ADVANCED TRANSFER LEARNING IN DOMAINS WITH LOW-QUALITY TEMPORAL DATA AND SCARCE LABELS

A Dissertation

Submitted to

Temple University Graduate Board

In Partial Fulfillment
of the Requirements for the Degree
of DOCTOR OF PHILOSOPHY

---

by
Ameen Abdel Hai
May 2024

Examining committee members:

PhD. Zoran Obradovic, Advisory Chair, Department of Computer & Information Sciences,
PhD. Eduard Dragut, Department of Computer & Information Sciences,
PhD. Hongchang Gao, Department of Computer & Information Sciences,
MD. Daniel J. Rubin, Endocrinology, Diabetes, and Metabolism

ABSTRACT

Numerous of high-impact applications involve predictive modeling of real-world data. This spans from hospital readmission prediction for enhanced patient care up to event detection in power systems for grid stabilization. Developing performant machine learning models necessitates extensive high-quality training data, ample labeled samples, and training and testing datasets derived from identical distributions. Though, such methodologies may be impractical in applications where obtaining labeled data is expensive or challenging, the quality of data is low, or when challenged with covariate or concept shifts. Our emphasis was on devising transfer learning methods to address the inherent challenges across two distinct applications.

We delved into a notably challenging transfer learning application that revolves around predicting hospital readmission risks using electronic health record (EHR) data to identify patients who may benefit from extra care. Readmission models based on EHR data can be compromised by quality variations due to manual data input methods. Utilizing high-quality EHR data from a different hospital system to enhance prediction on a target hospital using traditional approaches might bias the dataset if distributions of the source and target data are different. To address this, we introduce an Early Readmission Risk Temporal Deep Adaptation Network, ERR-TDAN, for cross-domain knowledge transfer. A model developed using target data from an urban academic hospital was enhanced by transferring knowledge from high-quality source data. Given the success of our method in learning from data sourced from multiple hospital systems with different distributions, we further addressed the challenge and infeasibility of developing hospital-specific readmission risk prediction models using data from individual hospital systems. Herein, based on an

extension of the previous method, we introduce an Early Readmission Risk Domain Generalization Network, ERR-DGN. It is adept at generalizing across multiple EHR data sources and seamlessly adapting to previously unseen test domains.

In another challenging application, we addressed event detection in electrical grids where dependencies are spatiotemporal, highly non-linear, and non-linear systems using high-volume field-recorded data from multiple Phasor Measurement Units (PMUs). Existing historical event logs created manually do not correlate well with the corresponding PMU measurements due to scarce and temporally imprecise labels. Extending event logs to a more complete set of labeled events is very costly and often infeasible to obtain. We focused on utilizing a transfer learning method tailored for event detection from PMU data to reduce the need for additional manual labeling. To demonstrate the feasibility, we tested our approach on large datasets collected from the Western and Eastern Interconnections of the U.S.A. by reusing a small number of carefully selected labeled PMU data from a power system to detect events from another.

Experimental findings suggest that the proposed knowledge transfer methods for healthcare and power system applications have the potential to effectively address the identified challenges and limitations. Evaluation of the proposed readmission models show that readmission risk predictions can be enhanced when leveraging higher-quality EHR data from a different site, and when trained on data from multiple sites and subsequently applied to a novel hospital site. Moreover, labels scarcity in power systems can be addressed by a transfer learning method in conjunction with a semi-supervised algorithm that is capable of detecting events based on minimal labeled instances.

To my beloved father, Mofeed Abdel Hai

ACKNOWLEDGMENTS

First and foremost, I would like to express my deepest gratitude to my advisor Prof. Zoran Obradovic for giving me the opportunity to join his laboratory, assigning me to world-class projects with leading experts in the field, his continuous guidance, support, and for funding my research and education. The immense assistance and expertise received were invaluable for conducting research. His mentoring approach driven by rigorous examination, collaborative lead for high-quality work, and immense knowledge were essential and key motivations throughout this study.

I would like to extend my profound gratitude to Dr. Daniel J. Rubin for allowing me to participate in the National Institute of Health project. His unwavering guidance, support, and funding have been indispensable. The depth of knowledge and assistance he provided is immeasurable. I would also like to thank Dr. Daniel J. Rubin for serving as an external examining committee member for the final dissertation defense.

Special thanks to Professors Eduard Dragut and Hongchang Gao for serving on the examining committee for my preliminary exams and dissertation defense. Their feedback and suggestions were of valuable assistance to improving my research study.

I am deeply grateful to the principal investigators of research projects I worked on throughout my study. I thank Dr. Daniel J. Rubin for his guidance, support, leadership, and mentorship while taking part of a project funded by the National Health Institutes of Health (NIH) to improving patients' healthcare. I also thank Dr. Mladen Kezunovic and his assistