. 39

2.8 Distribution of responses to positive and negative questions. Notice that negative questions usually have a lower value response and positive questions have a higher value response. 40

2.9 (a)Attention weight across the positions in an EMA sequence of length $T = 5$. Here the EMA positions in the sequence are denoted as Lag 5 - Lag 1 and the EMA for which response is being predicted is Lag 0. So Lag 1 is the EMA immediately before the one being predicted. (b)Attention weights on the different features in the input layer (first encoder layer) averaged across the time window. This visualizes the weighting performed by self-attention matrix on the input features. 44

3.1 Standard 12 lead ECG set up. Each lead records the cardiac depolarization in a different direction. The signal recorded at two leads on the chest - V_3, V_5 are illustrated. Notice that the structure of the signal recorded at each lead is different. 48

3.2 Architecture used for arrhythmia classification from ECG. This is our main predictive task on ECG signals to study domain shift across different leads. We train an arrhythmia classification model with this architecture on one domain and evaluate its performance on a different domain. 51

3.3 Architecture of the respiration rate detection model. This is our main predictive task on PPG signals to study domain shift. We train a respiration rate regression model on one domain and evaluate its performance on a different domain. 52

3.4 Example of flipping operation performed to create an augmented dataset for the self-supervised task. Here we show the original signal which is a Lead 1 ECG signal, and the flipped signal on the right. The auxiliary self-supervised task is to detect signals that have been flipped. 53

3.5 Example of mean shifting operation performed to create an augmented dataset for the self-supervised task of mean shifting prediction. The red dotted line indicates the mean of the original signal. 54

3.6 Example of jumbling operation performed to create a dataset for the auxiliary self-supervised task of detecting jumbled signals. Here we show the original signal which is randomly segmented into 10 segments. The segments are then permuted and combined to obtain the jumbled signal. Each segment is denoted by a unique symbol in this figure. Notice that the segments are ordered randomly on the right to result in the jumbled signal which no longer has the structure of a typical ECG signal. 54

3.7 Domain adaptation through general self-supervised tasks. The goal here is to align source and target representations through auxiliary tasks while preserving the task-specific discriminability of the representations. We simultaneously train the network to perform three self-supervised tasks on the source and target domain. In addition, we train the task head using source domain labels. The three SS tasks we use are detecting: 1) Flip left-right, 2) jumbled segments, 3) mean shifted signals. 55

3.8 Domain adaptation through physiologically-inspired self-supervised task. Here the goal is to align source and target representations at the encoder while preserving task-specific discriminability. We simultaneously train the model to perform the main task with source labels and the auxiliary self-supervised task on the source and target domains. The auxiliary self-supervised task used is peak detection (R-peak in the case of ECG). 56

3.9 Root mean squared error (RMSE - lower is better) of the respiration rate (RR) detection from PPG. This figure denotes the error when the RR regression model is trained on the source (WESAD/MIMIC) dataset and tested on the target (WESAD/MIMIC) dataset. Notice that the performance is best when the source and target dataset are the same, depicted by the lower error values in the diagonal entries. 57

3.10 AUROC (higher is better) value for arrhythmia classification from ECG. Here we present the performance when the model is trained and tested on different leads (Lead 1-6 in PTBXL). There is no domain shift when the source and the target leads are the same. This corresponds to the high AUROC values observed in the diagonal elements. Notice that the performance is poor when the model is tested on Leads 3 and 4. These are the leads with the largest domain shift when compared to the other leads. 58

3.11 Difference in arrhythmia classification (from ECG) performance to supervised learning (lower is better) when both, the time and frequency domain is used. The supervised performance for each target lead is the corresponding diagonal entry in Figure 3.10. The results shown here reflect the average performance when all other leads are used as the source for a given target lead. Notice that using the frequency domain (even without any domain adaptation) yields an improvement over time domain. We see that physiological self-supervised domain adaptation (SSDA) performs the best 59

3.12 Difference in arrhythmia classification (from ECG) performance to supervised learning (lower is better) when the methods are applied to the time domain signal. The supervised performance for each target lead is the corresponding diagonal entry in Figure 3.10. The results shown here reflect the average performance when all other leads are used as the source for a given target lead. We see that physiological self-supervised domain adaptation (SSDA) performs the best. 60

3.13 RMSE of RR detection from PPG when using the time domain signal. Lower is better. 60

3.14 RMSE of RR detection from PPG when using both time and frequency domain. Lower is better. 61

3.15 Difference in arrhythmia classification (from ECG) performance to supervised learning when using the self-supervised task on source domain only (No target domain data is used during training). We see that self-supervised tasks act as a regularizer and improve target domain generalization. 62

3.16 RMSE of RR detection from PPG when using the self-supervised task on the source domain only. 62

4.1 Our proposed approach: learn a VAE latent space that embeds input data points onto regions of the latent space whose labels are opposite those of the points. This can be used to generate counterfactuals for any binary prediction (BP) model. 68

4.2 TSNE plot of the latent space of a vanilla VAE and CF VAE with varying λ_{cf}. We visualize the two classes each point belongs to: requires intervention, or doesn't require intervention. We see that the CF VAE captures the classifier boundary and learns to separate the two classes in the latent space as λ_{cf} increases. 75

4.3 Model architecture for producing ranking and acute intervention prediction output. The architecture consists of a transformer encoder for modeling the temporal sequence of patient data. 77

4.4 Training pipeline for the ranking and predicting intervention. The multitask model shown here is the MLP (for slope-intercept form) and transformer model (for temporal sequence form). 77

4.5 Counterfactual (CF) generated using DICE and CF VAE. The original patient (blue dotted) requires a vasopressor within 24 hours. The orange line shows the generated CF. The CF generated using DICE neither looks plausible nor relevant. Vasopressors are provided when the blood pressure drops - the DICE CF changes oxygen saturation and respiration rate 82

4.6 Confusion matrix of the 24-hour intervention prediction task. The model falsely recommends 12% (resp., 8%) of patients who actually needed a ventilator (resp., vasopressor) for the home hospital program although they should remain in-hospital. 85

4.7 Pair-wise ranking error rate based on the difference in time to intervention between the pairs. In the top figure, the error rate is lower when one of the patients requires an intervention in $>$ 24 hours. This is important since we want to identify such candidates for home hospital. The bottom shows ranking performance when evaluated only on pairs where an acute intervention was required within 24 hours - we see a larger error rate when the two patients are $<$ 5 hours apart. 86

4.8 Example 1: Counterfactual produced with and without the sparsity term. Notice how only the systolic blood pressure changes when we include the sparsity constraint. 88

4.9 Example 2: Counterfactual produced with and without the sparsity term. . 89

SUMMARY

Human behavior is one of the key facets of health. A major portion of healthcare spending in the US is attributed to chronic diseases, which are linked to behavioral risk factors such as smoking, drinking, unhealthy eating. Mobile devices that are integrated into people's everyday lives make it possible for us to get a closer look into behavior. Two of the most commonly used sensing modalities include Ecological Momentary Assessments (EMAs): surveys about mental states, environment, and other factors, and wearable sensors that are used to capture high frequency contextual and physiological signals. One of the main visions of mobile health (mHealth) is sensor-based behavior modification. Contextual data collected from participants is typically used to train a risk prediction model for adverse events such as smoking, which can then be used to inform intervention design. However, there are several choices in an mHealth study such as the demographics of the participants in the study, the type of sensors used, the questions included in the EMA. This results in two technical challenges to using machine learning models effectively across mHealth studies. The first is the problem of domain shift where the data distribution varies across studies. This would result in models trained on one study to have sub-optimal performance on a different study. Domain shift is common in wearable sensor data since there are several sources of variability such as sensor design, the placement of the sensor on the body, demographics of the users, etc. The second challenge is that of covariate-space shift where the input-space changes across datasets. This is common across EMA datasets since questions can vary based on the study. This thesis studies the problem of covariate-space shift and domain shift in mHealth data. First, I study the problem of domain shift caused by differences in the sensor type and placement in ECG and PPG signals. I propose a self-supervised learning based domain adaptation method that captures the physiological structure of these signals to improve transfer performance of predictive models. Second, I present a method to find a common input representation irrespective of the fine-grained questions in EMA datasets to overcome the problem of covariate-space shift. The next challenge to the deployment of ML models in health is explainability. I explore the problem of bridging the gap between explainability methods and domain experts and present a method to generate plausible, relevant, and convincing explanations.

CHAPTER 1
INTRODUCTION

Chronic conditions in the United States account for close to 75% of the total healthcare spending and over 66% of the total deaths [1]. Chronic conditions such as heart failure and diabetes are among the leading causes of death and healthcare costs. A chronic condition is ‘a physical or health condition that lasts more than a year and causes functional restrictions or requires ongoing monitoring or treatment’ [2]. Patients who have been hospitalized for certain conditions such as heart failure have a higher risk of rehospitalization and mortality [3]. Chronic diseases are particularly difficult to tackle since often multiple diseases occur together and worsen the patient’s outcome [4], factors such as low socioeconomic status [5], ethnicity and race [6] are associated with increased risk. *This makes it critical to identify risk factors, and manage chronic conditions.*

In order to manage chronic conditions, appropriate medication, and patient monitoring are essential. While clinical care makes up one aspect of health, *human behavior is a key facet of health conditions*. Recent estimates attribute 10-20% of health outcomes to medical care, 30% to genetics, 20% to the social and physical environment and a *major portion **(40-50%)** to behavior [7].* Modifiable behavioral risk factors such as tobacco, poor diet, physical inactivity, and alcohol consumption are the leading causes of death in the United States [8].

Sensor driven behavior modification is one of the main visions of mobile health (mHealth) [9]. mHealth technology makes it possible to continuously sense a person’s physical, physiological, and psychological states [10]. These can be used to develop models that predict the risk of an adverse event (such as smoking, drinking, etc.) [11, 12]. The predictive models can then inform the development of targeted interventions for behavior modification. Interventions delivered through this technology can be used to target a variety of applications such as smoking cessation [13], physical activity [14], stress management [15]. There are currently over 300,000 mHealth applications on popular app stores, and this number is expected to rise [16], with increased adoption driven by improved mobile computational power and miniaturization of sensors and devices.

Two of the most commonly used sensing technologies in mHealth include Ecological momentary assessments (EMAs), and physiological wearable sensors. In Ecological Momentary Assessments (EMAs), users answer survey questions about their mental state, behaviors, and other contextual factors by completing smartphone questionnaires, typically multiple times per day [17]. EMAs enable the collection of rich contextual data in real-time during the ecologically-valid settings of daily life and can inform the delivery of real-time mobile interventions under field conditions. Wearable sensors on the other hand can be used to monitor a user's physiology. Two commonly measured physiological signals include ECG and PPG [18, 19, 20, 21]. The information from these sensors is utilized to continuously sense a user's psychological, environmental and physiological states. Machine learning techniques can be used to model and predict adverse outcomes using the data collected by these sensors.

One characteristic of mHealth studies however is that they are conducted separately, typically on small number of participants. There are several choices to be made by behavioral scientists while designing the study such as the demographics of the study participants, the different emotion items and questions included in the EMA, the type of sensor used for the study, etc. This would result in independently collected datasets having different properties from each other. For example, a research goal while conducting a behavioral study might be to study a different emotion item, that was not included in the previous study. Or a research goal might be to conduct the study on a different demographic population. Similarly, there are several choices while identifying the sensor used in a study - sometimes outside the control of the researchers designing the study. We might want to use a new sensor that is more accurate, or we might have to use a different sensor because the previously used sensor is out of production, etc. Our goal is to support health researchers to make these choices based on the study constraints while utilizing the different datasets to develop machine learning models with good generalizability.

The differences in the context in which different mHealth datasets are collected results in two main shift issues that are problematic to machine learning practitioners: 1) Covariate-space shift, 2) Domain shift. Covariate-space shift is a scenario when the input space of two datasets are different, commonly occuring in EMA data. Each EMA study design will often employ a unique set of EMA items in order to measure different constructs, with the result that *few EMA datasets will share a common EMA dataspace.* This is a fundamental problem that hinders the use of a model trained on one study with data from a different study. Further, we cannot pool data from multiple studies to benefit from all the data available. This is also a substantial barrier to the use of standard domain adaptation methods [22], as they employ a shared encoder architecture to achieve alignment of domains, which requires the input spaces to be identical (e.g. as with two different populations of RGB images). The second is the standard problem of domain shift or covariate shift, where the data distribution of two studies might be different. For example physiological data collected in two different studies using sensors which differ in their design, location on the body they are used, etc. could result in data with different properties, which leads to poor generalization of machine learning models across these datasets. *It is critical to address both, covariate-space shift and domain shift across mHealth datasets to develop models with good transfer performance across these datasets.*

The goal of our work is to develop methods to overcome covariate-space shift and covariate shift across mHealth datasets. Specifically, we want to develop a method to address covariate-space shift across different EMA datasets. Further, we want to study the problem of domain shift in pulsative physiological signals and develop a domain adaptation method for ECG and PPG signals.

There is quite extensive prior work on survey data integration in the in the field of statistics. This corresponds to maximizing the information that can be obtained from survey data. This is a recent review paper [23]. The works studying combining samples from different surveys are often considering that there is a *common set of features* in both surveys. One of the two samples contains additional features. This is treated as a problem where the additional feature is missing in one of the surveys and a model can be constructed based on the survey with complete data [24, 25, 26, 27]. However, this is different from our problem where the set of EMA questions (features) can be completely different across the two datasets. Our aim is to find a common representation that can be used to train an machine learning model on one dataset and use it on the other EMA dataset. To the best of our knowledge, this is the first work to do this in the machine learning context of finding a common input representation such that predictive models can be transferred across EMA datasets.

There is some recent work on domain adaptation on ECG such as [28, 29, 30] however, they do not study domain shift caused by real sources of variability such as sensor variation, population, etc. The domains are obtained by randomly splitting one dataset. The work on lab to field generalization of cocaine use prediction [31] is one work which studies the performance of a predictive model across two datasets which are collected separately. However, the main source of domain shift between these two datasets is prior shift or class imbalance. Our work is one of the first to study domain shift in pulsative physiological signals (ECG, PPG) where the sources of variation are due to differences in the sensor type, and population.

While accounting for different sources of variability and developing generalizable models for mHealth is essential, the next step in the deployment of machine learning models for health is addressing the black-box nature of models. Explainability is a key property that machine learning solutions must possess if they are to be employed in clinical settings [32]. In this work, we perform the first exploration of utilizing counterfactuals for explanation and the properties they must have to be useful for explaining predictive models in health.

1.1 Thesis statement

Self-supervised tasks leveraging the physiological properties of pulsative signals are an effective method to minimize domain shift.

1.2 Overview

In this thesis, we study the two problems of covariate-space shift and domain shift in mHealth. We explore the topic of model explainability, which is essential for predictive models in health to be implemented. The thesis is organized into three main topics: 1. Studying covariate-space shift in EMA datasets with the task of non-response prediction (chapter 2), 2. Domain adaptation through self-supervised tasks for pulsative physiological signals (chapter 3), 3. Bridging the gap between predictive models and physicians through counterfactuals (chapter 4).

1.2.1 Transformers for EMA non-response prediction under covariate-space shift

Ecological Momentary Assessments (EMAs) are surveys in which users answer questions about their mental states, behavior, environment, and other contextual factors typically on a smartphone [17]. EMAs provide rich contextual information about the user in real-time and hence are used in a number of behavioral studies [33, 34, 35] and as a clinical research tool [36, 37, 38, 39]. EMAs designed for different studies capture a different set of features. The design of EMA questions depends on the research aim of the mHealth study. Behavioral researchers who conduct these studies often tend to vary questions between different studies to study the effect of different emotions and contextual factors. This is potentially problematic for using machine learning models with all the available data since the feature space is not constant across the studies. We cannot pool data from different studies to benefit from all the EMA data availabale. Further, a model trained on one EMA study cannot be directly used with data from a different study. A straightforward solution of collecting all features (super-set of the features in different studies) from the different study populations would impose a substantial burden on the mHealth researchers and participants. Hence, our aim is to support the flexibility of mHealth researchers to freely design EMA surveys, while addressing the problem of covariate-space shift methodologically. In this chapter, we use EMA data from two separately conducted smoking cessation studies: Breakfree, and CARE. Breakfree is a smoking cessation study conducted on a population of 300 African American smokers, and CARE is a smoking cessation study conducted on 400 African Americal, Mexican American and White American smokers. To address covariate-space shift, we require a common input representation across the datasets. We require a machine learning task with which we evaluate the predictive performance of common input representation as compared to using a different data representation for each dataset. We predict non-response to EMA prompts using the history of responses as the machine learning task. Predicting non-response is an important task in EMAs since *non-response is a major challenge in EMA data collection.* Prior work on predicting non-response has focused on identifying predictors using classical machine learning methods. Given the variety of factors that could affect non-response, a data-driven feature representation for prediction could be attractive. The goal of this chapter is to develop a deep model for non-response prediction that is transferrable to a new EMA dataset (e.g., a new study with a different set of questions).

1.2.2 Domain adaptation through self-supervised learning for pulsative physiological signals

In addition to sensing psychological states with EMA, mHealth devices make it possible to continuously sense users physiology through wearable sensors. This is particularly exciting since complex health factors such as vital signs, and adverse events can be monitored and predicted and this can be used to deliver timely interventions. For example, the fall detection feature in Apple watch which can call emergency services is the beginning of the many potential life-saving use cases of mHealth. One potential hurdle to the progress of machine learning research in utilizing physiological data is the variability across different studies. For example, the mHealth study on stress detection in [35] used a sensor suite consisting of ECG electrodes and an RIP sensor described [18]. Whereas, the study [21] used a commercially available chest and wrist band to detect stress events. There are several sources such as the population, the placement of the sensor on the body, individual differences, and sensor design that could result in differences in the properties of data collected in different mHealth studies, known as domain shift. Domain shift is a phenomenon where the distribution of the data used for training and testing are different [40]. The problem of domain shift is extensively studied in the field of computer vision [41, 40, 42, 43], where the sources of variability such as lighting, resolution, camera type are known to a certain extent. On the other hand, there is far less prior work studying the problem on domain shift in the case of pulsative physiological signals [28, 29, 30]. However, none of these study domain shift in a scenario with a real source of variability. The goal of our work is to study differences across domains of pulsative physiological signals and develop a method for domain adaptation that utilizes the properties of these signals. We explore the problem of domain shift due to changing the position of the ECG electrode, which results in a change in the direction of measurement of the electrical activity. Our ECG experiments are performed on the PTBXL [44] 12-lead ECG dataset with arrhythmia classification from ECG as the main machine learning task. In the case of PPG, we perform domain adaptation across the MIMIC [45] and WESAD [21] PPG dataset. Here the sources of variation are the demographics (ICU patients vs mHealth study participants), the sensor type and location on the body (hospital grade fingertip PPG sensor vs wrist watch PPG sensor).

1.2.3 CFVAE: Counterfactual VAE for explainable ranking of patients for home hospital care

Explainability is a key property that machine learning solutions must possess in order to be employed in healthcare settings. Typically, machine learning research in healthcare aims to assist domain experts (behavioral reserachers, physicians) with models to make decisions such as the time to provide an mHealth intervention, discharge a patient, etc. These are decisions that are usually made by domain experts however, machine learning models can be helpful in reducing the burden on them and can be used to scale to a larger number of patients. *Explainability is critical for such a model since it has to be convincing to the domain expert in order to be deployed.* There is extensive prior work on exaplainability in AI, comprehensive surveys on the topic include [46, 47, 48, 49]. In this chapter, we focus on explainability via counterfactual (CF) generation [50, 51, 52], where the determining features of the classification model are highlighted by comparison to a diverse set of other similar (synthetic) patients to whom the classifier would assign an opposite label. It was inspired by observing our medical collaborators express clinical judgement in terms of hypothetical trends in vital signs, e.g., "This patient is a good candidate to be sent out of ICU, but if their systolic blood pressure had been 110 and falling then they would have needed a vasopressor." We rank patients based on their suitability to be sent to home hospital [53] for the machine learning decision making task. Home hospital is a program in which medical capabilities that would usually be provided in a hospital are brought to the patient's home. It is attractive to both patients and healthcare providers, and is likely to be more relevant as the demand for hospital beds continues to grow. Our goal in this chapter is to develop a model to rank patients based on how suitable they are to be sent to home hospital and explain the decision to a physician. Our approach to generating CFs focuses on the CF being plausible and relevant to the healthcare task, not previously found in prior CF generation approaches. We develop a counterfactual generation methodology to explain the learned machine learning model. Our solution is a variational autoencoder (VAE)-based approach where the latent space is trained to capture the decision boundary of the machine learning model and sample from the counterfactual distribution.

1.3 Contributions

This dissertation makes the following contributions:

Valence features to address covariate-space shift in EMA datasets:

- We develop a state-of-the-art transformer model for predicting EMA non-response using the history of responses.
- We demonstrate the feasibility of constructing valence features to overcome covariate-space shift, which are common across different EMA studies while preserving their predictive power.
- The valence features enable the use of standard domain adaptation methods across different EMA studies, which was not possible when each study had a different input representation.

Domain adaptation through self-supervised learning for pulsative physiological signals:

- This is the first work to study the problem of domain shift with different sources of variation such as population, sensor design, etc. on pulsative physiological signals (ECG, PPG).
- This is the first work to use self-supervised learning for domain adaptation on physiological signals. We design self-supervised tasks based on physiological properties of the signal to capture invariance across domains.
- Our self-supervised method of domain adaptation outperform standard domain adaptation methods on ECG and PPG data.
- Our self-supervised method enables models to be adapted without access to data from the target domain at training time.

CFVAE: Counterfactual VAE for explainable ranking of patients for home hospital care

- This is the first work to develop a model for ranking patients for home hospital care. We frame this problem on the publicly available MIMIC dataset.
- We develop CFVAE: a VAE based feedforward method to produce plausible, relevant and sparsely perturbed counterfactuals (CF) to explain a given prediction model.
- Our method outperformes prior CF generation methods based on a quantitative evaluation of their plausibility outperforms prior methods based on a qualitative evaluation of plausibility by a physician.

CHAPTER 2

TRANSFORMERS FOR ECOLOGICAL MOMENTARY ASSESSMENT NON-RESPONSE PREDICTION UNDER COVARIATE-SPACE SHIFT

2.1 Introduction

In this chapter we describe our approach to address covariate-space shift across EMA data collected from different studies. This work is described in detail in [54].

Mobile health (mHealth) technology is a promising tool for health behavior change and maintenance with a broad array of applications, including smoking cessation [13], physical activity [14], stress management [15] and medication adherence [55]. mHealth data sources such as wearable sensors, self-reports, GPS, etc. provide key insights into the contextual and behavioral factors that influence health outcomes, through the ability to collect data from participants in real-time in the field environment. A particularly valuable source of data comes from ecological momentary assessments (EMAs), in which participants answer questions about their mental state, behaviors, and other factors by completing questionnaires, typically multiple times per day. EMA data provides unique insights which are difficult to glean from other sensing modalities, and is widely-used as a result. It can be used to assess the risk of adverse behaviors, trigger interventions, or estimate treatment effects.

In addition, EMAs are a widely-used research tool in clinical psychology and psychiatry [36, 56, 57]. They have also been used in a diverse set of additional fields including cardiology [37, 58, 38], diabetes [59, 60, 61], chronic pain [62, 63, 64], reproductive health [39, 65], medication adherence [66, 55], and in mental health and neurological conditions such as epilepsy [67, 68], schizophrenia [69, 70, 71], and depression [72, 73, 74].

A major challenge in EMA data collection is *participant non-response*, which arises when users fail to complete a survey when prompted. Non-response results in the loss of EMA data, and is problematic for three reasons. First, it reduces the statistical power of hypothesis testing based on EMA data. Second, if the non-response is systematic, then it is likely to be missing not at random (MNAR), a form of bias which is difficult to correct for. Third, missing EMA samples make it more challenging to assess time-varying contextual variables such as emotions, environments, and behaviors. Note that EMA non-response is related to the problems of medication non-compliance [55, 66] and non-response in electronic patient-reported-outcomes (ePROs) as used in clinical trials [75, 76].[1] Hence, a capability for predicting EMA non-response can play a critical role in improving the quality and effectiveness of EMA-based data collection in a broad range of clinical research domains.

A model that takes as input a sequence of outcomes to past EMA prompts and predicts the risk of non-response in real-time could be a powerful tool for improving the response rate of a study. It could support the adaptation of EMA prompt times and enable the delivery of interventions to improve the response rate. However, this requires the solution to two key challenges. The first is the standard problem of *covariate shift* [78, 22, 42]: Two study populations responding to the same EMA items (questions) may have different distributions of answers based on demographics, study design, and other properties. Second, and substantially more challenging, is the problem of *covariate-space shift*: Each EMA study design will often employ a unique set of EMA items in order to measure different constructs, with the result that *few EMA datasets will share a common EMA dataspace.* This is a substantial barrier to the use of standard domain adaptation methods such as DeepCORAL [22], as they employ a shared encoder architecture to achieve alignment of domains, which requires the input spaces to be identical (e.g. as with two different populations of RGB images).

The goal of this work is to develop a deep model for non-response prediction which is

[1] Non-response is also related to the problem of patient dropout [77], but is different in that participants remain engaged in the study, but fail to provide the full complement of data.

transferrable to a novel EMA data domain (e.g. a new study with a different set of items) without finetuning, thereby providing a generalizable solution to non-response prediction. Such a capability could be used to design and trigger interventions to improve compliance. Our solution has two components: 1) A novel valence feature construction approach which exploits the unique properties of EMA data to address the covariate-space shift problem; and 2) A transformer [79] architecture for non-response prediction, including an investigation of the effectiveness of positional encoding for EMA data analysis. This is the first comprehensive work to address EMA non-response prediction using deep learning with a solution to covariate-space shift.

Prior work on predicting non-response [80, 81] has focused on identifying effective predictors using classical machine learning methods. There are two main advantages of using deep models: 1.) Deep models learn a *data-driven feature representation* and do not rely on hand-crafted features. Given the variety of factors that can contribute to non-response,, and the difficulty of engineering general-purpose features by hand, a data-driven feature learning approach is attractive. 2.) *Domain adaptation and transfer learning* methods can be used to learn from multiple small datasets collected independently (typically the case in mHealth studies). However, only a few prior works have used deep learning (DL) for EMA data modeling [82, 83], in contrast to other mHealth data types such as accelerometry [84, 85, 86, 87], and no prior works have used DL for non-response prediction. Recently, transformers [79] have emerged as a powerful new class of tools for modeling sequential observations. Following their initial success in NLP [88, 89], transformers have proven effective in several other fields [90] [91] [92] [93]. Sequences of EMA observations differ significantly from NLP in that the arrival times are *irregularly-sampled* and important to model (e.g. EMA responses which are closer together in time are more likely to be correlated).

The transformer architecture initially introduced for sequence modeling tasks in NLP [88] forgoes recurrence and uses attention mechanism [79] to learn temporal pattern in the data.

The scaled dot product form of attention involves computing pair-wise attention values via matrix multiplication operations, which can be computed efficiently and easily interpreted. We are motivated by the success of the transformer on a wide range of data types and tasks [90, 91, 92, 93], as well as by its interpretability and scalability [79]. Transformers can be pre-trained in an unsupervised manner to learn good data representations and then be used for fine-tuning downstream tasks. This is particularly useful when we do not have sufficient labeled data for our downstream task of interest. Transformer models have shown state-of-the-art performance in several other fields such as computer vision [90] [91] [92], speech [93], and multi-modal tasks. *This is the first work to develop transformer models for EMA data analysis in general, and non-response prediction in particular.* Through this work, we aim to show that there is potential for deep models (and specifically transformers) to learn patterns in EMA data. With increasing number of mHealth datasets being collected, we anticipate a growing increase in DL (and specifically transformer) tools for EMA analysis.

Sequences of EMA observations differ significantly from natural language in that the arrival times are *irregularly-sampled* and contain useful information (e.g., EMA responses which are closer together in time are more likely to be correlated). In this work, we address the following questions in EMA data modeling: 1) How can the irregular temporal structure of EMA data be captured in a transformer model? 2) What pre-training tasks are the most effective in reducing the need for extensive training data? 3) How are EMA responses encoded in the learned feature representation?

2.2 Contributions

This chapter makes the following contributions:

- This is the first work to explore the utility of transformer models for sequences of EMA response data. We present a transformer architecture for predicting non-response to EMA prompts using the EMA response history and the design decisions that yield the most effective transformer architecture. The transformer model achieves an AUC of 0.77 (for EMA sequence length = 15) for predicting future non-response and is substantially more accurate than both classical ML models and DL models based on the LSTM architecture (particularly for longer input sequences.)

- We visualize the learned self-attention weights and observe that the model learns meaningful feature interactions that are consistent with findings in [94], which is a behavioral study on the factors affecting EMA compliance among adolescent smokers.
- We present the design decisions that yield the most effective transformers for EMA sequence analysis.
- We design a self-supervised pre-training task and demonstrate that pre-training yields a modest performance gain.
- We evaluate the transfer performance of our valence feature representation designed to overcome covariate-space shift when models are trained and tested on different EMA datasets, and demonstrate encouraging performance.
- The valence feature representation enables us to utilize domain adaptation techniques across different EMA datasets. We evaluate the utility of DeepCORAL, a popular domain adaptation method and find that it provides a boost in the generalization performance of our transformer model. We will make the code and trained weights of our model freely-available, enabling future research on EMA analysis to utilize transformers and begin with an effective feature representation.

2.3 Related work

There are three bodies of prior work which are most closely-related to our work: 1) Analysis and prediction of EMA non-response, as well as the related topics of interruptability and availability; 2) Use of deep learning models to analyze EMA data; and 3) Transformer models for electronic health record (EHR) data, which shares with our task the need to model irregularly-sampled data. We discuss each of these topics in detail.

2.3.1 Analyzing and predicting EMA non-response

A significant body of prior work analyzes the factors that are related to non-response to EMA prompts. [94, 95] identify the factors that have a significant effect on non-response (which they term compliance, adherence, engagement). [96, 97] study EMA response rates and determine the feasibility of using EMA as a research tool based on the response rates. [96] further underscores the importance of differentiating between human factors and factors related to technology in non-response while reporting response rates (which they refer to as adherence level). Two recent review papers on EMA non-response, [98] and [62], provide additional evidence for the importance of the problem. [62] reviews studies involving patients with chronic pain, while [98] reviews studies related to substance abuse. In contrast to the current study, these prior works do not address the development of a *predictive model* for non-response to EMA.

Two recent works [81, 80] have focused on *predicting* participant non-response. Both works use contextual factors (such as location, activity, etc.) in a predictive model. One common factor among all these prior works is their use of classical machine learning models for analyzing and predicting EMA non-response. We share with these prior works an investigation into the predictive power of various contextual factors and mental states (e.g. emotions). At the same time, our work is uniquely-distinguished by its focus on developing transformer models for non-response prediction in order to exploit the benefits of feature learning in modeling complex sequential data.

The tasks of assessing interruptibility and availability in mHealth are related to our problem of non-response prediction. A representative example of availability modeling is Sarkar et. al. [99], which developed a classifier that combined mobile sensor data with past

EMA responses to classify whether or not a participant is available at the current moment in time. The topic of interruptibility has been widely-explored in the context of intelligent notification systems [100, 101, 102, 103]. The goal of these works is to design a system that delivers notifications at opportune moments based on contextual factors. The focus of [101] is the optimization of the user experience. [103] presents a reinforcement learning based method for scheduling notifications. This is similar to the study of receptivity to mHealth interventions in [104, 105], where the goal is to determine opportune moments using contextual factors (such as activity, location, phone battery, etc.). The topics of availability, receptivity, and interruptibility prediction are critically important for avoiding unnecessary participant burden and considering external contextual factors in determining availability. In contrast, our focus is on developing a predictive model for non-response based on feature learning derived from factors such as participant mental states and emotions, and their history of EMA responses.

An additional related topic is participant disengagement, which manifests as a steady decline over time in the participation of a user in a study or treatment program [106, 107], often resulting in loss to follow-up [108]. The focus of these works is on longitudinal analyses and long-term study outcomes. In contrast, our focus is on quantifying the short-term risk for non-response at the EMA prompt level. Such a capability could inform the design of interventions to maximize the utility of EMA as a measurement tool, which is distinct from the important task of improving long-term participant engagement.

A final related topic is in the domain of ePROs (electronic patient-reported outcomes). ePROs are patient-provided information about symptoms, side effects, drug timing and other questions during a clinical trial [75]. ePROs generally lack the momentary, frequent sampling found in our EMA dataset. The extension of our work to developing transformer models for sequences of ePRO data is an interesting avenue for future work.

2.3.2 Deep models for EMA data

There are two prior works that develop deep models for prediction tasks using EMA data [82, 83]. In [82], the authors propose a recurrent neural network (RNN) for forecasting depressed mood using the history of EMA data. In [83], the focus is on predicting short term mood developments from EMA data using an RNN. In addition, there are numerous works that analyze EMA data using classical statistical and machine learning tools, such as logistic regression and SVMs [109, 110, 111, 112, 113]. The current article differs from these prior works in two ways. First, we address the problem of predicting whether the next prompt will result in an EMA response, which is distinct from the task of predicting the responses themselves, as in the case of predicting self-reported mood. Second, we develop a *transformer model for EMA* sequences and analyze its utility for predicting EMA non-response. Transformer models have been shown to deliver state of the art results in fields such as NLP [79][88] and computer vision. We extend this class of models to the EMA setting.

We note that there has been significant work on using DL models to analyze clinical data such as Electronic Health Records (EHR), a domain with some similarity to EMA analysis. While EHR data is diverse, it includes categorical variables that capture clinical states, which is analogous to EMA response data. Two representative works that use classical sequential DL models for EHR analysis are [114, 115]. Both works use an attention layer with a recurrent temporal model (an RNN) for EHR sequence analysis. In contrast, our focus is on the exploration of transformer-style models for irregularly-sampled EMA data.

2.3.3 Transformers for Electronic Health Records Data

Based on the success of transformer models on NLP tasks [79][88][116], recent works have explored their application to a broad range of other domains, including the analysis of EHR data. EHR analysis includes several prediction tasks, such as length of stay, mortality, and sepsis onset, which share our focus on predictive modeling from irregularly-sampled data.

One representative work is [117], which applies transformer models to irregularly sampled clinical data. In [118], a BERT-style model is developed using a pre-training task that is appropriate for irregularly sampled diagnosis codes.

There are several significant differences between EHR and EMA analysis. First, EHR datasets consist primarily of categorical observations (e.g. diagnostic codes) and real-valued biomarker measurements, while EMA data consists primarily of ordinal vectors. Second, in EHR datasets only a subset of possible observations are available at any point in time, whereas for EMA it tends to be all or nothing (participants either respond to the prompt and answer all of the items or fail to respond at all). Third, EHR data contains many more variables and data item types in comparison to EMA. Given these differences, it is unlikely that findings from modeling EHR data will transfer in any significant way to EMA data analysis.

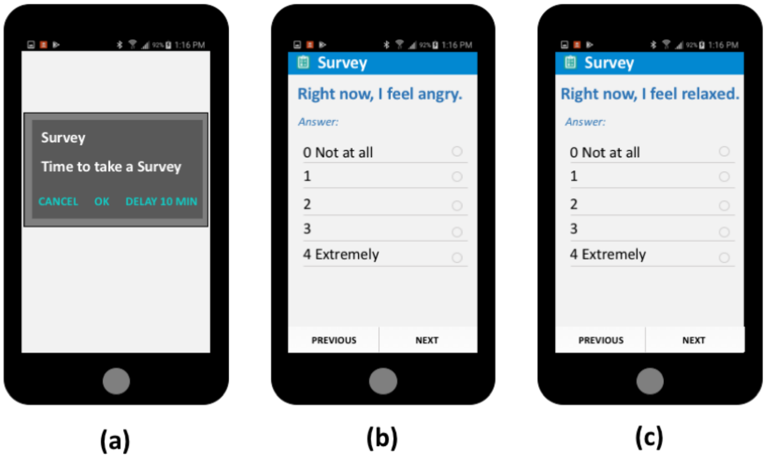
2.4 Study protocol and dataset

Our primary dataset comes from a study that examines the influence of intrapersonal and contextual factors on smoking lapse among African American smokers. Data was collected from multiple modalities including EMA prompts, on body sensors, and location from GPS. The study participants carried a smartphone provided to them with the study software installed. The mobile app delivered EMA prompts and collected real time continuous data in the participant's natural environment from multiple sensors. Data was processed in real time on the smartphone and machine learning algorithms were used to extract biomarkers corresponding to specific behavioral and physiological indicators of smoking and stress. In this analysis, we focus on the EMA data, as this provides a rich set of items that capture aspects of contextual and mental state, and is also the most widely-collected datatype in health applications.

EMA collection process: The study participants carried a smartphone provided to them with the study software installed to deliver EMA prompts. In order to begin and end triggering EMAs for the day, participants had to press a button indicating start and end of day. Participants were prompted by the phone app to complete three types of Ecological Momentary Assessments (EMAs) on their smartphones during the study - random EMA, stress triggered EMA, smoking triggered EMA. On each day, a participant was prompted with an average of four random EMAs. After the day start button was pressed, the day was divided into 4 equal blocks of time. In each block, the phone app checked for the 'participant availability,' determined by the battery level (being above 10%), whether the participant was driving, and if the participant had enabled a 'do not disturb' option. The 'do not disturb' mode could be used by participants to stop receiving any EMA prompts when they were unavailable. The data collected from the sensors was used to determine smoking events and events of stress. In case of these events, the phone app checked for the same conditions for firing an EMA and triggered a smoking EMA or stress EMA. *In our work, we are interested in predicting non-response to the random EMAs.*

Figure Figure 2.1 shows the interface for an EMA notification and the UI while responding to some example survey questions. Once a notification was triggered, the participant could either: 1. Accept the notification and begin answering the survey by clicking 'OK', 2. Dismiss the notification by clicking 'Cancel', 3. Snooze the notification and receive it again after 10 minutes.

The dataset consists a total of 255 participants, after excluding participants who dropped out of the study. The participants range between age 20 to 82 (mean 51 ± 12 years) and we have a roughly balanced split between the male and female subjects. The data collection process spanned two contiguous weeks (4 days pre-smoking-cessation through 10 days post-smoking-cessation). Over the course of the study a total of 9043 random EMAs were triggered and 5636 of them were completed (average compliance rate of 62.8%).


Survey

Time to take a Survey

Survey

Right now, I feel angry.

Answer:

0 Not at all

1

2

3

4 Extremely

Survey

Right now, I feel relaxed.

Answer:

0 Not at all

1

2

3

4 Extremely

Figure 2.1: (a) EMA notification on the study phone (b) Survey question: angry (c) Survey question: relaxed.

2.5 Methodology

2.5.1 Non-response problem framing

Consider a set of n participants indexed as $i = 1, 2, \cdots, n$. Each participant then has EMAs (observations) indexed by $j = 1, 2, \cdots, n_i$, where n_i is the number of observations (EMAs) for participant i. We design a model to use a sequence of T EMAs as input and predict if the $(T + 1)^{th}$ EMA is completed.

We frame this as a binary classification problem where our label is

$$Y_j = \begin{cases} 1 & \text{if } j^{th} \text{ EMA is completed} \\ 0 & \text{if } j^{th} \text{ EMA is missed} \end{cases}$$

The feature vector derived from the j^{th} EMA is denoted as X_j. Section subsection 2.5.2 describes the process to create X_j from the EMA data. Here, we assume that X_j is a feature vector consisting of K features.

We developed models under two scenarios. In the first, we used a sequence length of one, meaning that for each EMA we predict the compliance for the next EMA prompt. In the second scenario, we used a sliding window (of length T) approach as shown in

Figure 2.2 to compute the feature sequence and the corresponding binary label for classification. In this case, we use a transformer architecture to perform feature learning, and predict next EMA compliance from a sequence of feature vectors. For our experiments, we use a sequence length T = 5, 10, 15, and 25.

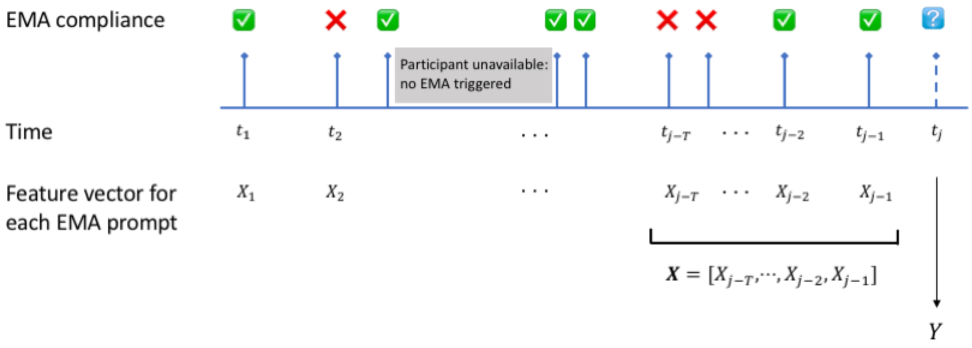


Figure 2.2: Sliding window approach used to create our feature vector sequence and classification label. EMAs are prompted irregularly in time, feature vectors from a sequence of T EMAs are used to predict compliance to the $(T+1)^{\text{th}}$ EMA.

2.5.2 Dataset specific feature construction:

We use a set of raw features derived directly from the EMA response along with meta-data logged as part of the study ('Features' in Table 2.1). In addition to these, we construct features to summarize the history of the raw features ('Summary features' in Table 2.1). There are two types of summary features constructed, they are designed to: 1) Capture the completion history summary (long term and short term); 2) Capture the variance in the positive and negative affect, and completion pattern.

The long-term completion rate is the average number of EMAs completed by a participant from the beginning of the study up to the current EMA and the short-term rate is the average number of EMAs completed the previous day. The long-term and short-term completion rate features are designed to capture the 'trait' and 'state' aspects of the participant's behavior.

$$\text{Long-term completion rate (CR)} = \begin{cases} \frac{\sum_{i=1}^{j} Y_i}{j} & \text{if } j \neq 0 \\ 0 & \text{if } j = 0 \end{cases}$$

$$\text{Short-term completion rate (CR)} = \begin{cases} \frac{\text{\#EMA completed on day } (d-1)}{n_{d-1}} & \text{if } n_{d-1} \neq 0 \\ 0 & \text{if } n_{d-1} = 0 \end{cases}$$

where d is the day the current EMA is triggered, n_{d-1} is the total number of EMAs triggered on day $d - 1$.

The variance feature is computed for the positive and negative affect and smoking urge. The variance feature for each covariate for the j^{th} EMA is computed as the variance of the covariate until the j^{th} EMA. For example, the variance of the positive affect 'Happy' computed for the j^{th} EMA is the variance in response to the question 'Happy' for EMA 1 to EMA j.

Table 2.1: Features used in our analysis - we include both raw features and summary features. The raw features directly obtained from the EMA response. We compute additional summary features from the history of the raw features. We perform an ablation without using 'Summary features' in Table Table 2.3 to assess the importance of history.

Type	Raw features	Value	Summary features
Positive affect	Enthusiastic Happy Relaxed	Likert scale (1-5)	Variance of each item (until current EMA)
Negative affect	Bored Sad Angry Restless Urge	Likert scale (1-5)	Variance of each item (until current EMA)
Compliance	Current EMA status	Binary	Long term CR Short term CR

2.5.3 Valence feature construction

To overcome the covariate-space shift between different EMA datasets, we need a common feature representation that can be constructed irrespective of the fine-grained EMA questions. We design a feature representation that captures the average positive and negative affect. The average of the responses to all positive emotion items in an EMA response is computed as the average positive response and similarly, the average negative response is computed. The raw features for each EMA prompt consists of the average positive response, average negative response, and the compliance. Similar to the approach described in the previous subsection, we construct summary features to capture completion history summary and variance of positive and negative responses. We call this the valence feature representation.

2.5.4 Transformer model

Background

The transformer is a sequence modeling architecture based entirely on attention proposed in [79]. A self-attention mechanism is a mapping between pairs of words in a sentence/input points in a sequence to the output. The scaled dot product attention mechanism introduced in [79] is computed as

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right) V$$

where Q, K, V are the query, key, and value matrices computed as a projection of the input sequence X into query, key and value spaces. $Q = XW^Q, K = XW^K, V = XW^V$. The matrix multiplication $QK^\top$ computes pairwise inner products between every query and key vector pair. The value vector is weighted by this attention matrix.

Multihead attention performs the attention mechanism described above in h different feature spaces, where h is the number of heads. The attention is computed on the key, query, value matrices projected with h different learned projections and concatenated together.

$$\text{Multihead}(Q, K, V) = \text{Concat}(head_1, head_2, \cdots head_h)W^o$$

where $head_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$ and W^o projects the concatenated output back to the original size.

Two initial operations are performed on a sentence prior to computing attention:

1. Input embedding: learned embeddings are used to convert words to vectors of dimension d_{model}.

2. Positional encoding: since the transformer model does not contain any form of recurrence, information about the position of different words is added to the input representation. Sine and cosine embeddings are computed as shown below and added to the input representation. Here pos corresponds to the position of a word in a sentence.

$$PE_{(pos,2i)} = sin(pos/1000^{2i/d_{model}})$$
$$PE_{(pos,2i+1)} = cos(pos/1000^{2i/d_{model}})$$

EMA Transformer

There are two main differences between a sequence of words and a sequence of EMA responses: 1. EMA responses are ordinal and responses are already in a vector form, 2. Continuous time associated with EMA responses: e.g., a sequence of 4 EMAs could have been completed at 10 AM, 11.30 AM, 3 PM, 4 PM respectively. We account for these two differences architecturally in this manner:

Input embedding: The feature vector computed for each EMA is used directly as the input embedding. For a sequence of T EMAs, we represent the input embeddings as

$X_1, X_2, \cdots X_T$.

Positional/time encoding: Since transformer models do not contain recurrence, they do not have a natural mechanism for representing the ordering of an input sequence. To address this, positional information is injected into the input sequence through positional encoding. In sentence modeling tasks (NLP), only discrete positional information is encoded and combined with the input embedding. The standard positional encoding technique is to compute an embedding of the positions using the sine-cosine representation described in [79] represented as $PE(pos)$ which is then *added* to the input embedding. In the case of EMAs, a sequence of T EMAs has a sequence of discrete positions (i.e., $1, 2, \cdots T$) and continuous times (i.e., $t_1, t_2, \cdots t_T$) associated with it. We evaluate the performance of encoding the temporal vs positional information into the input embedding. Given a sequence of EMA input embeddings X, position pos, time t, positional/temporal encoding function $PE(\cdot)$, we explore *two ways to encode temporal/positional information* into the EMA input embeddings and compute the input to the attention mechanism.

1. Addition: The embeddings from the position/time values are computed using the sine and cosine functions described in Section subsubsection 2.5.4. Given a sequence of EMA input embeddings X, position pos, time t, the input to multihead attention when encoding the discrete position and continuous time values respectively are:

$$\mathrm{X}_{inp}^{\text{pos}} = X + PE(pos)$$
$$\mathrm{X}_{inp}^{\text{time}} = X + PE(t)$$

2. Concatenation: The embeddings from the position/time values are computed and *concatenated* with the input embeddings. Given a sequence of EMA input embeddings X, position pos, time t, the input to multihead attention when encoding the

discrete position and continuous time values respectively are:

$$\mathrm{X}_{inp}^{\text{pos}} = \begin{bmatrix} X \\ PE(pos) \end{bmatrix}$$

$$\mathrm{X}_{inp}^{\text{time}} = \begin{bmatrix} X \\ PE(t) \end{bmatrix}$$

where $\begin{bmatrix} X \\ PE(t) \end{bmatrix} = \begin{bmatrix} X_1 \\ PE(t_1) \end{bmatrix}, \begin{bmatrix} X_2 \\ PE(t_2) \end{bmatrix}, \cdots, \begin{bmatrix} X_N \\ PE(t_N) \end{bmatrix}$

Where X_{inp} is the input to multihead attention after encoding the continuous time embeddings.

The model architecture is the standard transformer [79] encoder architecture followed by a linear layer for the classification output. See Figure Figure 2.3 for the model architecture. Our architecture consists of 6 encoder layers, 8 attention heads per encoder layer. The dimension of the key, query and value vectors is 64.

2.5.5 Self-supervised pre-training of EMA transformer

The goal of pre-training a transformer model in a self-supervised manner is so that it can learn the structure in the data, which can be useful for other downstream tasks. In EMA, we might be interested in some particular prediction problem like predicting the probability of a person drinking alcohol. Since we are limited by the amount of labeled data, pre-training aims to leverage self-supervised learning from a large corpus of EMA data. If we have a large EMA corpus, we can train a model that can learn the structure between different EMA items and the temporal structure in the data. The model can then be fine-tuned for the prediction task that we are interested in. In this work, we are evaluating the utility of self-supervised pre-training of transformers with EMA data. Given the expense of data

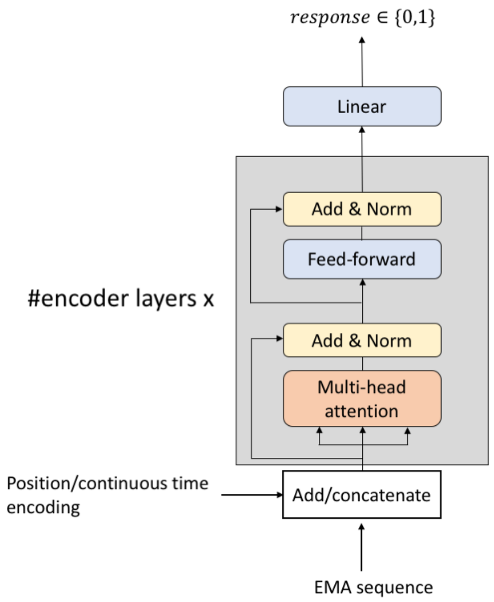


Figure 2.3: EMA Transformer model architecture. This includes a transformer encoder followed by a linear layer.

labelling in healthcare and the findings in NLP from BERT [88], we envision that self-supervised pre-training might be an attractive strategy.

Background

[88] introduced BERT, which is designed to pre-train bidirectional representations from a large corpus of text in a self-supervised manner. This is done by first pre-training BERT (a bidirectional transformer model) on two tasks: 1. Masked language modeling (MLM), 2. Next sentence prediction (NSP). In the MLM task, some words in the input sentence are replaced by a MASK token. *The model is trained to impute these words correctly*. In the NSP task, a pair of sentences is provided as the input and the model is trained to recognize if the second sentence is a valid 'next sentence'. The exact description of the pre-training can be found in [88].

The idea behind pre-training BERT in this manner is to learn the structure in language: structure of words within a sentence (MLM) and structure at the sentence level (NSP) without having any specialized labels. Once it has learned the structure, a pre-trained

BERT can be used with an additional linear layer for other downstream tasks.

Pre-training: EMA transformer

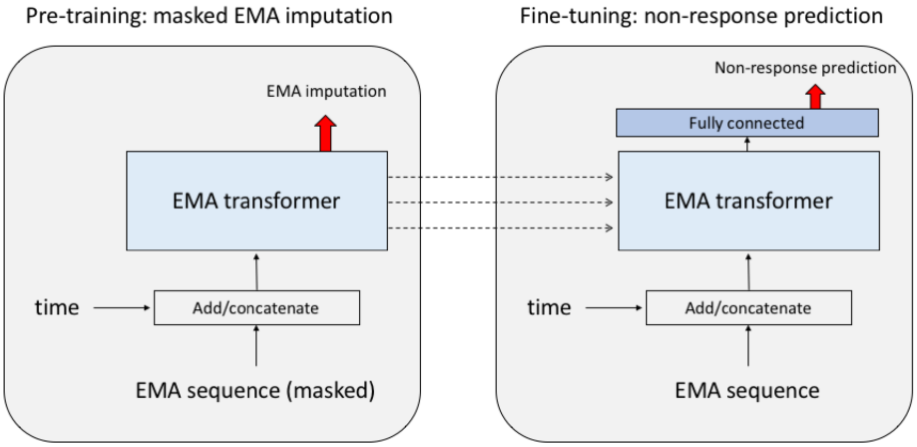


Figure 2.4: Pre-training the EMA transformer model with an EMA imputation task. The pre-trained model is then fine-tuned for the non-response prediction task. This pre-training strategy is similar to BERT where the model is pre-trained in a self-supervised manner on a large text corpus and the model is fine-tuned for downstream tasks.

We design a masked EMA imputation task, similar to the masked language modeling (MLM) pre-training task in BERT. Given the EMA features of a T length EMA sequence $X_1, X_2, \cdots X_T$, a subset of features are masked at random time points. Note that the values masked are always the response to emotion items. We do not mask out the compliance history results for the masked imputation task. The number of emotion items masked is pre-determined by us and the positions where the response is masked is chosen at random. We mask out the emotion item(s) in 15% of the sequence, determined randomly. The goal of the masked EMA imputation task is to mask out responses to some emotion items in the sequence, and learn to reconstruct the response to these items. This will help the model learn the structure between the different emotion items and their temporal pattern. Note that the masked imputation task is for pre-training purposes only. The input sequence does not contain mask tokens during a downstream fine-tuning task. To account for this, after a particular position is chosen for masking, we replace the input with the mask token 80% of the time. The input value is retained as is 10% of the time and changed to a random value 10% of the time similar to MLM in [88].

Several pre-training tasks are possible, based on the choice of the emotion items that we mask out. The choices range from (a) masking out a single emotion item to (b) masking out all emotion items. These are illustrated schematically in Figure Figure 2.5. Once we pre-train an EMA transformer model, we add a linear layer to it and fine-tune it for the non-response prediction task.

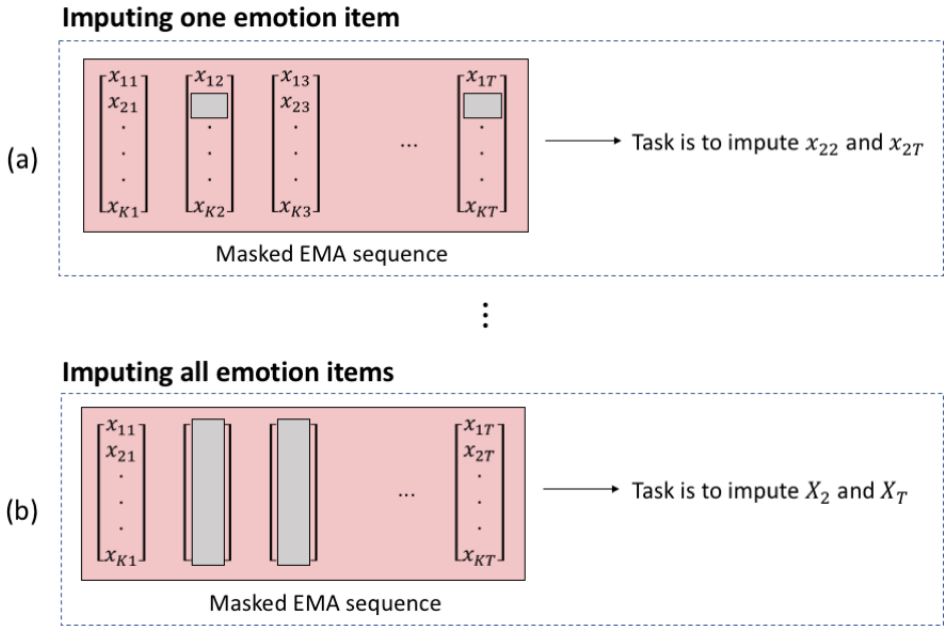

Imputing one emotion item
(a)
x_{11} x_{21} x_{K1}
x_{12} x_{K2}
x_{13} x_{23} x_{K3}
...
x_{1T} x_{KT}
Task is to impute x_{22} and x_{2T}
Masked EMA sequence
Imputing all emotion items
(b)
x_{11} x_{21} x_{K1}
...
x_{1T} x_{2T} x_{KT}
Task is to impute X_2 and X_T
Masked EMA sequence

Figure 2.5: Self-supervised EMA masked imputation tasks. Here we assume that there are K emotion items in each EMA. The input sequence here is depicting *only* the responses to emotion items. We do not mask out other features in the input sequence. (a) The first task shown is to impute one emotion item at a time in 15% of the sequence positions that are randomly masked. For example, the value of the emotion ‘Happy’ can be masked off in some positions of the sequence and the task is to impute this value correctly. Note that we explore imputing each emotion item one at a time as a pre-training task and evaluate the downstream non-response prediction performance. *There are intermediate tasks possible such as masking out 3 emotion items, 4 emotion items, etc.* (b) The second task is to impute all emotion items in 15% of the sequence.

2.5.6 Input representation for zero shot transfer to other EMA datasets

While zero shot transfer is a standard way to measure generalizability in domains like NLP, it is challenging in EMA because different studies use different EMA items (questions), resulting in different input feature spaces. We explore an approach to summarize diverse EMA items based on their valence (positive or negative) as a means to create a novel and generalizable feature representation. For example, In our main dataset, we can compute the average response to Enthusiastic, Happy, Relaxed as positive affect and that of Bored, Sad, Angry as negative affect. This provides a standard input representation that can be used across different EMA studies.

2.6 Experimental results

We present findings in three areas. The first relates to the choice of input representation. While some applications benefit from pre-trained embeddings (e.g. word embeddings in the case of NLP), we find that the fixed length EMA response vector itself is an effective input representation. This has the advantage that the self-attention weights can be easily interpreted as weights on the individual EMA items. Our visualizations of the learned transformer representation suggest that it encodes structure in the EMA response data which is meaningful for non-response prediction. The second finding relates to positional encoding [79], which adds a vector to each input embedding that provides a global encoding of the position of each token in the input sequence. We find that positional encoding improves performance, but concatenating the temporal information is more effective than adding it. The third finding has to do with pre-training. The BERT architecture for NLP tasks [88] demonstrated the effectiveness of pre-training a transformer-based model on a large unlabeled corpus prior to fine-tuning it with labels on a smaller task-specific training dataset. We designed a self-supervised pre-training task based on EMA imputation. We found that pre-training produced a small performance benefit which was not statistically-significant.

We hypothesize that this approach may be more effective in the future as larger EMA datasets become available. We present visualizations of the learned transformer representation that suggest that it encodes structure in the EMA response data which is meaningful for non-response prediction.

We consider two non temporal models - Logistic regression, Support Vector Machine (SVM) to compare the performance of deep models (LSTM and transformers) against the methods used in prior work for predicting EMA non-response [80] [81]. Both these prior works use SVM as the prediction model. We also compare the performance of our transformer model to two LSTM based architectures (vanilla LSTM and attention LSTM proposed in [115]). All the results reported are 5 fold cross-subject validation where we use one set of subjects for training and a held-out set of subjects for validation.

2.6.1 Predicting non-response to the next EMA using the current EMA response

We evaluate the prediction performance of non-temporal models when the sequence length $T = 1$. We present results under two scenarios: 1. using only the raw features, 2. including the summary features. The results presented in Table Table 2.3 indicate that capturing the summary of *previous* EMA responses significantly improves the prediction performance.

2.6.2 Predicting non-response to the next EMA using a sequence of EMA responses

We explore the utility of learning representations to summarize the EMA history with deep temporal models such as Transformers and LSTMs. We perform prediction using a sequence of T EMAs where sequence length $T = 5, 10, 15, 25$. The performance of different models is reported in Table Table 2.4. We see that the deep models show an improvement over logistic regression. The transformer model (with self-supervised pre-training) performs the best and particularly shows an improvement over LSTM in modeling long sequences (T = 15, 25). Note: here we show results from the pre-training task of imputing one emotion item that performs the best. We show results from different pre-training tasks in Table Table 2.2. We see that the performance is similar across different pre-training tasks. We discuss this more in Section section 2.7.

Table 2.2: Cross validation AUC for predicting non-response to next EMA prompt using a sequence of $T = 15$ EMAs with different pre-training tasks. The different pre-training tasks here are: 1. Imputing one emotion item at a time, 2. Imputing all the emotion items, 3. Imputing five emotion items. We do not see a large change in performance with a change in the pre-training task.

Pre-training task	T = 15
One item	0.76 ± 0.01
All items	0.75 ± 0.02
Imputing 5 items	0.76 ± 0.02

Table 2.3: Average 5 fold cross validation AUC for predicting next EMA response using features from the current EMA (sequence length T = 1). We use two sets of features: 1) raw features only, 2) raw features and summary features. Using summary features improves performance significantly, indicating that history is important in predicting response.

Dataset	Model	Raw features only	Raw features & summary features
Dataset A	Logistic regression	0.63 ± 0.02	0.71 ± 0.02
	SVM (RBF kernel)	0.64 ± 0.02	0.71 ± 0.02
Dataset B	Logistic regression	0.63 ± 0.01	0.70 ± 0.02
	SVM (RBF kernel)	0.63 ± 0.02	0.68 ± 0.02

Table 2.4: Average 5 fold cross validation AUC for predicting non-response to next EMA using a sequence of T EMAs. The deep models LSTM and Transformer perform significantly better than Logistic regression. Among the deep models, the EMA transformer with pre-training performs the best. The transformer performance is significantly better than LSTM when using longer sequences ($T = 15, 25$). All of these results are 5 fold cross validation results on the main dataset. Note that here we use dataset specific features for prediction.

Dataset	Model	T = 5	T = 10	T = 15	T = 25
Dataset A	Logistic regression	0.70 ± 0.03	0.66 ± 0.02	0.65 ± 0.02	0.58 ± 0.02
	Vanilla LSTM	0.73 ± 0.02	0.73 ± 0.02	0.72 ± 0.02	0.71 ± 0.01
	EMA transfomer	$\mathbf{0.75 \pm 0.02}$	0.76 ± 0.01	0.76 ± 0.01	$\mathbf{0.75 \pm 0.01}$
	EMA transformer (with S-S pre-training)	$\mathbf{0.75 \pm 0.01}$	$\mathbf{0.77 \pm 0.01}$	$\mathbf{0.77 \pm 0.01}$	$\mathbf{0.75 \pm 0.02}$
Dataset B	Logistic regression	0.66 ± 0.02	0.62 ± 0.02	0.63 ± 0.01	0.58 ± 0.04
	Vanilla LSTM	0.74 ± 0.03	0.74 ± 0.03	0.72 ± 0.04	0.73 ± 0.03
	EMA transfomer	0.77 ± 0.02	0.75 ± 0.01	0.76 ± 0.02	$\mathbf{0.73 \pm 0.02}$
	EMA transformer (with S-S pre-training)	$\mathbf{0.78 \pm 0.02}$	$\mathbf{0.78 \pm 0.02}$	$\mathbf{0.77 \pm 0.03}$	0.72 ± 0.02

2.6.3 Role of temporal/positional encoding:

We explored two kinds of incorporating positional information with the standard sine-cosine function for encoding: 1. Addition, 2. Concatenation. We evaluate the performance of EMA transformer when performing temporal/positional encoding in these two ways. Figure Figure 2.6 (a) shows the error of the masked prediction task with the 4 different strategies (lower is better) (b) shows the AUC of the downstream non-response prediction task with a sequence of T EMAs. In all of these cases, we see that *concatenation* results in better performance than the standard *addition*. We discuss this in more detail in Section section 2.7.

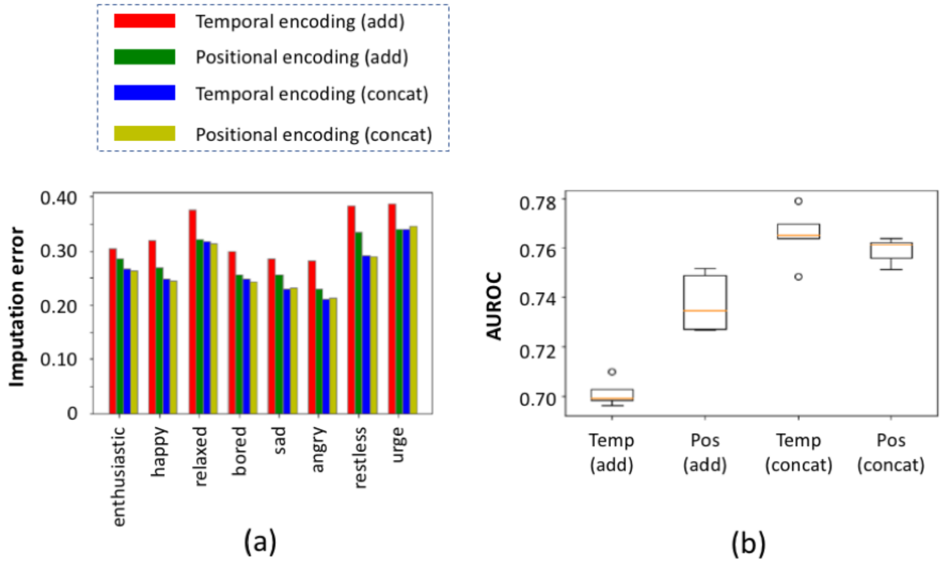


Figure 2.6: Performance of transformer with different temporal/positional encoding (a) On the pre-training task of imputing EMA responses (here we impute the features one at a time); (b) On the downstream response prediction task with sequence length $T = 15$. The box plot shows performance across different cross-validation folds. *Concatenation* performs the best in all these cases and *temporal concatenation* performs the best in the prediction task.

2.6.4 Predictive performance of valence features

We construct valence features as described in the previous section to overcome the covariance-shift problem across different EMA datasets. We evaluate the predictive performance using the valence features and compare them to using the entire set of features in the rows 'using valence features' and 'using dataset specific features' in Table Table 2.5. The performance using valence features is slightly lower than the performance with the full set of emotion items (dataset specific features). *Hence, the valence feature representation serves as a common feature representation that can be computed across any EMA dataset with only a slight performance degradation.*

2.6.5 Generalization to other EMA studies

We evaluate the performance of the EMA transformer model trained one dataset and evaluate its performance on the other dataset. Note that we are able to use the model trained on dataset A at test time on dataset B using the valence feature representation. We report results in Table Table 2.5 in the row 'Transfer performance'. We see that while the performance is slightly lower than what is obtained when trained on the main dataset, these results show that the model learns representations that generalize to new datasets without additional fine-tuning. The transfer performance is evaluated by training the dataset on one dataset (e.g., dataset A) and testing it on the other dataset (e.g., dataset B). Note that there are several variations across these datasets such as difference in population, context of the study, etc. which could result in domain shift. We utilize a popular domain adaptation technique (DeepCORAL [22]) to align the learned representations across the two datasets. We see that domain adaptation provides a boost in transfer performance without the need for additional fine-tuning on the target dataset.

Table 2.5: Average 5 fold cross validation AUC for predicting non-response to next EMA using a sequence of T EMAs in the following settings: a) Using all the emotion items in each dataset, b) Using the valence feature representation, c) Transfer performance (e.g., model trained on dataset B and tested on dataset A), d) Transfer performance after performing domain adaptation using DeepCoral [22]. We see that using the valence features results in a slight degradation in performance. However, valence features give us the ability to use the trained model on other datasets. We see that domain adaptation provides a gain in the transfer performance.

Dataset	Experiment	T = 5	T = 10	T = 15	T = 25
Dataset A	Using dataset specific features	0.75 ± 0.02	0.76 ± 0.01	0.76 ± 0.01	0.75 ± 0.01
	Using valence features	0.76 ± 0.02	0.70 ± 0.02	0.73 ± 0.03	0.72 ± 0.04
	Transfer performance	0.73 ± 0.01	0.70 ± 0.04	0.71 ± 0.04	0.66 ± 0.1
	Transfer performance (with domain adaptation)	0.73 ± 0.02	0.70 ± 0.02	0.72 ± 0.03	0.66 ± 0.02
Dataset B	Using dataset specific features	0.77 ± 0.02	0.75 ± 0.01	0.76 ± 0.02	0.73 ± 0.02
	Using valence features	0.76 ± 0.02	0.74 ± 0.03	0.73 ± 0.03	0.70 ± 0.04
	Transfer performance	0.64 ± 0.02	0.60 ± 0.04	0.58 ± 0.01	0.55 ± 0.1
	Transfer performance (with domain adaptation)	0.64 ± 0.01	0.65 ± 0.02	0.64 ± 0.01	0.60 ± 0.02

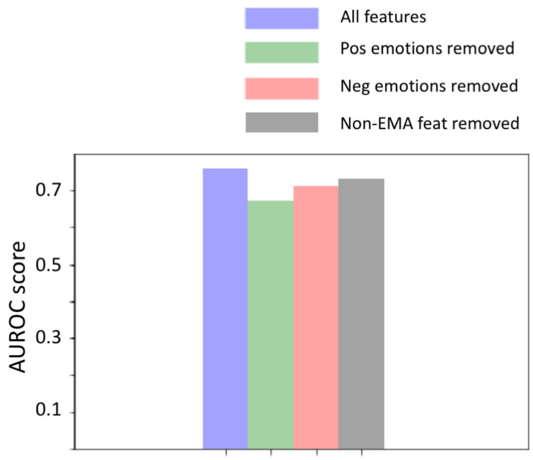


Figure 2.7: Feature ablation study: the performance of EMA transformer is evaluated in 4 settings: 1. using all features, 2. removing positive emotions, 3. removing negative emotions, 4. removing non-EMA features. We see the largest drop in performance when the positive emotion items are removed, indicating that the positive emotions have the most predictive power.

2.6.6 Feature ablation results

We perform a feature ablation study removing the positive, negative, and non-EMA response items and train an EMA transformer (See Figure Figure 2.7) . We see that the largest degradation in performance occurs when we remove the positive emotion items. This indicates that the positive emotions have the most predictive power. Analyzing the responses to EMA, we observe that participants generally report higher values (higher agreement) to positive questions and lower values to negative questions (see Figure Figure 2.8). One hypothesis for why positive emotions have higher predictive value is that if someone is strongly feeling a negative emotion, they might not respond and the emotion is not recorded. We see this in the data where negative emotions are generally not reported with high likert scale values.

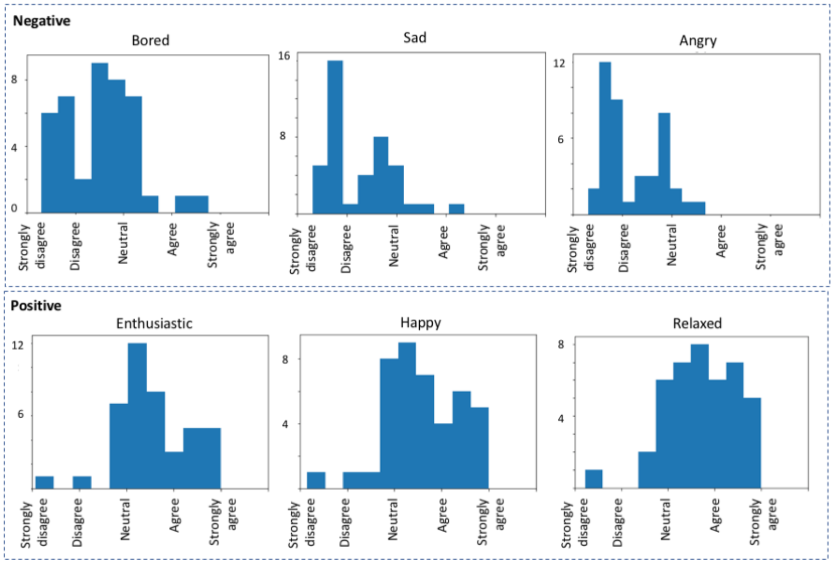


Figure 2.8: Distribution of responses to positive and negative questions. Notice that negative questions usually have a lower value response and positive questions have a higher value response.

2.7 Discussion

2.7.1 Importance of temporal history in non-response prediction

We evaluate the performance of classical statistical and machine learning models such as logistic regression and SVM (RBF kernel) for predicting next EMA compliance (compliance to $(T+1)^{th}$ EMA) using a single EMA (sequence length T = 1). Our feature vector summarizes the EMA history by using average completion and affect variance. We perform an experiment to *remove all summary features* computed from the EMA items. In this setting, we use only the information contained in one EMA response to predict next EMA compliance. This reduces performance substantially, as shown in Table Table 2.3 column 'Raw features only'. *These findings suggest that the EMA history summary is important for predicting next EMA compliance.* Next, we explicitly incorporate the history of EMAs into the prediction model by using a fixed sequence of EMAs (sequence of T EMAs) to predict next EMA ($(T+1)^{th}$ EMA) compliance. We develop a transformer based model for the task of forecasting using the sequence of EMA features. We see in table Table 2.4

that using a sequence of EMAs improves performance (in LR and SVM when compared to using T = 1). We see in table Table 2.4 that for the task of predicting compliance using T EMAs, the deep temporal models perform significantly better than the classical baseline. Among the models predicting compliance from the sequence of EMAs, the EMA transformer performs the best. *Hence, explicitly modeling the history of EMA responses is important for predicting compliance.*

2.7.2 Temporal/positional encoding for EMA input to transformers

We explore two ways to incorporate temporal (or positional) information into the transformer model: 1. Adding ; 2. Concatenating the positional embeddings. In the case of EMAs, data items are sampled at continuous time points, corresponding to the time when each EMA is triggered. Encoding the continuous time values into the input would provide a higher resolution of information than just the positions of each EMA. In the standard NLP literature [88], positional information is incorporated by *adding* sine and cosine functions of the position to the input. We believe that the *additive* positional encoding strategy (used in NLP) is sub-optimal for EMA (See Fig.Figure 2.6) data where we have much smaller datasets. The idea here being that adding temporal/positional information changes the input values and the model has to learn to differentiate the effect of position/time and the actual input variation. In the case of NLP where datasets are larger, we hypothesize that the model can learn this distinction better.

2.7.3 Self-supervised pre-training of EMA transformer

Pre-training transformer models in a self-suppervised manner has been shown to be beneficial in other domains such as NLP. The BERT model is pre-trained on a large corpus of text to learn structure in sentences. This model is then fine-tuned for new tasks where labeled datasets are of smaller sizes. Such a capability would be beneficial in fields such as healthcare where labeled datasets are difficult to collect. We see an improvement when using pre-training ('EMA transformer with S-S pre-training') over the results without pre-training ('EMA transformer') in Table Table 2.4. However, this isn't a statistically significant improvement. We hypothesize that this is due to the small size of our datasets. Table Table 2.2 presents the best non-response prediction task performance when the model is pre-trained with three different tasks: imputing one feature at a time, inputing all features, imputing a subset (5) features. We see that the performance is similar across the three pre-training tasks. One hypothesis is that the emotion items are related in some ways (e.g., if you are happy, you might also be feeling relaxed, etc.) and hence the representation to predict one feature might be useful for predicting multiple of them. The question of the utility pre-training with EMA sequences as in the NLP setting remains a question for future work. However, the preliminary results that we obtain show that pre-training might have the potential to result in significant improvements with larger datasets.

2.7.4 Interpreting the learned attention weights

The transformer self-attention matrix produces a weighted average of all the positions and features in the EMA sequence. Figure Figure 2.9 visualizes the attention weights across the EMA positions and features. Figure Figure 2.9(a) is a box plot of the attention across the EMA positions within a sequence (in the different transformer encoder layers). In this case, we are predicting EMA response with sequence length $T = 5$. We denote the EMA position for which response is being predicted as lag 0. The EMAs in the input sequence are denoted by lag 5 - lag 1. So lag 1 is the EMA immediately before the one being predicted. We see that lag 1 has the highest weight across all of the encoder layers. This is intuitive, as the most recent EMA is likely to be most closely-related to the participant's current mental state. We see an interesting pattern in the attention weights corresponding to the other lags. The mean of the lag 2, 3, and 4 weights are similar, but the weight corresponding to lag 5 is higher. One possible explanation for this trend is that since there are 4 EMAs on average per day, an EMA at lag 5 would correspond to the same time window as the current prediction task, but on the previous day. This may be capturing periodic behavior in the participant's daily routine that is relevant to their response or non-response. Explicitly handling periodicity in the model is an interesting topic for future work.

In Figure 2.9(b), we visualize the attention weights on the different features in the first encoder layer. This corresponds to the relative importance of different types of features in predicting non-response, averaged over an input sequence of length 5. We see that time is an important feature in predicting non-response. This is also aligned with our finding that the way we encode time in the input affects the performance of the transformer model. We also find that the emotion features *Enthusiastic*, *Angry*, and *Bored* have higher attention weights in comparison to other emotion items. This makes sense, as these are examples of strongly positive and negative emotions, which could influence response behavior, as well as the feeling of boredom which may correlate with a lack of engagement. These findings are consistent with those in [94], a behavioral study that finds that "higher mean negative affect" is a predictor of compliance in a study among adolescent smokers. The completion feature captures the detailed pattern of completion to each EMA in the sequence, while CR long and short are the average completion rate features that capture a summary of the pattern of completion. We see in Figure Figure 2.9(b) that the feature 'Completion' (capturing the detailed pattern of completion) is more useful in predicting non-response when compared to the long and short summaries of completion. This suggests that the model gets significant benefit from modeling the more detailed patterns in the response history. Collectively, these visualizations provide qualitative evidence that the transformer model is capable of learning meaningful structure from the sequence of EMA responses. *The ability to identify and visualize the feature interactions learned by the transformer can be a potentially useful capability for domain scientists who are interested in designing related interventions.*

2.7.5 Generalization to other EMA datasets

A critical challenge in utilizing predictive models developed on EMA studies is the problem of changing input-space due to the difference in the EMA questions. In this work, we provide a first solution by summarizing responses into positive/negative valence features. This enables us to use predictive models across EMA datasets (with a slight decrease in performance). We see that our transformer model performance is encouraging when tested on a different EMA dataset. Accounting for covariate shift across the datasets using DeepCORAL further improves the model performance. Developing a domain adaptation method that utilizes the structure of EMA data is an interesting future direction.

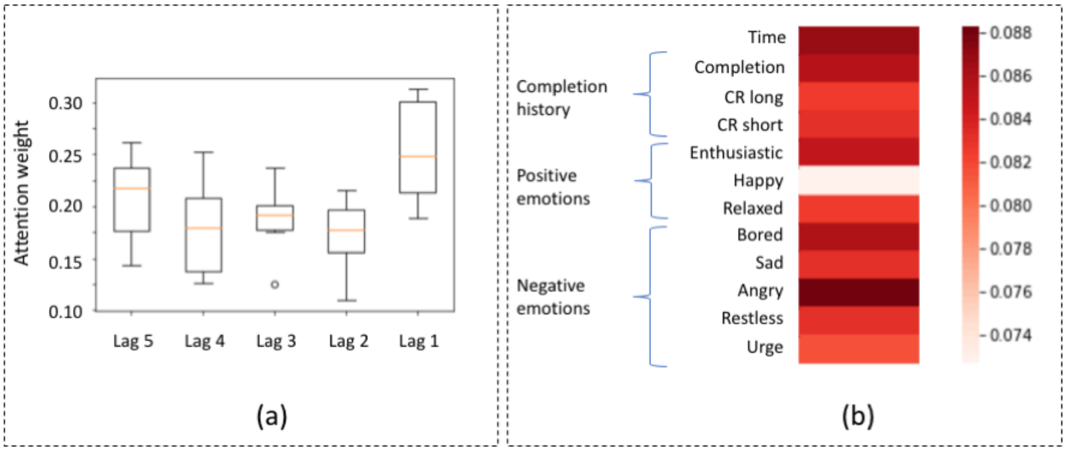


Figure 2.9: (a)Attention weight across the positions in an EMA sequence of length $T = 5$. Here the EMA positions in the sequence are denoted as Lag 5 - Lag 1 and the EMA for which response is being predicted is Lag 0. So Lag 1 is the EMA immediately before the one being predicted. (b)Attention weights on the different features in the input layer (first encoder layer) averaged across the time window. This visualizes the weighting performed by self-attention matrix on the input features.

2.8 Conclusion

In this work, we present a transformer architecture for modeling EMA sequences to predict non-response to future EMA prompts. Existing work on analyzing and predicting non-response have used classical machine learning models for this task. We are the first to explore the use of transformers for modeling sequences of irregularly sampled EMA responses and predicting non-response to future EMA prompts. We address three issues in this work: 1. Choice of the input representation for EMA sequences, 2. Designing a feature representation that can be used acros different EMA datasets irrespective of the fine-grained questions, 3. encoding the temporal information into the input, 4. analyzing the utility of self-supervised pre-training on EMA data for improving the non-response prediction task, 5. Utility of domain adaptation in improving the generalization performance of our non-response prediction model across different EMA datasets. We find that the transformer model achieves a classification AUC of 0.77 and outperforms both classical ML and LSTM based DL models. We find that the design choice for positional/temporal encoding affects the performance of the model and concatenating the temporal information leads to better performance when compared to the standard practice of adding the positional embedding. We design a self-supervised pre-training task on EMA sequences and find that it leads to an improvement (but not statistically significant). We present visualizations of the learned attention weights that illustrate the ability of the transformer to learn meaningful representations. An important future step will be using these prompt level compliance forecasts to inform the timing of compliance interventions.

CHAPTER 3
DOMAIN ADAPTATION THROUGH SELF-SUPERVISED LEARNING FOR PULSATIVE PHYSIOLOGICAL SIGNALS

3.1 Introduction

In the previous chapter, we developed a method to address the first challenge of covariate-space shift in the context of EMA non-response prediction. In this chapter, we study the second challenge, domain shift in the context of pulsative physiological signals collected in different studies.

The field of healthcare is ripe with opportunity for leveraging data-driven modeling. The complexity of tasks, various number of factors that could contribute to a prediction and the importance of accurate predictions makes deep feature learning desirable. Some works have shown promising results, achieving physician level performance [119, 120, 121] when trained in a supervised manner with large high quality labeled datasets. However, the utility of predictive models in health comes with their ability to transfer to other datasets and populations. Furthermore, large scale healthcare datasets are inaccessible due to privacy concerns (e.g., hospital data), participant burden (e.g., running long mHealth studies), cost of labeling datasets, etc. and typically we have access to small individually collected datasets.

Different datasets in mhealth are collected in different experimental contexts such as different sensors, different demographics, etc. This could result in a difference in the statistics of the data, called domain shift. Domain shift results in poor performance of models trained on one domain and tested on a different domain [40]. This results in two main challenges: 1) Utilizing multiple small datasets collected separately to train a model, 2) Generalization of a model trained on one mHealth dataset to the real world. *Hence, it is* *critical to account for the domain shift across mHealth datasets to realize the potential in making predictions and delivering interventions.*

One of the main sources of variability across mHealth studies is the hardware used. For example, the mHealth study on stress detection in [35] used a sensor suite consisting of ECG electrodes and an RIP sensor described [18]. Whereas, the study [21] used a commercially available chest and wrist band to detect stress events. In addition to this, there are differences in population, location of the study, etc. which could result in a domain shift across studies. Among the different sensors and data modalities used in mHealth [17, 122, 123, 18], high frequency pulsative physiological signals (such as the ECG and PPG) enable measurement of different kinds of health states through wearable sensors. The ECG records the electrical activity in the heart using electrodes placed on the body. The standard ECG measurement consists of 12 leads where each lead views the heart in a different direction [124]. As a result, the ECG signal recorded at each of these leads differs in structure from the other. See Figure 3.1 for an illustration of the ECG signal recorded at two leads on the chest. Notice that the structure of the ECG signal recorded changes substantially with a change in the sensor location. Similarly, the PPG signal records cardiac activity and is commonly used to estimate heart rate and oxygen saturation. The location of the PPG sensor on the body (Fingertip, wrist, etc.) has an effect on the signal. Similarly, the wavelength of light used in the PPG sensor determines the statistics of the data collected. Given that there are several kinds of wearable physiological sensors available commercially [125] and used in research [18], and the difficulty associated with obtaining labels for each individually collected dataset for training, *it is critical to use domain adaptation methods across physiological datasets and develop transferrable predictive models.*

The problem of domain adaptation has been extensively studied in computer vision [40, 41, 43, 42, 126, 127] where the different sources of variability across domains can be listed to some extent as differences in lighting, resolution, camera design, etc. The domain adaptation methods can be grouped intro three major categories: 1) minimizing a measure

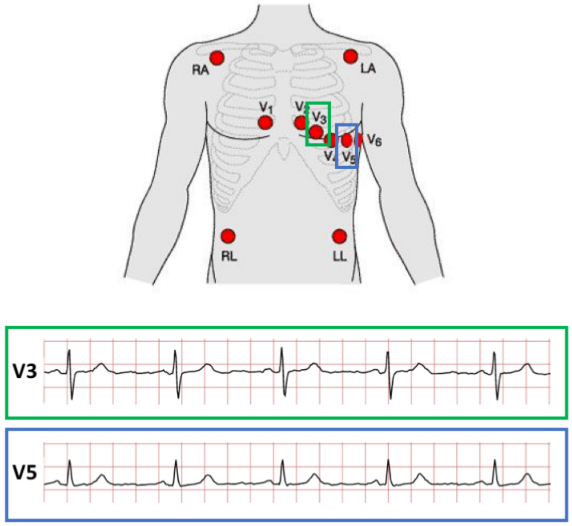


Figure 3.1: Standard 12 lead ECG set up. Each lead records the cardiac depolarization in a different direction. The signal recorded at two leads on the chest - V_3, V_5 are illustrated. Notice that the structure of the signal recorded at each lead is different.

of distributional discrepancy [41, 22], 2) adversarial training using a discriminator to align domains [42, 43], and 3) through self-supervised tasks [126, 127]. Domain adaptation through self-supervised learning has an advantage over the other two classes of methods that it can be used for improving model generalizability even without access to data from the target domain at training time [127].

Compared to this, there have been only a few prior works on domain adaptation on pulsative physiological signals such as [28, 29, 30, 128]. However, these works do not study domain shift caused by a real source of variation such as sensor design, etc. The domains in all these cases are obtained by randomly splitting one dataset. Additionally, these works use methods from the first and second category (metric based and adversarial training) for domain adaptation. The work on lab to field generalization of cocaine use prediction [31] is one work which studies the performance of a predictive model across two datasets which are collected separately. However, the main source of domain shift between these two datasets is prior shift or class imbalance. Our aim is to use study domain shift caused by variations in the sensor location, design, population, etc. on ECG and PPG signals. Additionally, we aim to propose self-supervised tasks for ECG and PPG signals that help with domain adaptation since these tasks can be used as a regularizer without requiring target domain data at training time.

The goal of this work is to study domain shift in pulsative physiological signals (ECG,PPG) caused by factors such as variation of population, sensor design, etc. and present a method for domain adaptation using self-supervised tasks. To study domain shift in ECG, we use the PTB-XL dataset [44] which is publicly available on Physionet [129]. PTB-XL is a clinical 12 lead ECG dataset. The leads capture ECG recorded in different directions from different positions on the body. We treat each lead as a separate ECG sensor (domain) and study domain shift across the different leads using arrhythmia classification as the main task. To study domain adaptation in PPG, our main task is estimation respiration rate (RR) from PPG data. We use data from the MIMIC III waveform database [45], and from an mHealth study WESAD [21]. MIMIC III is an ICU dataset where PPG data is collected from a hospital-grade fingertip PPG sensor and WESAD is an mHealth dataset where PPG data is collected using a commercially available wrist watch.

3.2 Contributions

In summary, this chapter makes the following contributions:

- This is the first work to study the problem of domain shift across pulsative physiological signals (ECG,PPG).
- This is the first work to use self-supervised learning for domain adaptation on physiological signals.
- Our self-supervised tasks result in a better transfer performance as compared to standard domain adaptation methods.

- Our method enables the use of self-supervised tasks for adaptation without access to target data during training time.

3.3 Related work

The problem of domain adaptation has been extensively studied in the field of vision, described in this review article [130]. Comparatively, domain adaptation in physiological signals is not well-studied. Recent prior work in ECG domain adaptation [28, 29, 30] exists. However, in all of these papers, the two domains are obtained by randomly splitting the MIT-BIH ECG dataset [131]. Hence these prior works are not studying domain shift caused due to changes that we would expect across different mHealth studies such as change in sensor design, placement of sensor, etc. Our work in this chapter studies domains shift in ECG caused due to change in the placement of ECG leads on the body, and domain shift in PPG caused by changing the sensor design (hospital grade fingertip sensor vs wrist worn sensor). Ours is the first work to study domain shift in pulsative physiological signals (ECG, PPG) across datasets collected in different contexts. A second difference is that the prior work on ECG domain adaptation [28, 29, 30] is based on adversarial domain adaptation methods. Domain adaptation has also been studied in the related area of accelerometry data [132, 133, 134]. Accelerometers are commonly used for action recognition and activity monitoring in mHealth studies. However, similar to the prior work in ECG, these methods apply standard domain adaptation methods to accelerometry data. In this chapter, we propose to use self-supervised tasks to learn domain agnostic representations which is superior since it does not involve a minimax optimization, and can be trained without access to target domain data [126, 127]. To the best of our knowledge, this is the first work to perform domain adaptation for self-supervised learning on time series data in general and pulsative physiological signals specifically.

3.4 Methods

To study domain shift across two datasets, we need a predictive model for our main task of interest. We then train the model on one dataset (source) and evaluate its performance on the other dataset (target). In this work we demonstrate our domain adaptation method on two pulsative physiological signals - ECG and PPG. In this section, we describe the main predictive task and the architecture used for each of these signals, followed by our method of self-supervised learning for domain adaptation.

3.4.1 Arrhythmia classification from ECG

We perform binary arrhythmia classification as the main predictive task on ECG. Given an 10second ECG signal, we perform classification where the two classes are normal and arrhythmia. The architecture used for arrhythmia classification from ECG consists of an LSTM encoder follower by fully connected layers illustrated in Figure 3.2. The model is trained using the crossentropy loss.

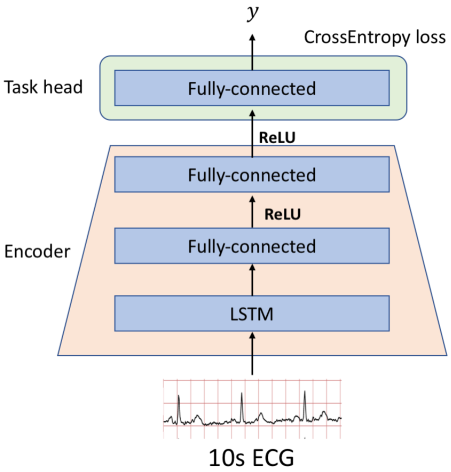


Figure 3.2: Architecture used for arrhythmia classification from ECG. This is our main predictive task on ECG signals to study domain shift across different leads. We train an arrhythmia classification model with this architecture on one domain and evaluate its performance on a different domain.

3.4.2 Respiration rate detection from PPG

Our main downstream task is respiration rate regression from PPG signals. The task is to detect respiration rate given a 60second length PPG signal. The architecture we use for this task is shown in Figure 3.3. The model is trained using the mean squared error loss.

Note that the architecture we use for both, the ECG arrhythmia task (Figure 3.2) and the PPG respiration rate task (Figure 3.3) consist of an encoder followed by a task head that consists of a fully-connected layer. The encoder/task-head structure is an important notation that we will refer to while describing our domain adaptation method.

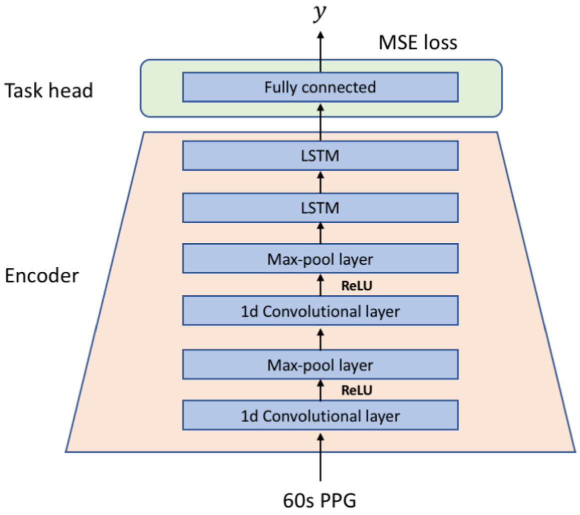


Figure 3.3: Architecture of the respiration rate detection model. This is our main predictive task on PPG signals to study domain shift. We train a respiration rate regression model on one domain and evaluate its performance on a different domain.

3.4.3 Domain adaptation through self-supervised tasks

The goal of domain adaptation is to align the source and target representations while maintaining task-specific discriminability. As described earlier, the architecture we use for the ECG/PPG tasks can be represented as an encoder followed by a task head. We utilize self-supervised tasks to perform domain alignment where the representations extracted by the encoder for the source and target domains are aligned. The idea is to simultaneously train the model to perform auxiliary self-supervised tasks on the source and target domain. Hence the tasks are parameterized by the same network for both the domains, aligning the representations.

General self-supervised tasks

We utilize three self-supervised tasks for the time series signals. For each of the source and target dataset, we create an augmented dataset for these three tasks.

1. Flip left-right: We randomly select 50% of the input signals and flip them along the horizontal direction. See!Figure 3.4 for an example of this operation performed on an ECG lead 1 signal. The auxiliary task is a binary classification problem to detect input signals that have been flipped.

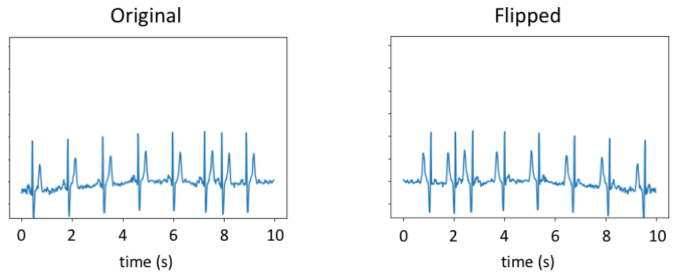


Figure 3.4: Example of flipping operation performed to create an augmented dataset for the self-supervised task. Here we show the original signal which is a Lead 1 ECG signal, and the flipped signal on the right. The auxiliary self-supervised task is to detect signals that have been flipped.

2. Mean shifting: We randomly select 50% of the input signals that are to be modified. For each of these, we add a random constant value over the entire time series to shift the mean. The auxiliary task is to detect inputs that have been mean shifted. See Figure 3.5 for an example of this operation performed on an ECG lead 1 signal.

3. Jumble segments: We select 50% of the input signals that will be jumbled along the time axis. For a given time series signal that is to be jumbled, we segment it into N segments (the number of segments is determined by randomly picking a number

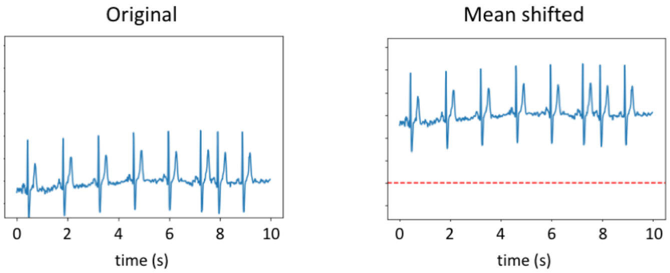


Figure 3.5: Example of mean shifting operation performed to create an augmented dataset for the self-supervised task of mean shifting prediction. The red dotted line indicates the mean of the original signal.

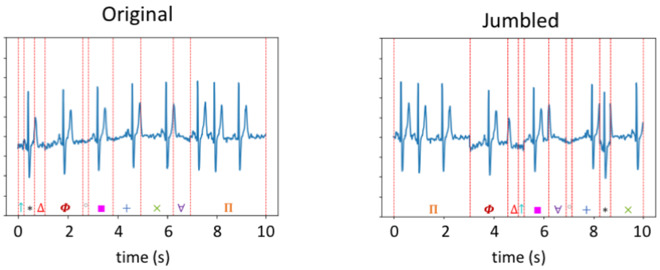


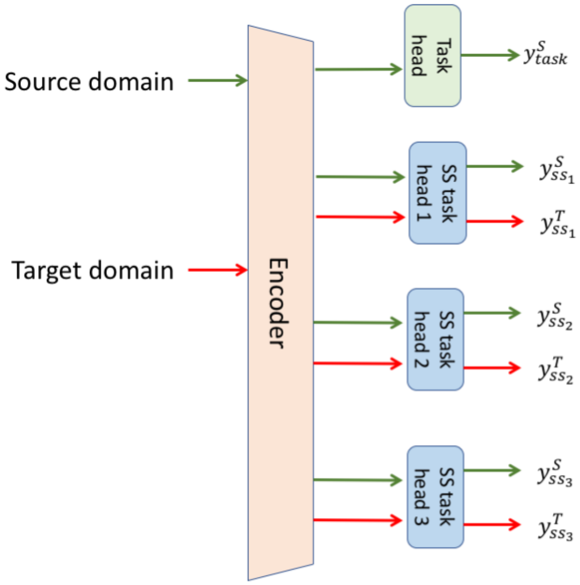
Figure 3.6: Example of jumbling operation performed to create a dataset for the auxiliary self-supervised task of detecting jumbled signals. Here we show the original signal which is randomly segmented into 10 segments. The segments are then permuted and combined to obtain the jumbled signal. Each segment is denoted by a unique symbol in this figure. Notice that the segments are ordered randomly on the right to result in the jumbled signal which no longer has the structure of a typical ECG signal.

between 30 - 40). The segments are then shuffled and the resulting time series is the ‘jumbled signal’. The auxiliary task is a binary classification to detect input signals that have been jumbled.

Once we have the auxiliary datasets for the self-supervised tasks for both the source and target domain, we train the model as shown in Figure 3.7. We use the task labels (arrhythmia labels in the case of ECG and respiration rate in the case of PPG) from the source domain to extract task-relevant representations.

Physiologically inspired self-supervised task

We design a self-supervised task of peak detection. In the case of ECG signals, the task is to detect R-peaks and in PPG signals, the task is to detect the main peak in the PPG wave.



$$
\begin{aligned}
loss = {} & CrossEntropy(y^S_{task}, label^S_{task}) \\
& + CrossEntropy(y^S_{ss_1}, label^S_{ss_1}) + CrossEntropy(y^T_{ss_1}, label^T_{ss_1}) \\
& + CrossEntropy(y^S_{ss_2}, label^S_{ss_2}) + CrossEntropy(y^T_{ss_2}, label^T_{ss_2}) \\
& + CrossEntropy(y^S_{ss_3}, label^S_{ss_3}) + CrossEntropy(y^T_{ss_3}, label^T_{ss_3})
\end{aligned}
$$

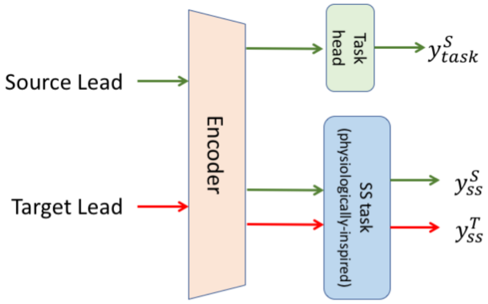
Figure 3.7: Domain adaptation through general self-supervised tasks. The goal here is to align source and target representations through auxiliary tasks while preserving the task-specific discriminability of the representations. We simultaneously train the network to perform three self-supervised tasks on the source and target domain. In addition, we train the task head using source domain labels. The three SS tasks we use are detecting: 1) Flip left-right, 2) jumbled segments, 3) mean shifted signals.

The peaks in physiological signals represent certain physiological events. For example, the R-peak in the ECG signal corresponds to ventricular depolarization. Different leads of the ECG record the same cardiac activity from different angles. Similarly, different types of PPG sensors are measuring the same activity. Hence the auxiliary task would be to detect the a physiological event given different kinds of measurements of the physiological signal. See Figure 3.8 for the overall training pipeline.

The ground-truth R-peaks for the source and target ECG inputs are obtained using a differentiation based peak detector in the neurokit toolbox. The ground truth PPG peaks are obtained using a differentiation based peak detector.

3.4.4 Finding an invariant input-representation

Pulsative signals are observing periodic cardiac activity in the body. In our case, the different ECG leads measure the electrical activity of the heart from different directions. The PPG datasets we consider measure the cardiac activity from two different locations on the body using different types of sensors. Finding an invariant input-representation across domains would help improve generalizability of predictive models across different domains. Since pulsative signals are measuring periodic cardiac activity in the body, the frequency domain would be able to capture this information. In our work, we include the short time fourier transform (STFT) of the input signal (ECG/PPG).



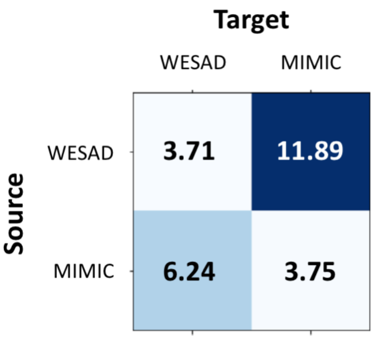
$$\begin{aligned} loss = {} & CrossEntropy(y_s, label_s) \\ & + MSELoss(y^S_{ss}, peak^S) \ + MSELoss(y^T_{ss}, peak^T) \end{aligned}$$

Figure 3.8: Domain adaptation through physiologically-inspired self-supervised task. Here the goal is to align source and target representations at the encoder while preserving task-specific discriminability. We simultaneously train the model to perform the main task with source labels and the auxiliary self-supervised task on the source and target domains. The auxiliary self-supervised task used is peak detection (R-peak in the case of ECG).

3.5 Experiments and results

3.5.1 Classification results without domain adaptation

We perform the binary arrhythmia classification task on data from Lead 1 - 6 in the PTB-XL datatset. We treat each lead as a different domain, which results in 36 scenarios where an arrhythmia classification model is trained on a lead which we call source lead and tested



	WESAD	MIMIC
WESAD	3.71	11.89
MIMIC	6.24	3.75

Figure 3.9: Root mean squared error (RMSE - lower is better) of the respiration rate (RR) detection from PPG. This figure denotes the error when the RR regression model is trained on the source (WESAD/MIMIC) dataset and tested on the target (WESAD/MIMIC) dataset. Notice that the performance is best when the source and target dataset are the same, depicted by the lower error values in the diagonal entries.

on a lead which we will call the target lead. The test AUROC score is shown in Figure 3.10. Notice that the diagonal elements are the highest for every lead (along the row and column), which is the scenario when there is no domain shift. We see that the performance drops substantially when the target leads are lead 3 and 4. These leads measure electrical activity in opposite directions as compared to the other leads and hence have the highest domain shift.

Similarly, we train a RR estimation model using PPG data from the two datasets MIMIC and WESAD and test it on data from each of these datasets. The root mean squared error (RMSE) of RR estimation is illustrated in Figure 3.9 where a lower value indicates better performance. We see that the diagonal elements where the source and target domain are the same, indicating no domain shift have the best performance. The error increases substantially in the off-diagonal elements where there is domain shift.

3.5.2 Using domain adaptation on ECG and PPG

We compare the performance of domain adaptation methods DeepCORAL [22], Adversarial discriminative domain adaptation (ADDA) [42] with our two self-supervised learning based methods: 1) general self-supervised tasks (SSDA) and 2) physiologically-inspired self-supervised tasks (Physiological SSDA).

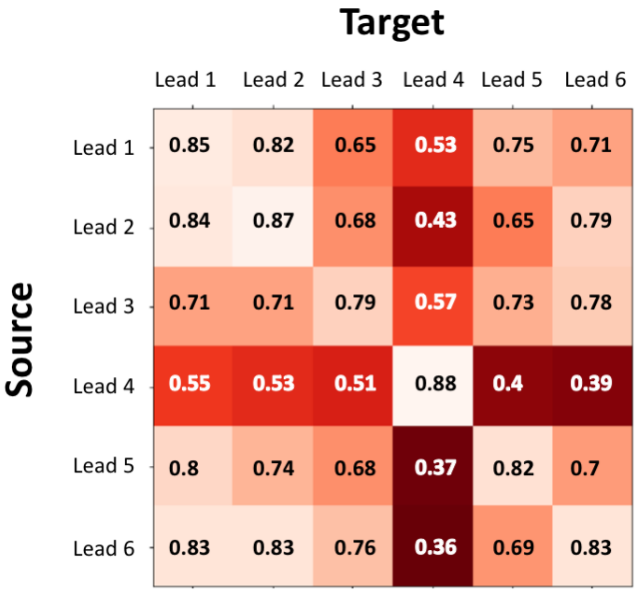


Figure 3.10: AUROC (higher is better) value for arrhythmia classification from ECG. Here we present the performance when the model is trained and tested on different leads (Lead 1-6 in PTBXL). There is no domain shift when the source and the target leads are the same. This corresponds to the high AUROC values observed in the diagonal elements. Notice that the performance is poor when the model is tested on Leads 3 and 4. These are the leads with the largest domain shift when compared to the other leads.

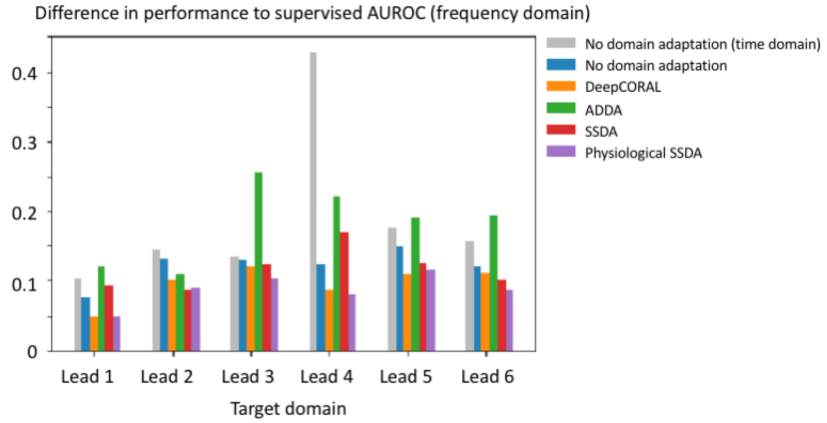


Figure 3.11: Difference in arrhythmia classification (from ECG) performance to supervised learning (lower is better) when both, the time and frequency domain is used. The supervised performance for each target lead is the corresponding diagonal entry in Figure 3.10. The results shown here reflect the average performance when all other leads are used as the source for a given target lead. Notice that using the frequency domain (even without any domain adaptation) yields an improvement over time domain. We see that physiological self-supervised domain adaptation (SSDA) performs the best

The performance of these different methods for ECG lead to lead domain adaptation are illustrated in Figure 3.11 (frequency and time domain data) and Figure 3.12 (only time domain data). Here, the bar plot for each target lead indicates the difference to supervised learning performance without any domain shift (corresponding diagonal element in Figure 3.10) averaged over all other leads as the source. Since we want to reach a performance on the target domain closer to the performance without domain shift, a lower value is better. We see here that the two self-supervised tasks outperform the other two domain adaptation methods particularly in the case of Figure 3.11 (using frequency and time domain data).

The performance of the different domain adaptation methods on PPG across the two domains is illustrated in Figure 3.14 (frequency and time domain data) and Figure 3.13 (only time domain data). The bar plot indicates the RMSE when a RR estimation model is trained on the other dataset and tested on the dataset indicated as target domain. We see that the performance of the self-supervised based methods is better than the other two domain adaptation methods in both these figures.

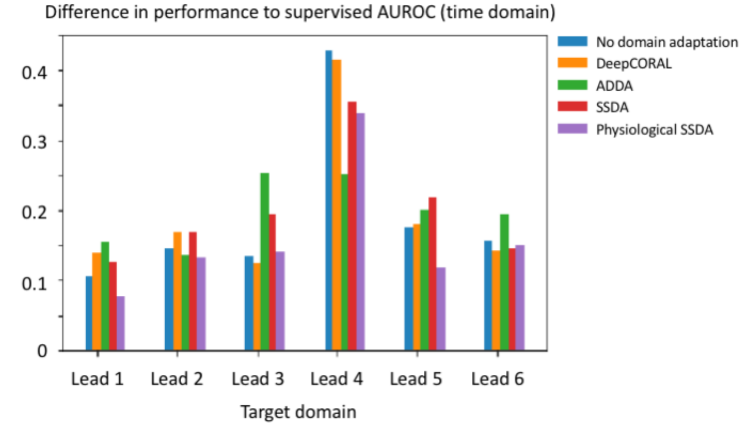


Figure 3.12: Difference in arrhythmia classification (from ECG) performance to supervised learning (lower is better) when the methods are applied to the time domain signal. The supervised performance for each target lead is the corresponding diagonal entry in Figure 3.10. The results shown here reflect the average performance when all other leads are used as the source for a given target lead. We see that physiological self-supervised domain adaptation (SSDA) performs the best.

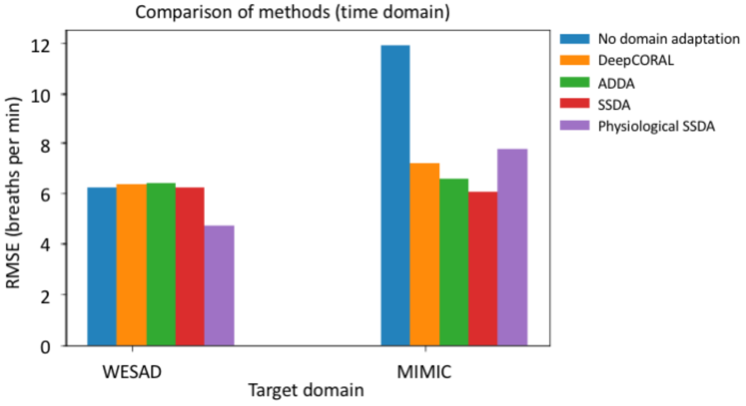


Figure 3.13: RMSE of RR detection from PPG when using the time domain signal. Lower is better.

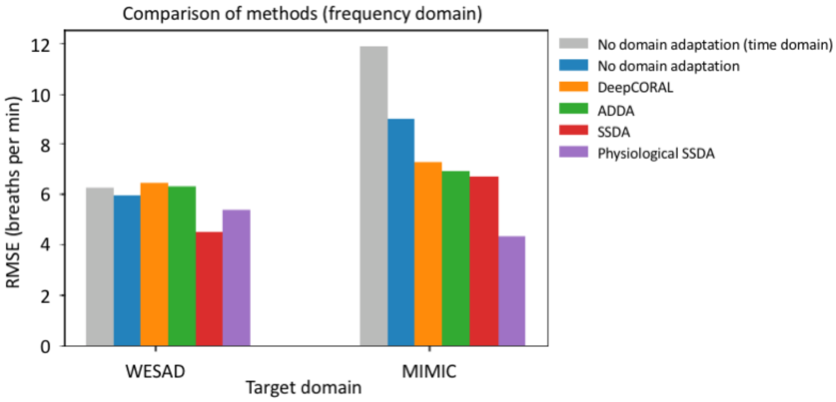


Figure 3.14: RMSE of RR detection from PPG when using both time and frequency domain. Lower is better.

3.6 Discussion

3.6.1 Invariant input representation

Both the ECG and PPG signals are recording periodic cardiac activity within the body and are pulsative in nature. In the case of the different ECG leads, each lead is observing the same cardiac electrical activity from different directions. Similarly, the PPG signal is measuring cardiac activity from different location on the body. The frequency domain should be able to capture this information which is invariant across the domains. We see that including the frequency domain information improves generalization to target domain, both in the case of ECG and PPG signals. Notice that particularly in the case of ECG (Figure 3.11), the performance without any domain adaptation improves substantially when the frequency domain information is included.

3.6.2 Performance of self-supervised domain adaptation when trained on source data only

One of the main advantages of using self-supervised tasks for domain adaptation is that the tasks can be used as a regularizer during training on the source domain only (without using target domain data at training time), which cannot be performed with metric based or adversarial domain adaptation methods. This is useful since target data is often real-world

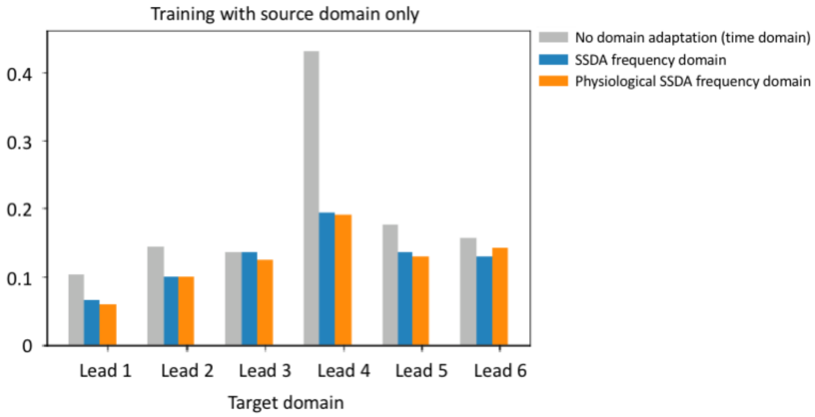


Figure 3.15: Difference in arrhythmia classification (from ECG) performance to supervised learning when using the self-supervised task on source domain only (No target domain data is used during training). We see that self-supervised tasks act as a regularizer and improve target domain generalization.

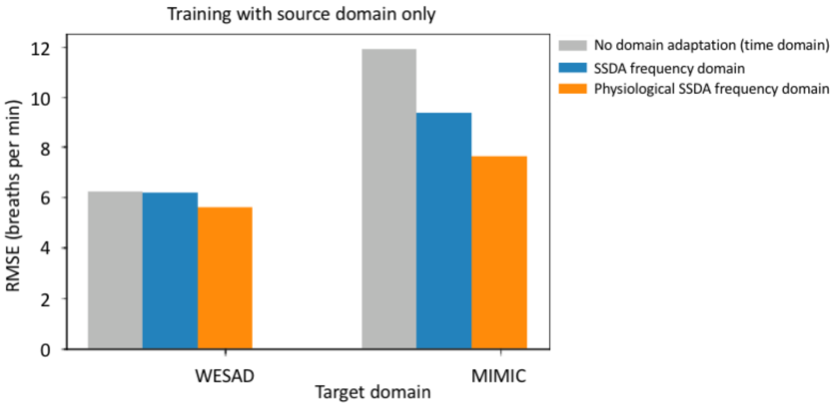


Figure 3.16: RMSE of RR detection from PPG when using the self-supervised task on the source domain only.

data that is not available for adaptation at training time. We evaluate the performance on the ECG and PPG target domains using self-supervised tasks on only the source domain data during training. Figure 3.15 illustrates the performance on each target lead (averaged across all source leads). The result indicated here is the difference to supervised performance, lower is better. We see that both the self-supervised methods result in an improvement in target domain performance when compared to the performance without any domain adaptation. Similarly, Figure 3.16 illustrated the performance of using self-supervised domain adaptation methods with the source domain data only during training. We see that both the tasks help improve target domain performance when compared to performance without domain adaptation. The physiologically-inspired self-supervised task performs better among the two methods.

CHAPTER 4
EXPLAINABLE PATIENT RANKING FOR HOME HOSPITAL CARE

4.1 Introduction

In the previous two chapters we address the shift problems that get reflected in the properties of the data as a result of the choices made while designing the study. An important next challenge to the deployment of ML models in health is bridging the gap between ML models and domain-experts. A predictive model that is used for a decision making task must be *convincing* for it to be deployed in an mHealth study or hospital. We study this problem of model explainability in the context of a health decision making problem: selecting candidate patients for home hospital.

Hospitals are evolving their model of patient care with newly created 'home-hospital' programs, in which patients are sent home to receive care they otherwise would have received in the hospital [53]. The program is made possible by advances in remote sensor monitoring, home-administered interventions, and also in-home internist/nurse visits. The fluctuating availability of hospital beds and fears of hospital-acquired infection inspires interest in home hospital programs both within and outside of the US [135]. Such programs are attractive to both patients and hospitals. Patients are motivated by the benefits including improved sleep, home-cooked food, etc. and hospitals are motivated to send less-critical patients home to expand bed capacity for more-critical patients. Past studies show that home hospital care tends to be substantially less expensive than in-hospital – one study suggests 52% cheaper [53]. Moreover, early studies [53, 136] suggest that home hospital care enjoys similar to slightly better outcomes.

Selecting the right patients to be sent for home care is vital for the success of this program, e.g., patients who might require *acute* interventions (those that can only be performed in a hospital) must not be sent home. The current process of assigning a patient to home hospital relies on manual workflows of physicians constantly reviewing data, which is laborious and not scalable. Machine learning algorithms can be used to learn effective representations from large datasets containing patient records to identify and rank candidates based on their suitability for home hospital care. Our goal is to frame the home hospital task as a patient ranking problem based on the predicted likelihood of an acute intervention. Constructing a high-quality ranking allows the physician to direct their limited resources to those patients most likely to qualify and assess them for suitability and take the *final decision* to send a patient to home care.

The machine learning community has actively worked on clinical outcomes such as sepsis detection [115], mortality prediction [137], intervention prediction [138, 139], 30-day rehospitalization probability [140], disease progression modeling [141]. The *home-hospital problem* we introduce in this work differs from prior works in its focus on *producing a ranking of patients based on predictions of their risk for needing an acute in-hospital intervention within the next 24 hours.* We demonstrate that this task can be addressed using the open source MIMIC III dataset, and our annotations and baseline models will be freely available to encourage more progress on this new clinical prediction task of pressing importance.

Explainabilty is a key property that machine learning solutions must possess if they are to be employed in clinical settings [32]. We develop an approach to explain the home hospital decision of a given black-box model by generating counterfactual [142] patient histories comprised of vital sign *trajectories*. It was inspired by observing our medical collaborators express clinical judgement in terms of hypothetical trends in vital signs, e.g., "This patient is a good candidate for home hospital, but if their systolic blood pressure had been 110 and falling then they would have needed a vasopressor."

In order to be effective in explaining home hospital recommendations to physicians, we must be able to generate counterfactuals which are *plausible*, *relevant* and *sparsely* *perturbed*. Plausibility means that each counterfactual (CF) is consistent with the patient population and does not contain time series data which is unlikely or impossible (e.g., diastolic blood pressure exceeding systolic). Relevant means that counterfactuals reflect the key dimensions that are relevant to an intervention prediction task e.g., generated CFs for predicting the use of a ventilator would differ in SpO2 and respiration rate trajectories (and not for example in their bilirubin level). Sparsity in the perturbation means that the CF alters a minimal number of features to change the prediction and hence the explanation focuses on those factors important for the model's decision.

In this work, we introduce the counterfactual variational autoencoder (CF VAE), a variant of the classic variational autoencoder (VAE) [143] where we modify the latent space to capture the decision boundary of the ML model. The CF VAE is trained to generate CFs at test time through a feed-forward mapping. The key idea behind CF VAE is to sample from the distribution of plausible counterfactuals and generate reconstructions that are relevant to intervention prediction. Our method has two main advantages: 1. We produce plausible counterfactuals by sampling from the VAE latent space, and 2. The latent space is trained to capture the black-box model and thus learns relevant representations for the task. We train a multi-head self-attention [79] based CF VAE to produce plausible *time series* counterfactuals. For the home hospital task, we present results for pair-wise ranking based on the time to next acute intervention with an attention-based pair-wise ranking model. We use this as the black-box model and generate counterfactuals from the CF VAE.

In this work, we introduce the counterfactual variational autoencoder (CF VAE). Traditionally, VAEs [143] are used to generate realistic synthetic data. We show how the traditional VAE loss function (first two terms in Figure Figure 4.1) can be modified to generate compelling CFs. The CF must be close to the original patient's data (first term), be of the opposite class (third term) and yet not change too many features (fourth term). By including additional terms in the loss function we are able to use stochastic gradient descent to train a VAE that will generate desirable CFs. Our method has two main advantages:

1. We produce plausible counterfactuals by sampling from the VAE latent space, and 2. The latent space is trained to capture the binary prediction model and thus learns relevant representations for the task. We train a multi-head self-attention [79] based CF VAE to produce plausible *time series* counterfactuals. For the home hospital task, we present results for pairwise ranking based on the time to next acute intervention with an attention-based pairwise ranking model. We use this as the binary prediction model and generate counterfactuals from the CF VAE.

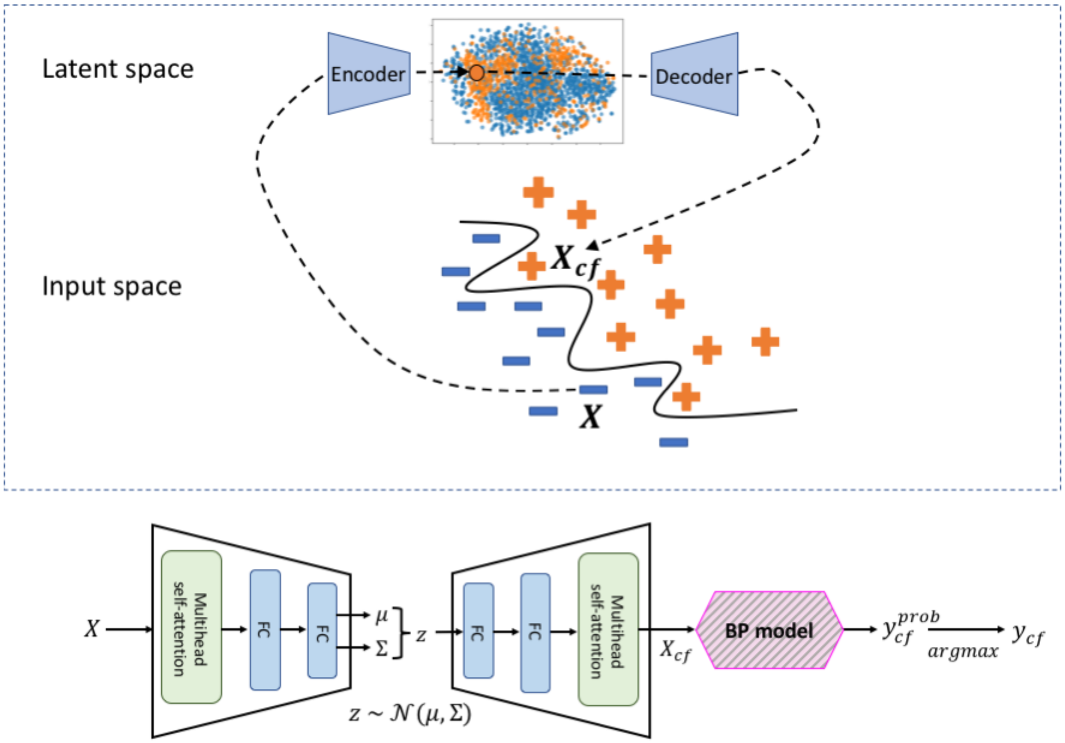
4.2 Contributions

This chapter makes the following contributions:

- We introduce the home-hospital ranking problem and frame it on the open source MIMIC dataset.
- We present results from a ranking model and achieve over 90% accuracy based on two acute interventions: ventilator and vasopressor.
- We develop CF VAE: A VAE based feed-forward method to produce plausible, relevant, and sparsely perturbed counterfactuals.
- We present a quantitative evaluation of the counterfactuals produced by our method and prior works and note that a higher fraction of our counterfactals are plausible.
- We present results from a qualitative analysis of the counterfactuals produced: counterfactuals generated by our method receive a plausibility score of 75% when compared to 30% for a prior method.

4.3 Related work

While clinical prediction tasks have been widely-studied, we are not aware of past work that defines and tackles the home hospital problem via machine learning methods that leverage



$$loss = \underbrace{\|X - X_{CF}\|_2{}^2 + \mathcal{KL}(\mathcal{N}(\mu,\Sigma)||\mathcal{N}(0,\mathrm{I}))}_{\text{VAE loss}} + \underbrace{\lambda_{cf}\ CrossEntropy\left(y_{cf}^{prob}, 1 - y\right)}_{\text{Counterfactual loss}} + \underbrace{\lambda_S \|X - X_{CF}\|_1}_{\text{Sparsity loss}}$$

Figure 4.1: Our proposed approach: learn a VAE latent space that embeds input data points onto regions of the latent space whose labels are opposite those of the points. This can be used to generate counterfactuals for any binary prediction (BP) model.

the large, publicly available MIMIC III dataset. Previously, clinical sites have developed their home hospital programs and protocols around datasets that are not broadly-available to the ML research community [136, 53], and were therefore difficult for the community to iterate and innovative upon.

A second contribution of this work is a VAE-based deep architecture for the home hospital problem. Our approach addresses the three main tasks of counterfactual generation, intervention prediction, and patient ranking. We discuss related work for each of these tasks below.

Table 4.1: Comparison of different counterfactual generation methods: CEM [144], REVISE [51], DICE [52], NG-CF [145], CF VAE - this work.

	CEM	REVISE	DICE	NG-CF	CF VAE (ours)
Learned representation	✓	✓	X	✓	✓
Plausibility	X	✓	X	X	✓
Time series	X	X	X	✓	✓
Relevance	X	X	X	X	✓
Feed-forward approach	X	X	X	X	✓
Sparse perturbations	✓	X	✓	X	✓

4.3.1 Explainability via Counterfactual Generation

Explainable machine learning is a heavily studied field too vast to adequately summarize in this section. Comprehensive surveys on the topic include [46, 47, 48, 49]. We focus on explainability via counterfactual (CF) generation [50, 51, 52], where the determining features of the classification model are highlighted by comparison to a diverse set of other similar (synthetic) patients to whom the classifier would assign an opposite label.

Our approach to generating CFs possesses six key features not previously found in combination among prior CF approaches, as illustrated in Table 4.1. (1) We generate *time-series* CFs, a relatively understudied topic with some recent work [145, 146, 147] . (2) Our CFs are usually *biologically plausible*, meaning that we avoid reporting SpO$_2$ values of 105% or diastolic blood pressures exceeding systolic. This contrasts with methods that look for nearby CFs under the Euclidean metric [146, 147], which are prone to sampling from outside the true data manifold. Nearest unlike neighbor approaches as in [145] partially avoid this issue, but can produce unrealistic time-series of vital signs when the past history of the patient in question is stitched together with the time series data of its "unlikely neighbor". We describe this in more detail along with results in Sec **??**. [51] use a standard variational autoencoder with reconstruction loss to generate a patient embedding that encourages plausibility. (3) In contrast, our CF VAE incorporates a loss which captures relevance to the prediction task, encouraging CFs that are both plausible and relevant. (4) Our approach is *feed-forward only*, meaning that we do not require test-time optimization to generate CFs as in [145, 52, 144, 51]. (5) Our approach leverages a *learned representation*, which induces a probability distribution over patients and gives us a good distance metric for use in other applications on the same dataset. (6) Our approach produces *sparse perturbations* which can be interpreted as the factors important for the model outcome. We achieve sparse perturbations through an ℓ_1 regularization on the perturbation, similar to [144]. [52] achieves sparsity in perturbations through post-processing of the CF.

Note that model *interpretability* is a related and widely studied research area. Some representative papers include [148, 46]. However, the focus of this work is to only explain the model outcome and not feature representations learned.

4.3.2 Patient ranking

The closest work to this is [149], in which pneumonia patients are ranked according to their mortality risk. The lowest ranked patients are evaluated as candidates for home treatment. The model is further trained on a secondary task to predict lab outcomes to learn better representations for the ranking task. However, subsequent work [150] identified concerns with using this intervention-oblivious model to assess risk. Similarly, [151] attempts to prioritize patients according to the probability that a quick intervention can prevent their death. [152, 153] rank patients based on their likelihood of being discharged from the hospital, and [154, 155, 156] all attempt to predict length of stay in the ICU unit or in home hospital care. In contrast, we rank patients based on their predicted time to requiring an acute intervention. This is a different task than ranking based on the risk of mortality or the expected total duration of care.

4.3.3 Predicting Time to Intervention

Other prior work focuses on predicting when an ICU patient will need their next intervention (regression), or whether they will need an intervention within the next x hours (classification). For example, [139, 138] train autoregressive state space models to predict if interventions such as vasopressor or ventilator are needed in the next 6-8 hours. [157] benchmark a number of different classical ML and deep learning models on predicting the onset of interventions, among many other similar tasks. While the focus of this work is not to develop a model for predicting time to intervention per se, it is an auxillary task that is relevant to patient ranking. In particular, while we use the time to next acute intervention as a means to determine the ground truth ranking of patients, only the relative ordering is important to us.

4.4 Methods

Our solution approach for home hospital is designed to meet two requirements: 1) Given a population of patients, rank them by the likelihood that they will require an acute intervention; and 2) Given any target patient of interest, generate a counterfactual patient for visualization with the attendant clinician, as a means to explain why the target patient was selected (or not) for home hospital. Our solution architecture has three components: 1) A novel Counterfactual VAE (CF VAE) module (Fig. Figure 4.1), which provides a general, feed-forward approach to synthesizing counterfactuals for time series classification problems (and is used in home hospital to map target patients to their counterfactuals); 2) A Multitask Model (Fig. Figure 4.3) which learns a patient embedding from raw time series measurement data, and produces acute intervention prediction and patient ranking outputs; and 3) A Training Procedure (Fig. Figure 4.4) for learning to rank patients and generate counterfactuals.

4.4.1 CF VAE: Counterfactual Variational Autoencoder

In this section we describe our novel and general CF VAE architecture for counterfactual generation, illustrated in Figure 4.1. In subsection 4.4.2, we describe how this module is incorporated into the overall home hospital solution. The key idea is to train a VAE to generate counterfactuals via a *feed-forward mapping*. In contrast, prior methods perform optimization at test time to identify counterfactual samples, as exemplified by DICE [52] and REVISE [51], which optimize in input and latent spaces, respectively. In our approach, sampling from the counterfactual distribution only requires one feed-forward pass, as opposed to making multiple calls to an optimization module.

Our approach has two main advantages. First, we can learn a patient embedding that jointly optimizes for plausibility (samples respect the data distribution), validity (samples flip the outcome of the classifier) and sparsity (minimal feature change). Since CFs provide a means for a clinician to interrogate and understand the assessments performed by a deep model, generating realistic CFs is critical in order to establish trust. We show experimentally in Sec. section 4.5 that our joint training method produces more realistic CFs than prior methods. A related benefit of joint training is improved sample efficiency in using the training data. In contrast, the generation of one successful sample via optimization does not make it any easier to generate the next one, because the optimizations are done independently at test time.

A second benefit of CF VAE is that the learned patient embedding encodes the properties of the patient trajectories that are relevant to the intervention prediction task, facilitating the generation of relevant CFs.

In particular, generated CF patients will be clustered around a given input patient, but differ in their feature trajectories in ways that are consistent with the acute intervention prediction. For example, suppose a 10% drop in SpO2 over time triggered an acute intervention like ventilator for an input patient. We would like the CF patients to be similar on the vitals that are irrelevant to the treatment (e.g. blood pressure), but differ in their SpO2 trajectory in intuitive ways (e.g. a 5% drop or perhaps a slight rise). It is difficult to achieve this type of relevance in the case of prior methods that do not utilize a task-specific representation during CF generation. An example of an irrelevant CF produced by DICE is shown in Figure 4.5. Additionally, CFs that deviate minimally from the input (say only in the blood pressure) are easier to interpret by a physician. We include a constraint in our algorithm, similar to [144] to have sparse perturbations while still being plausible. In comparison, [52] adapts a post-processing approach that might result in sparse CFs that are not plausible.

We finally note that by incorporating multihead self-attention [79] blocks in the encoder and decoder, the VAE can handle time series measurements as inputs, and synthesize counterfactual time series as outputs, thereby achieving the goals in Table Table 4.1. We now describe our solution architecture, beginning with a brief overview of a vanilla VAE [143].

VAE background: The VAE approximation takes the form of a standard encoder-decoder pair where the encoder, Q, and the decoder, P, are each parameterized by neural networks. The encoder and decoder networks are trained by *maximizing* the objective:

$$E_{X \sim D}[E_{z \sim Q}[\log P(X|z)] - \mathcal{D}(Q(z|X)||P(z))] \tag{4.1}$$

Where X is a data point sampled from the dataset D, the encoder Q produces a distribution $\mathcal{N}(\mu_X, \Sigma_X)$ over the latent representation z, and $\mathcal{D}$ is the KL divergence between the latent multivariate Gaussian distribution and the prior distribution $P(z)$. The two terms in the objective function correspond to the reconstruction error and latent space normalization, respectively.

When both the prior $P(z)$ and output distributions $P(X|z)$ are assumed to be spherical Gaussians, maximizing Equation (4.1) can be shown to be equivalent to minimizing

$$E_{X \sim D}[||X - X'||_2{}^2 + \mathcal{KL}(\mathcal{N}(\mu_X, \Sigma_X)|\mathcal{N}(0,1))], \tag{4.2}$$

where $X' = P(Q(X))$ is the network's reconstruction of X. We therefore use Equation (4.2) as a starting point for defining a loss function for training our VAE. See [158] for a more complete derivation of this objective.

CF VAE objective: To produce realistic counterfactuals, we must generate samples with high probability under the data distribution that flip the output of a target classification model. In the context of home hospital, the target is a multitask model that maps a patient representation into a score that can be used for ranking, along with a binary prediction of whether the patient will receive an acute intervention. We use the acute intervention prediction as the output for the purpose of counterfactual generation. Note however that our CF VAE approach can be used for any binary prediction model. We denote the binary classification output of the target model as $y = \text{BP}(X)$ and the output of the CF VAE as X_{cf}.

We modify the VAE objective so that its output (X_{cf}) is penalized by a term proportional to the cross entropy of y_{cf}^{prob} and $1 - y$ (where y_{cf}^{prob} is the class probability output of $\text{BP}(X_{cf})$) in addition to the standard regularization and reconstruction loss on X and X_{cf}. Introducing this extra loss term allows the VAE to learn about the target model's decision boundary and incentivizes it to synthesize a counterfactual sample X_{cf} whose output y_{cf} is of the opposite class. Intuitively, as diagrammed in Fig. Figure 4.5(b), this teaches the VAE to encode the classifier boundary in its latent space, and to map a given X to a latent point of the opposite class. Another consideration is to have *minimal* changes to the input to produce a CF. We achieve this through a sparsity constraint on the perturbation. Thus, our modified loss function takes on the form

$$\begin{aligned} E_{X \sim D}[||X - X_{cf}||_2{}^2 + \mathcal{KL}(\mathcal{N}(\mu_X, \Sigma_X)|\mathcal{N}(0,1)) \\ + \lambda_{cf}\text{CrossEntropy}(y_{cf}^{prob}, 1 - y) + \lambda_S ||X - X_{cf}||_1] \end{aligned} \tag{4.3}$$

where X_{cf} is the decoder output and λ_S, λ_{cf} act as Lagrange multipliers for sparsity and for the "soft constraint" that CF class y_{cf} and y must differ, respectively. Note that Eq. Equation 4.3 differs from the standard VAE loss only in the cross-entropy and ℓ_1 norm term. Adjusting λ_{cf} allows us to tune the VAEs attention between focusing on its re-

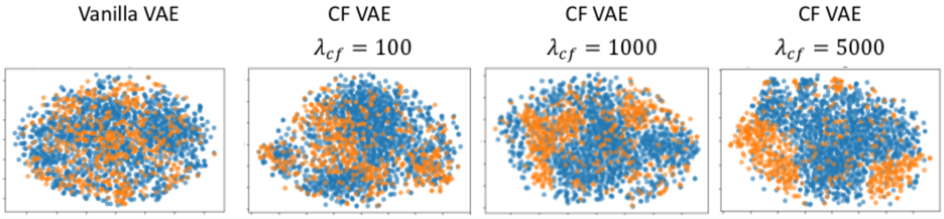


Figure 4.2: TSNE plot of the latent space of a vanilla VAE and CF VAE with varying λ_{cf}. We visualize the two classes each point belongs to: requires intervention, or doesn't require intervention. We see that the CF VAE captures the classifier boundary and learns to separate the two classes in the latent space as λ_{cf} increases.

construction/regularization objectives and on its counterfactual objective. Fig. Figure 4.2 visualizes the latent space using TSNE with varying λ_{cf} – as we increase λ_{cf}, we see more separation between the classes in the latent space. We choose the λ_S value proportional to the magnitude of the different loss terms on the training set. See Sec. **??** for examples of CFs generated with and without the sparsity term.

A strength of our approach is that the CF VAE can be trained in the same way as a Vanilla VAE, using stochastic gradient descent, allowing us to leverage the VAE optimization literature. Note that the parameters of the binary prediction model are held fixed while training the CF VAE.

The version of CF VAE described above represents each patient as a time series with an $N \times T$ data matrix, where T is the number of time samples and N is the number of measurements (e.g. vital signs). We have also explored an alternative temporal representation based on linear trends, where each vital measurement is modeling as trending (e.g. up or down) with a specific slope and intercept over the measurement window. We present results for both patient representations in the subsequent sections.

4.4.2 Multitask learning of ranking and acute intervention prediction

We require an ordering of patients in the hospital based on who is most suitable for home care. Patients being sent home *should not* require an acute intervention within the next day.

We frame this as a pair-wise ranking task: given patients A and B, we rank them based on who will first require a critical intervention. Let A require an acute intervention in T_A hours and B require an acute intervention in T_B hours. The ground truth ranking is $A > B$ if $T_A > T_B$ or vice versa. We model the ranking function using a neural network, as proposed in [159]. Given two patients A and B, the ranking network computes scores R_A and R_B, and produces the output $Sigmoid(R_A - R_B)$. We learn the network weights using the binary crossentropy loss where the binary targets correspond to $A > B$ or $A < B$. To obtain a ranking order for all patients, we can compute pair-wise rankings selecting two patients at a time and produce an overall ordering. Since the computed scores define a total ordering on patients, we obtain consistent pair-wise rankings: if $A > B$ and $B > C$, A will be greater than C when we perform a pair-wise comparison of A and C.

We train a multitask model to perform two tasks: (1) produce a ranking score, and (2) predict if an acute intervention is required in the next 24 hours. We believe that adding the second task would not hurt the ranking performance (which is the main focus of home hospital) since the two tasks are related. The architecture of the multitask model consists of a multi-head self attention module to model the time series input as shown in Fig. Figure 4.3 and it is trained as in Fig. Figure 4.4 using the ranking loss and prediction loss combined. As mentioned in the previous section, we explore another approach to represent the patient time series, by summarizing it in the form of a slope and intercept. When the data is represented in the slope-intercept form, the self-attention blocks in the CF VAE and multitask model architecture are replaced with an MLP with ReLU activation. We use the multi-task model with acute intervention prediction output as the binary prediction model while training the CF VAE to explain the multi-task model.

4.5 Experiments and Results

The two main objectives of our experiments are to show that: 1) We can reliably rank patients in the order of the time to intervention, and 2) Our CF VAE method produces realistic counterfactuals as explanations.

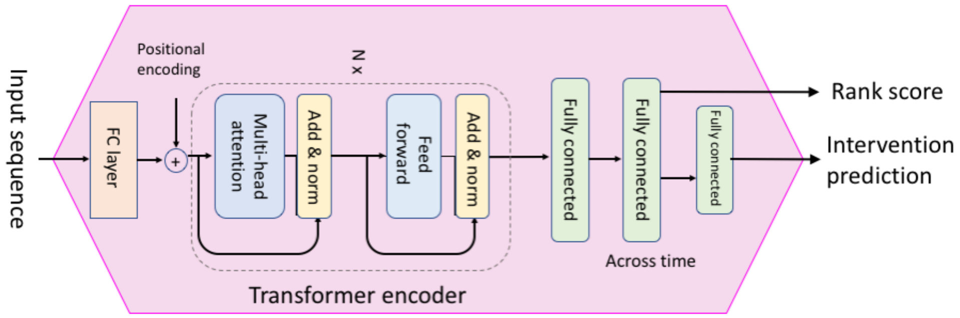


Figure 4.3: Model architecture for producing ranking and acute intervention prediction output. The architecture consists of a transformer encoder for modeling the temporal sequence of patient data.

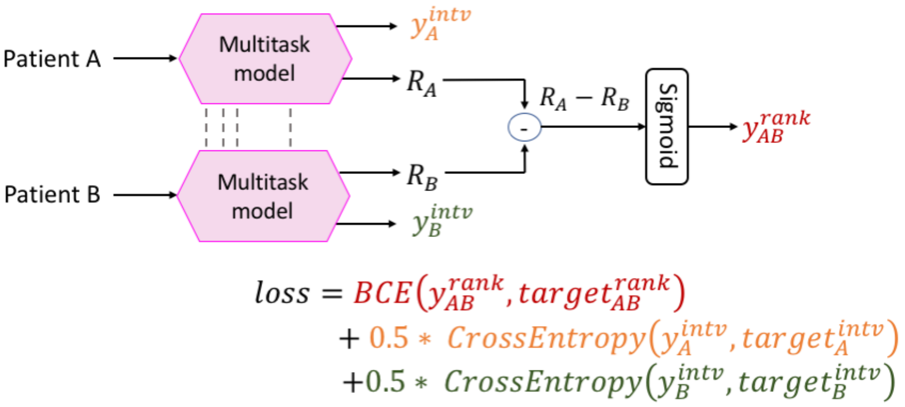

Patient A
Multitask model
y_A^{intv}
R_A
$R_A - R_B$
-
Sigmoid
y_{AB}^{rank}
Patient B
Multitask model
R_B
y_B^{intv}
$loss = BCE(y_{AB}^{rank}, target_{AB}^{rank})$
$+ 0.5 * CrossEntropy(y_A^{intv}, target_A^{intv})$
$+0.5 * CrossEntropy(y_B^{intv}, target_B^{intv})$

Figure 4.4: Training pipeline for the ranking and predicting intervention. The multitask model shown here is the MLP (for slope-intercept form) and transformer model (for temporal sequence form).

4.5.1 Dataset and pre-processing

We use the vitals, interventions, and other events recorded from patients in the publicly available MIMIC III dataset [45] in our experiments. MIMIC III consists of deidentified data from 53,000 patients admitted to the Beth Israel Deaconess Medical Center in Boston. We use the data pre-processing pipeline in [157] to transform the MIMIC III raw vital signs and interventions into hourly time series. The temporal patient data is segmented into 48 hour windows, where each window is a data point. For each 48 hour patient window, we have an associated time to next acute intervention (t_{intv}) and a binary label of whether they receive an acute intervention within the next 24 hours ($intv_{24}$). Given the 48 hour window of features for patient A and B, the multitask model produces 3 outputs: 1) pairwise ranking of A, B; 2) prediction that A receives an acute intervention within 24 hours; 3) prediction that B receives an acute intervention within 24 hours. We perform a patient-wise split of 70%-15%-15% for training, validation, and testing. We use features corresponding to **vital signs** (heart rate, systolic blood pressure, diastolic blood pressure, oxygen saturation, respiratory rate, temperature), **interventions** (ventilator, vasopressor, adenosine, dobutamine, dopamine, epinephrine, isuprel, milrinone, norepinephrine, phenylephrine, vasopressin, colloid bolus, crystalloid bolus), and **demographics** (age, gender) for the prediction and ranking task.

4.5.2 Baseline methods for generating counterfactuals

The space of prior CF methods can be roughly partitioned into three approaches: (1) Optimization-based approaches in the input space [144, 52], (2) Optimization-based approaches in the latent space of a generative model [51], and (3) Input perturbation-based approaches [145]. The third category is not suitable for us since substituting a part of a patient's vital signs with vital signs from another patient could be unrealistic. In our experiments, we compare to DICE [52] and REVISE [51] to represent the two optimization categories.

4.5.3 Multitask learning of ranking and acute intervention prediction

A pair of 48 hour patient windows A and B are input to the model and the model produces a ranking order based on who requires acute intervention I first. For the purpose of our experiments, we show results on two acute interventions I separately: ventilator and vasopressor.

The pairwise ranking and acute intervention prediction performance are shown in Table Table 4.2. We see that the model achieves over 90% accuracy for both tasks. We also notice that using the entire temporal sequence in the 48 hour window improves the performance of the ranking and prediction task as compared to the slope-intercept representation, indicating that the hourly pattern of the temporal data helps us rank and predict acute intervention more accurately. *For some interventions, such as ventilators, the presence of past interventions is very predictive of similar interventions being required in the near future.* To test this, we perform an ablation study excluding information about the history of the target intervention. We find that while there is a reduction in ranking and prediction accuracy, the model still learns a good representation from only the vitals and other interventions.

Table 4.2: Multitask model results from three experiments: 1. Ranking and acute intervention prediction using history of vitals, interventions and demographics, 2. Results *without using* the history of the target intervention as an input feature (to rule out any data leakage), 3. Results when we perform ranking only on the patients who require acute intervention within the next 24 hours. *Error bars generated by running the experiment with 10 random seeds.* Here, ACC = Accuracy, AUC = Area under ROC curve, Intv = intervention.

Experiment			**Slope-intercept input**		**Temporal input**	
			Vaso	Vent	Vaso	Vent
All features & all patients	Ranking	ACC	0.88 ± 0.01	0.93 ± 0.01	0.96 ± 0.00	0.94 ± 0.00
		AUC	0.95 ± 0.01	0.98 ± 0.00	0.99 ± 0.00	0.98 ± 0.00
	Intv pred	ACC	0.83 ± 0.01	0.83 ± 0.12	0.90 ± 0.01	0.91 ± 0.01
		AUC	0.88 ± 0.02	0.91 ± 0.01	0.96 ± 0.00	0.92 ± 0.01
Ablation study: w/o history of target intv as feature	Ranking	ACC	0.89 ± 0.00	0.86 ± 0.00	0.96 ± 0.00	0.88 ± 0.00
		AUC	0.96 ± 0.00	0.93 ± 0.00	0.99 ± 0.01	0.94 ± 0.01
	Intv pred	ACC	0.84 ± 0.01	0.79 ± 0.01	0.91 ± 0.00	0.83 ± 0.01
		AUC	0.89 ± 0.00	0.83 ± 0.00	0.96 ± 0.00	0.86 ± 0.00
Only patients who need intervention	Ranking	ACC	0.72 ± 0.08	0.95 ± 0.00	0.94 ± 0.01	0.97 ± 0.00
		AUC	0.80 ± 0.00	0.99 ± 0.00	0.97 ± 0.00	0.99 ± 0.00

4.5.4 Visualization of counterfactuals

The counterfactuals produced in the case of the vasopressor intervention are visualized in Figure 4.5 using the method of DICE and CF VAE. with the slope-intercept and time-series representation of the patient data respectively. In these figures, the patient (blue dotted) originally **did not require a vasopressor** in the next 24 hours. The CF VAE produced a counterfactual (orange line) in which case the patient **would require a vasopressor** in the next 24 hours. In both cases, we see that the counterfactual reduces the systolic blood pressure - which is a key trigger for providing vasopressors. The counterfactual indicates that the binary prediction model has learned patterns between decreasing systolic blood pressure, increasing heart rate, and the need for vasopressors. These patterns can then be evaluated by a physician to understand the model decision.

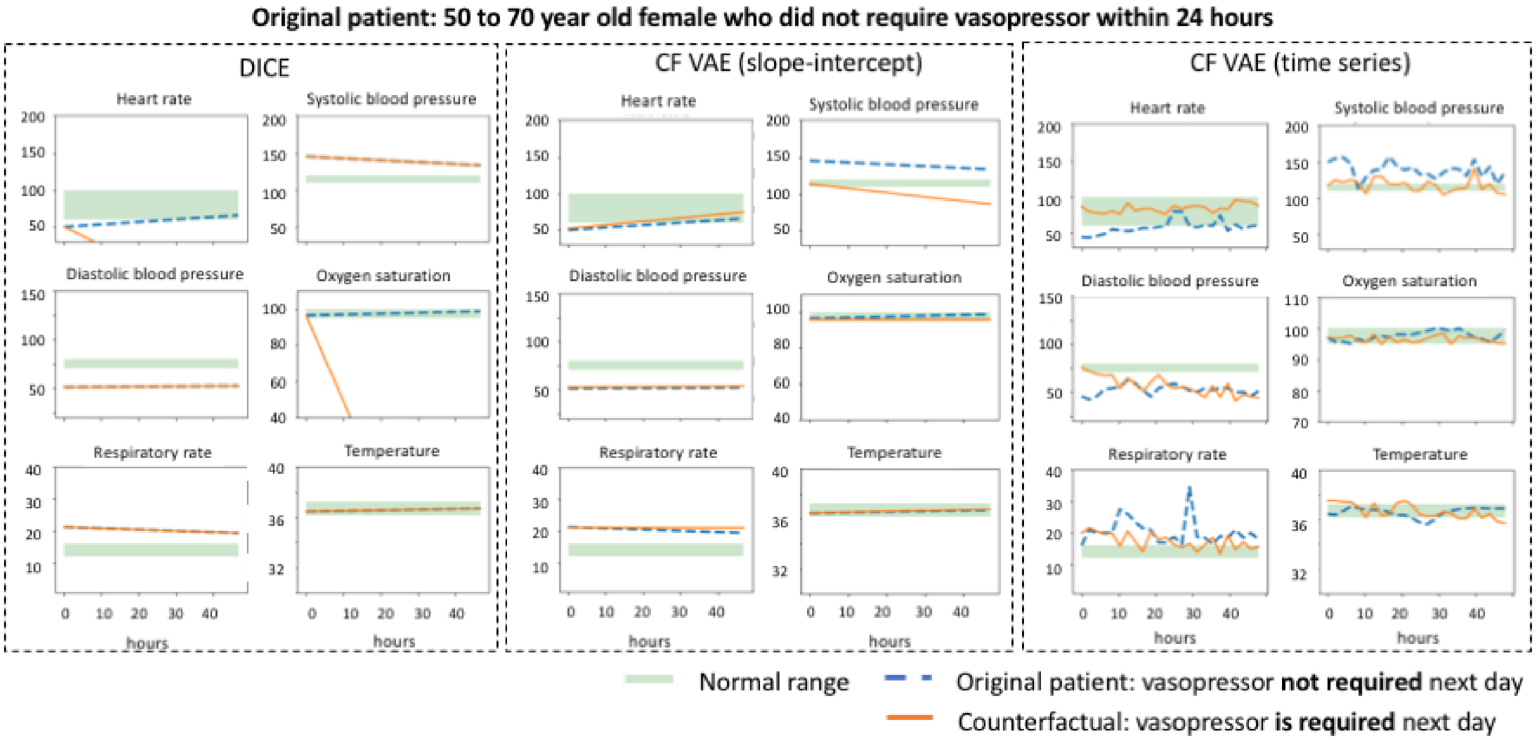


Figure 4.5: Counterfactual (CF) generated using DICE and CF VAE. The original patient (blue dotted) requires a vasopressor within 24 hours. The orange line shows the generated CF. The CF generated using DICE neither looks plausible nor relevant. Vasopressors are provided when the blood pressure drops - the DICE CF changes oxygen saturation and respiration rate

4.5.5 Evaluation of counterfactuals

We generate counterfactuals for 100 data points in the test set using three methods: 1. DICE [52], 2. REVISE [51], 3. CF VAE (ours). Table Table **??** compares the different methods based on three aspects:

Log likelihood score under a KDE model: We compute the log likelihood score of a generated counterfactual under the kernel density estimator (with Gaussian kernel) fit to the training data to quantify its plausibility. A higher log-likelihood score implies that the counterfactual is plausible and similar to a real patient in the training data. The column $\%l_{method} > l_{CFVAE}$ in Table Table **??** is the ratio of test samples for which the likelihood score of counterfactuals generated by DICE and REVISE were greater than that of counterfactuals generated by CF VAE. We see that DICE compares poorly with our method on the likelihood scores, while REVISE performs comparably to us.

Validity of counterfactuals generated: Out of the total set of test points, Table Table **??** presents the number of generated points that have the opposite outcome with respect to the binary prediction model (i.e. the true counterfactual). Note that REVISE has a very low $\%$ validity because the optimization did not converge to a counterfactual.

Train and test time: The training and test time for each method is listed in the table. The train time is a one-time cost, while the test time listed is the average time for generating a counterfactual for one test point. Since DICE and REVISE are optimization-based approaches, they incur a higher cost at test time. However, the cost of CF VAE is a one-time training cost. Generating a counterfactual with CF VAE involves performing a forward pass though the trained model, which is 100x faster than DICE and REVISE.

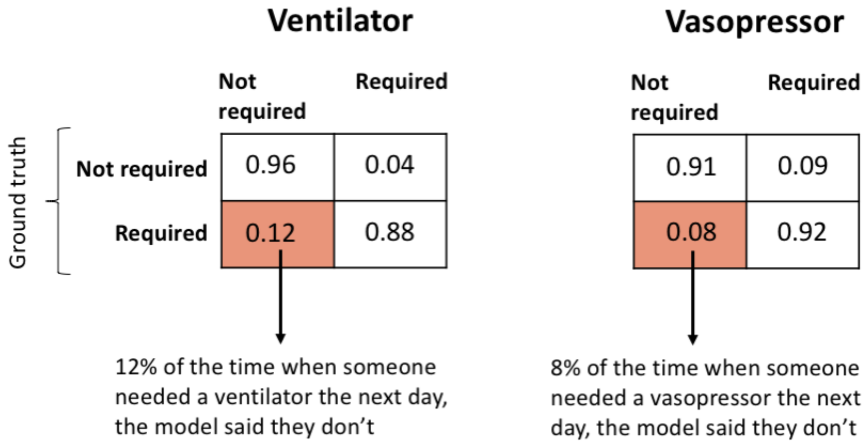
We see that DICE produces a valid CF 100$\%$ of the time since the method involves optimizing in the Euclidean space for a counterfactual. This could result in generating outputs that might not be plausible but flip the outcome of the binary prediction model (e.g., Fig Figure 4.5 where the respiration rate and oxygen saturation look unrealistic and not relevant to the prediction task). Whereas, REVISE is a method which performs optimization in the latent space to and hence produces plausible counterfactuals.

Counterfactual evaluation by an expert We presented the CFs generated by DICE and our method to an emergency medical physician to score based on plausibility and relevance. The physician was blinded to whether the CF came from DICE vs. our method. A *plausible and relevant* CF is one which convinces the physician that "if the patient looked as in the CF, their acute intervention outcome would be reversed". We provided 10 CFs each for the vasopressor prediction and ventilator prediction task (5 in each of the binary class). The physician marked 80% and 60% CF VAE CFs as plausible and relevant and 20% and 40% DICE CFs for the vasopressor and ventilator prediction tasks respectively Table Table **??**. We exclude REVISE from this evaluation due to its poor convergence rate(it fails to produce a counterfactual 75% of the time).

Compared to the two baselines, our method generates highly valid counterfactuals that are more plausible, relevant (based on the small sample of expert evaluation), and in less time. Additionally, we evaluate the proximity of CFs generated at test time.

4.5.6 Impact of the sparsity term on the counterfactuals

By sparsity in perturbation, we mean that a minimal number of features should be changed. This is an important consideration as it reduces the cognitive load on a physician. We include a soft constraint to encourage counterfactuals that alter a minimal number of dimensions with the $\lambda_s||X - X_{cf}||_1$ term in Eq.Equation 4.3. Figure 4.8 illustrates examples with and without sparsity in the CF VAE loss function. We see that without sparsity, the CF VAE produces a counterfactual that changes multiple input dimensions - systolic blood pressure, temperature and respiratory rate. However, adding the sparsity constraint results in a counterfactual that makes large changes only in the systolic blood pressure. Note that a vasopressor is administered to increase a patient's blood pressure up to normal levels. The CF VAE with sparsity is correctly identifying the key feature relevant to the target



Ventilator

Ground truth	Not required	Required
Not required	0.96	0.04
Required	0.12	0.88

Vasopressor

Ground truth	Not required	Required
Not required	0.91	0.09
Required	0.08	0.92

12% of the time when someone needed a ventilator the next day, the model said they don't

8% of the time when someone needed a vasopressor the next day, the model said they don't

Figure 4.6: Confusion matrix of the 24-hour intervention prediction task. The model falsely recommends 12% (resp., 8%) of patients who actually needed a ventilator (resp., vasopressor) for the home hospital program although they should remain in-hospital.

intervention.

4.6 Conclusion and future work

The ranking and intervention prediction performance of our model demonstrates the effectiveness of our solution to the home hospital problem. We are able to reliably rank patients, predict interventions, and generate high quality counterfactuals to explain the ML model's decision. In this section, we discuss the limitations of our model and some of the ample opportunities to improve upon it.

Figure 4.6 shows the confusion matrix for the acute intervention prediction task. In practice, false negatives are more severe than false positives. We find that 12% (8%) of patients who needed a ventilator (vasopressor) would have been recommended for home hospital. Cost-sensitive approaches that account for this imbalance deserve exploration.

We analyze the errors on the pairwise ranking task and observe that the majority of such mistakes are made when the time difference between the pairs is *small* which is a harder problem (e.g., ranking patients who might need intervention 5hr vs 7hr into the future). Figure Figure 4.7 quantifies the error. For the home hospital ranking problem, the most important case is being able to distinguish between patients who require acute-intervention

>24 hours apart - and our error is low in these cases.

In our experiments, we focused on a single intervention such as a vasopressor as an example of an acute intervention. This helped us understand whether vasopressor-related counterfactuals were meaningful. However, ideally, we would predict whether any acute intervention is needed. One complexity that arises is that future acute interventions are influenced by earlier sub-acute interventions. If a condition is caught early enough, then future acute interventions may not be necessary. Hence a home hospital algorithm when implemented should account for the complex relationship across all interventions.

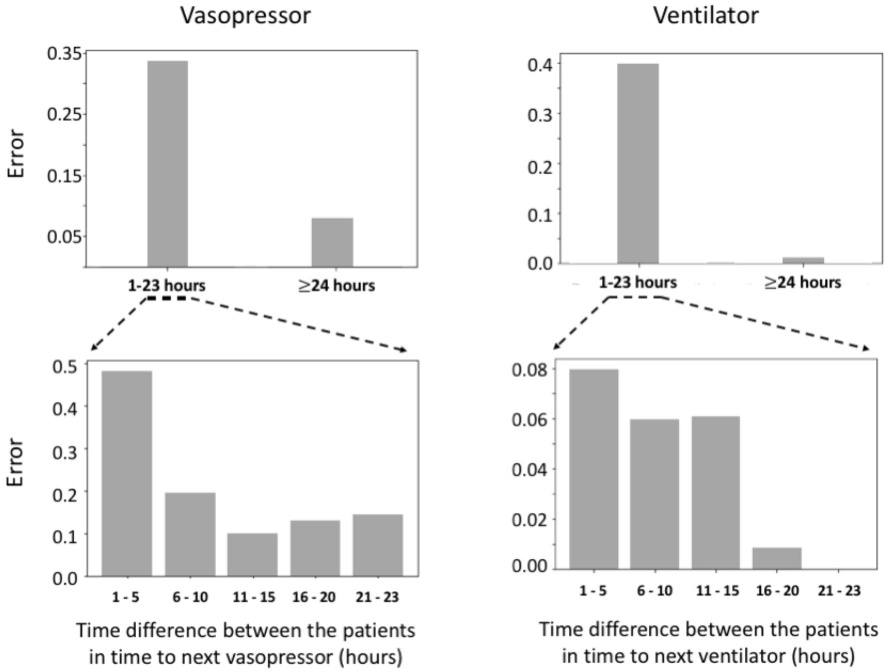


Figure 4.7: Pair-wise ranking error rate based on the difference in time to intervention between the pairs. In the top figure, the error rate is lower when one of the patients requires an intervention in >24 hours. This is important since we want to identify such candidates for home hospital. The bottom shows ranking performance when evaluated only on pairs where an acute intervention was required within 24 hours - we see a larger error rate when the two patients are <5 hours apart.

We chose MIMIC-III for our experiments because it is publicly available and facilitates replication. In reality, ICU patients are not candidates for home hospital care. Hence, our model will have to be trained on a data set representing patients more likely to be admitted to home hospital. Methods and analysis from this work can be directly transferred to another dataset. Another limitation of MIMIC is the data is collected from a hospital system in Boston, MA. This implies that the training data is skewed to the Boston demographic. If a group is under-represented in the training data, it may perform poorly on that same group in the test data resulting in more erroneous home hospital decisions for these groups. Prior to deploying any such model, accuracy should be evaluated across all groups.

[52] argue that diversity of CF is an important characteristic of a CF generation method. Diverse CFs alter *different* feature dimensions to reverse the classifier outcome. A potential weakness of our method is that sampling from a smooth latent space may reduce diversity. This is a very interesting avenue for future work.

Finally, while our approach is able to identify realistic CFs in the sense that CF patients are likely to exist, we do not consider whether there is a realistic path from a patient's present state to the CF state. We leave this as an interesting challenge for future work.

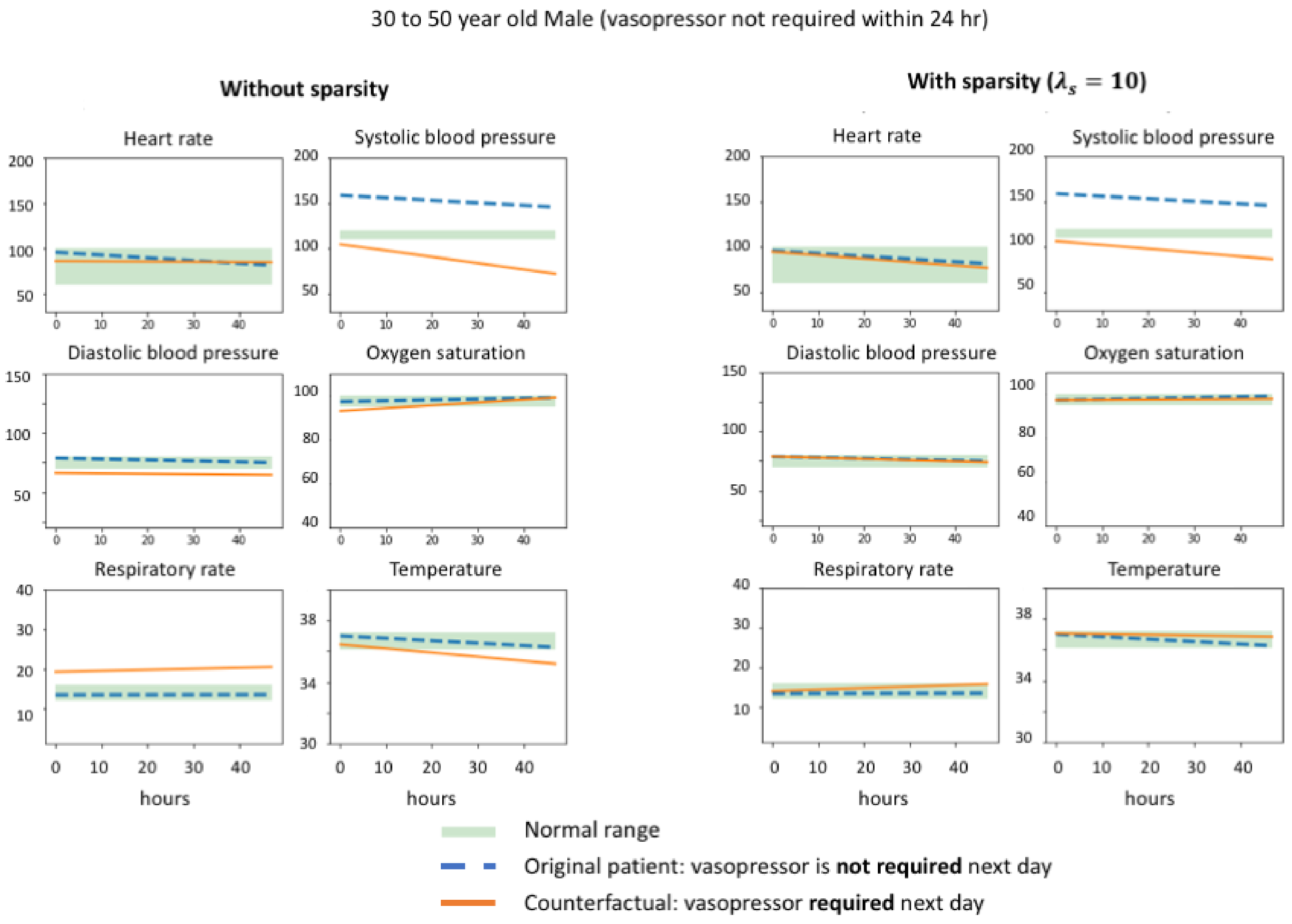


Figure 4.8: Example 1: Counterfactual produced with and without the sparsity term. Notice how only the systolic blood pressure changes when we include the sparsity constraint.

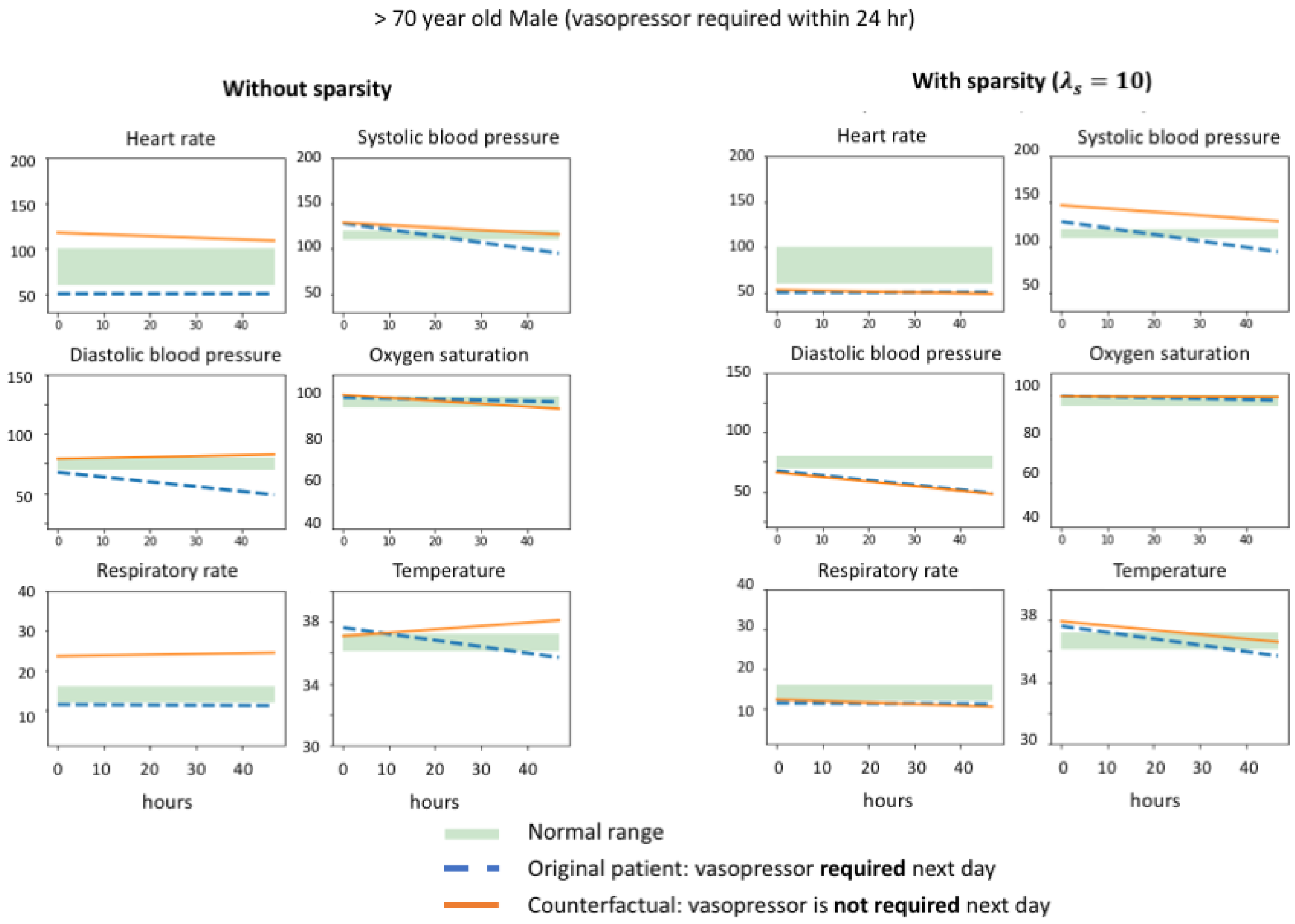


Figure 4.9: Example 2: Counterfactual produced with and without the sparsity term.

Table 4.3: Comparison of [52, 51] and CF VAE. $l_{method} > l_{CFVAE}$ is the fraction of test points where the log likelihood of a CF from prior work exceeded that of our own. We see that in the case of Ventilator, only 2% of the test CF from DICE had a higher likelihood score than our CF. The physician evaluated our counterfactual for plausibility and relevance. The physician's score is the fraction of CF that were deemed both *plausible* and *relevant*. Note that we don't have physician score for REVISE because of its poor validity percentage.

	Ventilator			Vasopressor				
Method	% l_{method} $\geq l_{CFVAE}$	% CF validity	Physician score	% l_{method} $\geq l_{CFVAE}$	% CF validity	Physician score	Time (s) (train)	Time (s) (test)
DICE	2%	100%	40%	2%	100%	20%	0	0.37
REVISE	16%	25%	-	12%	19%	-	10	0.38
CF VAE	-	90%	60%	-	85%	80%	180	0.001

CHAPTER 5
CONCLUSIONS AND FUTURE WORK

This thesis investigated two main shift problems that are a barrier to effectively utilizing machine learning in mHealth: Covariate-space shift and domain shift and presents methods to methodologically address them while providing health researchers with the flexibility to change different aspects of the study. As a first step in addressing the next challenge, this thesis explored bridging the gap between explainability techniques and domain experts.

Specifically, solutions to three problems were presented. First I presented a method to find a common input representation across EMA datasets to overcome covariate-space shift. This is done in the context of predicting non-response to EMAs. Second, I present a domain adaptation method based on self-supervised representation learning that captures the physiological aspects of ECG and PPG signals belonging to different domains. Third, I presented a feed-forward VAE based counterfactual generation model to explain the decision of a given binary classification model. The model generates counterfactuals that are plausible, relevant and convincing as evaluated by a physician.

Through these works, I have demonstrated that it is possible to predict EMA non-response using the history of a participant's mental states and response pattern. Valence features can be constructed as a common input-representation across EMA datasets, while preserving their predictive utility. Self-supervised tasks are helpful in learning aligned representations across domains of ECG and PPG data. Additionally, these tasks are useful as regularizers to help improve predictive model generalization without using target data at training time.

This thesis explored the feasibility of addressing the two covariate shift challenges in the context of EMA non-response prediction, arrhythmia classification from ECG, and respiration rate estimation from PPG. There is enormous potential for ML models for sensor based behavior modification, and there is scope for exploring covariate-space and domain shift in the context of predicting adverse or impulsive behavior.

In this thesis, we considered the problem of covariate-space shift in EMA data collected in different studies. A related challenge arises when different sensor modalities are used in different studies. For example, an mHealth study collecting ECG from participants while another study collecting PPG. Unlike EMA which measures similar mental states through differently phrased questions, ECG and PPG are capturing related but very different physiological measures of the patient. It would be a very challenging and exciting problem if we could develop predictive models that transfer without additional fine-tuning across these modalities. Such a method would be useful in practice since ECG is typically hard to collect with challenges ranging from difficulty of correctly placing electrodes to participant discomfort while wearing the chest band/electrode setup. However, ECG has the advantage of capturing detailed cardiac activity, which can be used along with clinical labels to develop predictive models of health conditions. In comparison to ECG, PPG has a lower barrier in terms of burden on the participants. It can be passively collected through a wrist band, fingertip sensor, ring, phone, or other wearable device. In addition, PPG sensors are cheaper, which makes it easier to deploy in the field. Hence a predictive model trained on ECG transferred to PPG can help improve the reach of predictive models to more people.

REFERENCES

[1] W. Raghupathi and V. Raghupathi, "An empirical study of chronic diseases in the united states: A visual analytics approach to public health," *International journal of environmental research and public health*, vol. 15, no. 3, p. 431, 2018.

[2] J. Basu, R. Avila, and R. Ricciardi, "Hospital readmission rates in us states: Are readmissions higher where more patients with multiple chronic conditions cluster?" *Health services research*, vol. 51, no. 3, pp. 1135–1151, 2016.

[3] C. W. Tsao *et al.*, "Heart disease and stroke statistics—2022 update: A report from the american heart association," *Circulation*, vol. 145, no. 8, e153–e639, 2022.

[4] S. M. Smith and T. O'Dowd, *Chronic diseases: What happens when they come in multiples?* 2007.

[5] K. M. Shaw, K. A. Theis, S. Self-Brown, D. W. Roblin, and L. Barker, "Peer reviewed: Chronic disease disparities by county economic status and metropolitan classification, behavioral risk factor surveillance system, 2013," *Preventing chronic disease*, vol. 13, 2016.

[6] A. R. Quiñones *et al.*, "Racial/ethnic differences in multimorbidity development and chronic disease accumulation for middle-aged adults," *PloS one*, vol. 14, no. 6, e0218462, 2019.

[7] T. O'Neill Hayes, *Understanding the social determinants of health*, https://www.americanactionforum.org/research/understanding-the-social-determinants-of-health, Sep. 2018.

[8] A. H. Mokdad, J. S. Marks, D. F. Stroup, and J. L. Gerberding, "Actual causes of death in the united states, 2000," *Jama*, vol. 291, no. 10, pp. 1238–1245, 2004.

[9] J. M. Rehg, S. A. Murphy, and S. Kumar, *Mobile health*. Springer, 2017.

[10] S. Kumar, M. Al'Absi, J. Beck, E. Ertin, and M. Scott, "Behavioral monitoring and assessment via mobile sensing technologies," *Behavioral Healthcare Technol.: Using Science-Based Innovations to Transform Practice*, vol. 27, pp. 621–624, 2014.

[11] S. Chatterjee *et al.*, "Smokingopp: Detecting the smoking'opportunity'context using mobile sensors," *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies*, vol. 4, no. 1, pp. 1–26, 2020.

[12] S. Hojjatinia *et al.*, "Dynamical system modeling to identify moments of vulnerability based on stress-smoking responses in daily smokers," in *ANNALS OF BE-*

HAVIORAL MEDICINE, OXFORD UNIV PRESS INC JOURNALS DEPT, 2001 EVANS RD, CARY, NC 27513 USA, vol. 56, 2022, S167–S167.

[13] W. T. Riley, D. E. Rivera, A. A. Atienza, W. Nilsen, S. M. Allison, and R. Mermelstein, “Health behavior models in the age of mobile interventions: Are our theories up to the task?” *Translational behavioral medicine*, vol. 1, no. 1, pp. 53–71, 2011.

[14] P. Klasnja and et.al., “Efficacy of contextually tailored suggestions for physical activity: A micro-randomized optimization trial of heartsteps,” *Annals of Behavioral Medicine*, vol. 53, no. 6, pp. 573–582, 2019.

[15] E. Mayor and L. Gamaiunova, “Mobile device-based mindfulness intervention promotes emotional regulation during anticipatory stress,” in *Proceedings of the 5th EAI International Conference on Wireless Mobile Communication and Healthcare*, 2015, pp. 258–262.

[16] *Https://www.mobius.md/blog/2019/03/11-mobile-health-statistics/*, (Date last accessed 03-April-2020).

[17] S. Shiffman, A. A. Stone, and M. R. Hufford, “Ecological momentary assessment,” *Annu. Rev. Clin. Psychol.*, vol. 4, pp. 1–32, 2008.

[18] E. Ertin, N. Stohs, S. Kumar, A. Raij, M. Al’Absi, and S. Shah, “Autosense: Unobtrusively wearable sensor suite for inferring the onset, causality, and consequences of stress in the field,” in *Proceedings of the 9th ACM conference on embedded networked sensor systems*, 2011, pp. 274–287.

[19] L. Giovangrandi, O. T. Inan, D. Banerjee, and G. T. Kovacs, “Preliminary results from bcg and ecg measurements in the heart failure clinic,” in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, IEEE, 2012, pp. 3780–3783.

[20] K. Vandecasteele *et al.*, “Automated epileptic seizure detection based on wearable ecg and ppg in a hospital environment,” *Sensors*, vol. 17, no. 10, p. 2338, 2017.

[21] P. Schmidt, A. Reiss, R. Duerichen, C. Marberger, and K. Van Laerhoven, “Introducing wesad, a multimodal dataset for wearable stress and affect detection,” in *Proceedings of the 20th ACM international conference on multimodal interaction*, 2018, pp. 400–408.

[22] B. Sun and K. Saenko, “Deep coral: Correlation alignment for deep domain adaptation,” in *European conference on computer vision*, Springer, 2016, pp. 443–450.

[23] S. Yang and J. K. Kim, “Statistical data integration in survey sampling: A review,” *Japanese Journal of Statistics and Data Science*, vol. 3, no. 2, pp. 625–650, 2020.

[24] F. Breidt, A. McVey, and W. Fuller, “Two-phase estimation by imputation,” *Journal of the Indian Society of Agricultural Statistics*, vol. 49, pp. 79–90, 1996.

[25] D. Rivers, “Sampling for web surveys,” in *Joint Statistical Meetings*, vol. 4, 2007.

[26] N. Schenker and T. E. Raghunathan, “Combining information from multiple surveys to enhance estimation of measures of health,” *Statistics in medicine*, vol. 26, no. 8, pp. 1802–1811, 2007.

[27] J. C. Legg and W. A. Fuller, “Two-phase sampling,” in *Handbook of statistics*, vol. 29, Elsevier, 2009, pp. 55–70.

[28] Y. Bazi, N. Alajlan, H. AlHichri, and S. Malek, “Domain adaptation methods for ecg classification,” in *2013 international conference on computer medical applications (ICCMA)*, IEEE, 2013, pp. 1–4.

[29] L. Niu, C. Chen, H. Liu, S. Zhou, and M. Shu, “A deep-learning approach to ecg classification based on adversarial domain adaptation,” in *Healthcare*, MDPI, vol. 8, 2020, p. 437.

[30] M. Chen, G. Wang, Z. Ding, J. Li, and H. Yang, “Unsupervised domain adaptation for ecg arrhythmia classification,” in *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, IEEE, 2020, pp. 304–307.

[31] A. Natarajan, G. Angarita, E. Gaiser, R. Malison, D. Ganesan, and B. M. Marlin, “Domain adaptation methods for improving lab-to-field generalization of cocaine detection using wearable ecg,” in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2016, pp. 875–885.

[32] J. Amann, A. Blasimme, E. Vayena, D. Frey, and V. I. Madai, “Explainability for artificial intelligence in healthcare: A multidisciplinary perspective,” *BMC Medical Informatics and Decision Making*, vol. 20, no. 1, pp. 1–9, 2020.

[33] R. Wang *et al.*, “Studentlife: Assessing mental health, academic performance and behavioral trends of college students using smartphones,” in *Proceedings of the 2014 ACM international joint conference on pervasive and ubiquitous computing*, 2014, pp. 3–14.

[34] N. Saleheen *et al.*, “Puffmarker: A multi-sensor approach for pinpointing the timing of first lapse in smoking cessation,” in *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2015, pp. 999–1010.

[35] K. Hovsepian, M. Al’Absi, E. Ertin, T. Kamarck, M. Nakajima, and S. Kumar, “Cstress: Towards a gold standard for continuous stress assessment in the mobile

environment," in *Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing*, 2015, pp. 493–504.

[36] T. J. Trull and U. W. Ebner-Priemer, "Using experience sampling methods/ecological momentary assessment (esm/ema) in clinical assessment and clinical research: Introduction to the special section.," 2009.

[37] F. A. Jean, I. Sibon, M. Husky, T. Couffinhal, and J. Swendsen, "Feasibility and validity of ecological momentary assessment in patients with acute coronary syndrome," *BMC cardiovascular disorders*, vol. 20, no. 1, pp. 1–6, 2020.

[38] T. H. Oreel *et al.*, "Ecological momentary assessment versus retrospective assessment for measuring change in health-related quality of life following cardiac intervention," *Journal of patient-reported outcomes*, vol. 4, no. 1, pp. 1–10, 2020.

[39] R. Schnall, J. Liu, and N. Reame, "Ecological momentary assessment of hiv vs. reproductive health symptoms in women of differing reproductive stages living with hiv," *Menopause (New York, NY)*, vol. 26, no. 12, p. 1375, 2019.

[40] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *European conference on computer vision*, Springer, 2010, pp. 213–226.

[41] B. Sun, J. Feng, and K. Saenko, "Return of frustratingly easy domain adaptation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, 2016.

[42] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7167–7176.

[43] J. Hoffman *et al.*, "Cycada: Cycle-consistent adversarial domain adaptation," in *International conference on machine learning*, Pmlr, 2018, pp. 1989–1998.

[44] P. Wagner *et al.*, "Ptb-xl, a large publicly available electrocardiography dataset," *Scientific data*, vol. 7, no. 1, pp. 1–15, 2020.

[45] A. E. Johnson *et al.*, "Mimic-iii, a freely accessible critical care database," *Scientific data*, vol. 3, no. 1, pp. 1–9, 2016.

[46] F. Doshi-Velez and B. Kim, "Towards a rigorous science of interpretable machine learning," *arXiv preprint arXiv:1702.08608*, 2017.

[47] A. Adadi and M. Berrada, "Peeking inside the black-box: A survey on explainable artificial intelligence (xai)," *IEEE access*, vol. 6, pp. 52 138–52 160, 2018.

[48] A. Abdul, J. Vermeulen, D. Wang, B. Y. Lim, and M. Kankanhalli, “Trends and trajectories for explainable, accountable and intelligible systems: An hci research agenda,” in *Proceedings of the 2018 CHI conference on human factors in computing systems*, 2018, pp. 1–18.

[49] E. Tjoa and C. Guan, “A survey on explainable artificial intelligence (xai): Toward medical xai,” *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[50] S. Wachter, B. Mittelstadt, and C. Russell, “Counterfactual explanations without opening the black box: Automated decisions and the gdpr,” *Harv. JL & Tech.*, vol. 31, p. 841, 2017.

[51] S. Joshi, O. Koyejo, W. Vijitbenjaronk, B. Kim, and J. Ghosh, “Towards realistic individual recourse and actionable explanations in black-box decision making systems,” *arXiv preprint arXiv:1907.09615*, 2019.

[52] R. K. Mothilal, A. Sharma, and C. Tan, “Explaining machine learning classifiers through diverse counterfactual explanations,” in *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, 2020, pp. 607–617.

[53] D. M. Levine, J. Pian, K. Mahendrakumar, A. Patel, A. Saenz, and J. L. Schnipper, “Hospital-level care at home for acutely ill adults: A qualitative evaluation of a randomized controlled trial,” *Journal of General Internal Medicine*, pp. 1–9, 2021.

[54] S. Nagesh *et al.*, “Transformers for prompt-level ema non-response prediction,” *arXiv preprint arXiv:2111.01193*, 2021.

[55] J. Y. E. Park, J. Li, A. Howren, N. W. Tsao, and M. De Vera, “Mobile phone apps targeting medication adherence: Quality assessment and content analysis of user reviews,” *JMIR mHealth and uHealth*, vol. 7, no. 1, e11919, 2019.

[56] D. S. Moskowitz and S. N. Young, “Ecological momentary assessment: What it is and why it is a method of the future in clinical psychopharmacology,” *Journal of Psychiatry and Neuroscience*, vol. 31, no. 1, pp. 13–20, 2006.

[57] I. H. Bell, M. H. Lim, S. L. Rossell, and N. Thomas, “Ecological momentary assessment and intervention in the treatment of psychotic disorders: A systematic review,” *Psychiatric Services*, vol. 68, no. 11, pp. 1172–1181, 2017.

[58] D. Arigo, J. A. Mogle, M. M. Brown, and A. Gupta, “A multi-study approach to refining ecological momentary assessment measures for use among midlife women with elevated risk for cardiovascular disease,” *Mhealth*, vol. 7, 2021.

[59] M. K. Ray, A. McMichael, M. Rivera-Santana, J. Noel, and T. Hershey, "Technological ecological momentary assessment tools to study type 1 diabetes in youth: Viewpoint of methodologies," *JMIR diabetes*, vol. 6, no. 2, e27027, 2021.

[60] S. Nam *et al.*, "Ecological momentary assessment for health behaviors and contextual factors in persons with diabetes: A systematic review," *Diabetes Research and Clinical Practice*, vol. 174, p. 108 745, 2021.

[61] L. S. Valentiner *et al.*, "Effect of ecological momentary assessment, goal-setting and personalized phone-calls on adherence to interval walking training using the interwalk application among patients with type 2 diabetes—a pilot randomized controlled trial," *PloS one*, vol. 14, no. 1, e0208181, 2019.

[62] M. Ono, S. Schneider, D. U. Junghaenel, and A. A. Stone, "What affects the completion of ecological momentary assessments in chronic pain research? an individual patient data meta-analysis," *Journal of medical Internet research*, vol. 21, no. 2, e11398, 2019.

[63] M. May, D. U. Junghaenel, M. Ono, A. A. Stone, and S. Schneider, "Ecological momentary assessment methodology in chronic pain research: A systematic review," *The Journal of Pain*, vol. 19, no. 7, pp. 699–716, 2018.

[64] S. Bruehl, X. Liu, J. W. Burns, M. Chont, and R. N. Jamison, "Associations between daily chronic pain intensity, daily anger expression, and trait anger expressiveness: An ecological momentary assessment study," *PAIN®*, vol. 153, no. 12, pp. 2352–2358, 2012.

[65] C. Lazarides *et al.*, "The association between history of prenatal loss and maternal psychological state in a subsequent pregnancy: An ecological momentary assessment (ema) study," *Psychological medicine*, pp. 1–11, 2021.

[66] E. Shacham *et al.*, "Testing the feasibility of using ecological momentary assessment to collect real-time behavior and mood to predict technology-measured hiv medication adherence," *AIDS and Behavior*, vol. 23, no. 8, pp. 2176–2184, 2019.

[67] S. Moontaha, N. Steckhan, A. Kappattanavar, R. Surges, and B. Arnrich, "Self-prediction of seizures in drug resistance epilepsy using digital phenotyping: A concept study," in *Proceedings of the 14th EAI International Conference on Pervasive Computing Technologies for Healthcare*, 2020, pp. 384–387.

[68] K. S. Willard *et al.*, "Affect in patients with epilepsy undergoing video/eeg monitoring: Retrospective versus momentary assessment and temporal relationship to seizures," *Epilepsy & Behavior*, vol. 8, no. 3, pp. 625–634, 2006.

[69] K. F. Visser, F. Z. Esfahlani, H. Sayama, and G. P. Strauss, "An ecological momentary assessment evaluation of emotion regulation abnormalities in schizophrenia," *Psychological medicine*, vol. 48, no. 14, pp. 2337–2345, 2018.

[70] J. Mote and D. Fulford, "Ecological momentary assessment of everyday social experiences of people with schizophrenia: A systematic review," *Schizophrenia research*, vol. 216, pp. 56–68, 2020.

[71] E. Granholm *et al.*, "What do people with schizophrenia do all day? ecological momentary assessment of real-world functioning in schizophrenia," *Schizophrenia bulletin*, vol. 46, no. 2, pp. 242–251, 2020.

[72] S. D. Targum, C. Sauder, M. Evans, J. N. Saber, and P. D. Harvey, "Ecological momentary assessment as a measurement tool in depression trials," *Journal of psychiatric research*, vol. 136, pp. 256–264, 2021.

[73] N. Hallensleben *et al.*, "Predicting suicidal ideation by interpersonal variables, hopelessness and depression in real-time. an ecological momentary assessment study in psychiatric inpatients with depression," *European Psychiatry*, vol. 56, no. 1, pp. 43–50, 2019.

[74] A. R. Dallman, A. Bailliard, and C. Harrop, "Identifying predictors of momentary negative affect and depression severity in adolescents with autism: An exploratory ecological momentary assessment study," *Journal of Autism and Developmental Disorders*, vol. 52, no. 1, pp. 291–303, 2022.

[75] S. J. Coons, S. Eremenco, J. J. Lundy, P. O'Donohoe, H. O'Gorman, and W. Malizia, "Capturing patient-reported outcome (pro) data electronically: The past, present, and promise of epro measurement in clinical trials," *The Patient-Patient-Centered Outcomes Research*, vol. 8, no. 4, pp. 301–309, 2015.

[76] S. T. Gary, N. Dias, E. Conrad, and K. G. Faulkner, *Home epro compliance in prostate cancer clinical studies.* 2020.

[77] M. F. Pradier, T. H. McCoy Jr, M. Hughes, R. H. Perlis, and F. Doshi-Velez, "Predicting treatment dropout after antidepressant initiation," *Translational psychiatry*, vol. 10, no. 1, pp. 1–8, 2020.

[78] H. Daumé III, "Frustratingly easy domain adaptation," *arXiv preprint arXiv:0907.1815*, 2009.

[79] A. Vaswani *et al.*, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017.

[80] M. Boukhechba *et al.*, "Contextual analysis to understand compliance with smartphone-based ecological momentary assessment," in *Proceedings of the 12th EAI International Conference on Pervasive Computing Technologies for Healthcare*, 2018, pp. 232–238.

[81] V. Mishra, B. Lowens, S. Lord, K. Caine, and D. Kotz, "Investigating contextual cues as indicators for ema delivery," in *Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers*, 2017, pp. 935–940.

[82] Y. Suhara, Y. Xu, and A. Pentland, "Deepmood: Forecasting depressed mood based on self-reported histories via recurrent neural networks," in *Proceedings of the 26th International Conference on World Wide Web*, 2017, pp. 715–724.

[83] A. Mikus, M. Hoogendoorn, A. Rocha, J. Gama, J. Ruwaard, and H. Riper, "Predicting short term mood developments among depressed patients using adherence and ecological momentary assessment data," *Internet interventions*, vol. 12, pp. 105–110, 2018.

[84] Y. Chen and Y. Xue, "A deep learning approach to human activity recognition based on single accelerometer," in *2015 IEEE International Conference on Systems, Man, and Cybernetics*, IEEE, 2015, pp. 1488–1492.

[85] W. Jiang and Z. Yin, "Human activity recognition using wearable sensors by deep convolutional neural networks," in *Proceedings of the 23rd ACM International Conference on Multimedia*, ser. MM '15, Brisbane, Australia: Association for Computing Machinery, 2015, pp. 1307–1310, ISBN: 9781450334594.

[86] C. A. Ronao and S.-B. Cho, "Human activity recognition with smartphone sensors using deep learning neural networks," *Expert systems with applications*, vol. 59, pp. 235–244, 2016.

[87] J. Yang, M. N. Nguyen, P. P. San, X. L. Li, and S. Krishnaswamy, "Deep convolutional neural networks on multichannel time series for human activity recognition," in *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

[88] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[89] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," 2018.

[90] R. Girdhar, J. Carreira, C. Doersch, and A. Zisserman, “Video action transformer network,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 244–253.

[91] R. Ranftl, A. Bochkovskiy, and V. Koltun, “Vision transformers for dense prediction,” *arXiv preprint arXiv:2103.13413*, 2021.

[92] S. Khan, M. Naseer, M. Hayat, S. W. Zamir, F. S. Khan, and M. Shah, “Transformers in vision: A survey,” *arXiv preprint arXiv:2101.01169*, 2021.

[93] G. Paraskevopoulos, S. Parthasarathy, A. Khare, and S. Sundaram, “Multimodal and Multiresolution Speech Recognition with Transformers,” in *Proceedings of 58th Annual Meeting of the Association for Computational Linguistics (ACL 20)*, 2020, pp. 2381–2387.

[94] A. W. Sokolovsky, R. J. Mermelstein, and D. Hedeker, “Factors predicting compliance to ecological momentary assessment among adolescent smokers,” *nicotine & tobacco research*, vol. 16, no. 3, pp. 351–358, 2014.

[95] D. S. Courvoisier, M. Eid, and T. Lischetzke, “Compliance to a cell phone-based ecological momentary assessment study: The effect of time and personality characteristics.,” *Psychological assessment*, vol. 24, no. 3, p. 713, 2012.

[96] M. P. Shiyko, S. Perkins, and L. Caldwell, “Feasibility and adherence paradigm to ecological momentary assessments in urban minority youth.,” *Psychological assessment*, vol. 29, no. 7, p. 926, 2017.

[97] C. M. Turner *et al.*, “Race/ethnicity, education, and age are associated with engagement in ecological momentary assessment text messaging among substance-using msm in san francisco,” *Journal of substance abuse treatment*, vol. 75, pp. 43–48, 2017.

[98] A. Jones *et al.*, “Compliance with ecological momentary assessment protocols in substance users: A meta-analysis,” *Addiction*, vol. 114, no. 4, pp. 609–619, 2019.

[99] H. Sarker *et al.*, “Assessing the availability of users to engage in just-in-time intervention in the natural environment,” in *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2014, pp. 909–920.

[100] S. Aminikhanghahi, M. Schmitter-Edgecombe, and D. J. Cook, “Context-aware delivery of ecological momentary assessment,” *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 4, pp. 1206–1214, 2019.

[101] A. Mehrotra and M. Musolesi, “Intelligent notification systems: A survey of the state of the art and research challenges,” *arXiv preprint arXiv:1711.10171*, 2017.

[102] B.-J. Ho, B. Balaji, N. Nikzad, and M. Srivastava, "Emu: Engagement modeling for user studies," in *Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers*, 2017, pp. 959–964.

[103] B.-J. Ho, B. Balaji, M. Koseoglu, S. Sandha, S. Pei, and M. Srivastava, "Quick question: Interrupting users for microtasks with reinforcement learning," *arXiv preprint arXiv:2007.09515*, 2020.

[104] F. Künzler, V. Mishra, J.-N. Kramer, D. Kotz, E. Fleisch, and T. Kowatsch, "Exploring the state-of-receptivity for mhealth interventions," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 3, no. 4, pp. 1–27, 2019.

[105] W. Choi, S. Park, D. Kim, Y.-k. Lim, and U. Lee, "Multi-stage receptivity model for mobile just-in-time health intervention," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 3, no. 2, pp. 1–26, 2019.

[106] J. Clawson, J. A. Pater, A. D. Miller, E. D. Mynatt, and L. Mamykina, "No longer wearing: Investigating the abandonment of personal health-tracking technologies on craigslist," in *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2015, pp. 647–658.

[107] A. Lazar, C. Koehler, J. Tanenbaum, and D. H. Nguyen, "Why we use and abandon smart devices," in *Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing*, 2015, pp. 635–646.

[108] K. L. Druce, W. G. Dixon, and J. McBeth, "Maximizing engagement in mobile health studies: Lessons learned and future directions," *Rheumatic Disease Clinics*, vol. 45, no. 2, pp. 159–172, 2019.

[109] G. Spanakis, G. Weiss, and A. Roefs, "Enhancing classification of ecological momentary assessment data using bagging and boosting," in *2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI)*, IEEE, 2016, pp. 388–395.

[110] K. Saha, L. Chan, K. De Barbaro, G. D. Abowd, and M. De Choudhury, "Inferring mood instability on social media by leveraging ecological momentary assessments," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 3, pp. 1–27, 2017.

[111] H. Kim, S. Lee, S. Lee, S. Hong, H. Kang, and N. Kim, "Depression prediction by using ecological momentary assessment, actiwatch data, and machine learning: Observational study on older adults living alone," *JMIR mHealth and uHealth*, vol. 7, no. 10, e14149, 2019.

[112] K. van Mens *et al.*, "Predicting future suicidal behaviour in young adults, with different machine learning techniques: A population-based longitudinal study," *Journal of Affective Disorders*, 2020.

[113] Z. King, J. Moskowitz, L. Wakschlag, and N. Alshurafa, "Predicting perceived stress through mirco-emas and a flexible wearable ecg device," in *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*, 2018, pp. 106–109.

[114] E. Choi, M. T. Bahadori, J. Sun, J. Kulas, A. Schuetz, and W. Stewart, "Retain: An interpretable predictive model for healthcare using reverse time attention mechanism," in *Advances in Neural Information Processing Systems*, 2016, pp. 3504–3512.

[115] D. A. Kaji *et al.*, "An attention based deep learning model of clinical events in the intensive care unit," *PloS one*, vol. 14, no. 2, e0211057, 2019.

[116] Y. Tay, M. Dehghani, D. Bahri, and D. Metzler, "Efficient transformers: A survey," *arXiv preprint arXiv:2009.06732*, 2020.

[117] H. Song, D. Rajan, J. Thiagarajan, and A. Spanias, "Attend and diagnose: Clinical time series analysis using attention models," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, 2018.

[118] Y. Li *et al.*, "Behrt: Transformer for electronic health records," *Scientific reports*, vol. 10, no. 1, pp. 1–12, 2020.

[119] P. Rajpurkar, A. Y. Hannun, M. Haghpanahi, C. Bourn, and A. Y. Ng, "Cardiologist-level arrhythmia detection with convolutional neural networks," *arXiv preprint arXiv:1707.01836*, 2017.

[120] G. H. Tison, J. Zhang, F. N. Delling, and R. C. Deo, "Automated and interpretable patient ecg profiles for disease detection, tracking, and discovery," *Circulation: Cardiovascular Quality and Outcomes*, vol. 12, no. 9, e005289, 2019.

[121] Z. I. Attia *et al.*, "An artificial intelligence-enabled ecg algorithm for the identification of patients with atrial fibrillation during sinus rhythm: A retrospective analysis of outcome prediction," *The Lancet*, vol. 394, no. 10201, pp. 861–867, 2019.

[122] V. B. Aydemir *et al.*, "Classification of decompensated heart failure from clinical and home ballistocardiography," *IEEE Transactions on Biomedical Engineering*, 2019.

[123] A. M. Carek, J. Conant, A. Joshi, H. Kang, and O. T. Inan, “Seismowatch: Wearable cuffless blood pressure monitoring using pulse transit time,” *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies*, vol. 1, no. 3, pp. 1–16, 2017.

[124] Y. Sattar and L. Chhabra, “Electrocardiogram,” in *StatPearls [Internet]*, StatPearls Publishing, 2021.

[125] R. Jaafar and A. S. A. Salam, “Portable electrocardiography with cloud based features: A review of current technologies,” in *2019 International Biomedical Instrumentation and Technology Conference (IBITeC)*, IEEE, vol. 1, 2019, pp. 118–122.

[126] Y. Sun, E. Tzeng, T. Darrell, and A. A. Efros, “Unsupervised domain adaptation through self-supervision,” *arXiv preprint arXiv:1909.11825*, 2019.

[127] F. M. Carlucci, A. D’Innocente, S. Bucci, B. Caputo, and T. Tommasi, “Domain generalization by solving jigsaw puzzles,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2229–2238.

[128] E. Lee, A. Ho, Y.-T. Wang, C.-H. Huang, and C.-Y. Lee, “Cross-domain adaptation for biometric identification using photoplethysmogram,” in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2020, pp. 1289–1293.

[129] A. L. Goldberger *et al.*, “Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals,” *circulation*, vol. 101, no. 23, e215–e220, 2000.

[130] M. Wang and W. Deng, “Deep visual domain adaptation: A survey,” *Neurocomputing*, vol. 312, pp. 135–153, 2018.

[131] G. B. Moody and R. G. Mark, “The impact of the mit-bih arrhythmia database,” *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 3, pp. 45–50, 2001.

[132] Y. Chang, A. Mathur, A. Isopoussu, J. Song, and F. Kawsar, “A systematic study of unsupervised domain adaptation for robust human-activity recognition,” *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 4, no. 1, pp. 1–30, 2020.

[133] G. Wilson, J. R. Doppa, and D. J. Cook, “Multi-source deep domain adaptation with weak supervision for time-series sensor data,” in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 1768–1778.

[134] S. An *et al.*, “Adaptnet: Human activity recognition via bilateral domain adaptation using semi-supervised deep translation networks,” *IEEE Sensors Journal*, vol. 21, no. 18, pp. 20 398–20 411, 2021.

[135] T. Knight, C. Harris, M. Mas, O. Shental, G. Ellis, and D. Lasserson, “The provision of hospital at home care: Results of a national survey of uk hospitals,” *International Journal of Clinical Practice*, e14814, 2021.

[136] B. Leff *et al.*, “Hospital at home: Feasibility and outcomes of a program to provide hospital-level care at home for acutely ill older patients,” *Annals of internal medicine*, vol. 143, no. 11, pp. 798–808, 2005.

[137] A. E. Johnson, T. J. Pollard, and R. G. Mark, “Reproducibility in critical care: A mortality prediction case study,” in *Machine Learning for Healthcare Conference*, PMLR, 2017, pp. 361–376.

[138] H. Suresh, N. Hunt, A. Johnson, L. A. Celi, P. Szolovits, and M. Ghassemi, “Clinical intervention prediction and understanding with deep neural networks,” in *Machine Learning for Healthcare Conference*, PMLR, 2017, pp. 322–337.

[139] M. Ghassemi, M. Wu, M. C. Hughes, P. Szolovits, and F. Doshi-Velez, “Predicting intervention onset in the icu with switching state space models,” *AMIA Summits on Translational Science Proceedings*, vol. 2017, p. 82, 2017.

[140] K. E. Henry, D. N. Hager, P. J. Pronovost, and S. Saria, “A targeted real-time early warning score (trewscore) for septic shock,” *Science translational medicine*, vol. 7, no. 299, 299ra122–299ra122, 2015.

[141] S. Nagesh, A. Moreno, H. Ishikawa, G. Wollstein, J. S. Shuman, and J. M. Rehg, “A spatiotemporal approach to predicting glaucoma progression using a ct-hmm,” in *Machine Learning for Healthcare Conference*, 2019, pp. 140–159.

[142] S. Verma, J. Dickerson, and K. Hines, “Counterfactual Explanations for Machine Learning: Challenges Revisited,” Tech. Rep., 2021. arXiv: 2106.07756.

[143] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” *arXiv preprint arXiv:1312.6114*, 2013.

[144] A. Dhurandhar *et al.*, “Explanations based on the missing: Towards contrastive explanations with pertinent negatives,” *arXiv preprint arXiv:1802.07623*, 2018.

[145] E. Delaney, D. Greene, and M. T. Keane, “Instance-based counterfactual explanations for time series classification,” *arXiv preprint arXiv:2009.13211*, 2020.

[146] E. Ates, B. Aksar, V. J. Leung, and A. K. Coskun, “Counterfactual explanations for multivariate time series,” in *2021 International Conference on Applied Artificial Intelligence (ICAPAI)*, IEEE, 2021, pp. 1–8.

[147] R. Guidotti, A. Monreale, F. Spinnato, D. Pedreschi, and F. Giannotti, “Explaining any time series classifier,” in *2020 IEEE Second International Conference on Cognitive Machine Intelligence (CogMI)*, IEEE, 2020, pp. 167–176.

[148] D. Bau, B. Zhou, A. Khosla, A. Oliva, and A. Torralba, “Network dissection: Quantifying interpretability of deep visual representations,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 6541–6549.

[149] R. Caruana, S. Baluja, T. Mitchell, *et al.*, “Using the future to” sort out” the present: Rankprop and multitask learning for medical risk evaluation,” *Advances in neural information processing systems*, pp. 959–965, 1996.

[150] R. Caruana, Y. Lou, J. Gehrke, P. Koch, M. Sturm, and N. Elhadad, “Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission,” in *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, 2015, pp. 1721–1730.

[151] L. Jing *et al.*, “A machine learning approach to management of heart failure populations,” *Heart Failure*, vol. 8, no. 7, pp. 578–587, 2020.

[152] S. Barnes, E. Hamrock, M. Toerper, S. Siddiqui, and S. Levin, “Real-time prediction of inpatient length of stay for discharge prioritization,” *Journal of the American Medical Informatics Association*, vol. 23, no. e1, e2–e10, 2016.

[153] J. A. Bishop *et al.*, “Improving patient flow during infectious disease outbreaks using machine learning for real-time prediction of patient readiness for discharge,” *Plos one*, vol. 16, no. 11, e0260476, 2021.

[154] S. Purushotham, C. Meng, Z. Che, and Y. Liu, “Benchmark of deep learning models on large healthcare mimic datasets,” *arXiv preprint arXiv:1710.08531*, 2017.

[155] T. Gentimis, A. Ala’J, A. Durante, K. Cook, and R. Steele, “Predicting hospital length of stay using neural networks on mimic iii data,” in *2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech)*, IEEE, 2017, pp. 1194–1201.

[156] A. Lajevardi-Khosh, A. Jalali, K. S. Rajput, and N. Selvaraj, “Novel dynamic prediction of daily patient discharge in acute and critical care,” in *2021 43rd Annual*

International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), IEEE, 2021, pp. 2347–2352.

[157] S. Wang, M. B. McDermott, G. Chauhan, M. Ghassemi, M. C. Hughes, and T. Naumann, “Mimic-extract: A data extraction, preprocessing, and representation pipeline for mimic-iii,” in *Proceedings of the ACM Conference on Health, Inference, and Learning*, 2020, pp. 222–235.

[158] C. Doersch, “Tutorial on variational autoencoders,” *arXiv preprint arXiv:1606.05908*, 2016.

[159] C. Burges *et al.*, “Learning to rank using gradient descent,” in *Proceedings of the 22nd international conference on Machine learning*, 2005, pp. 89–96.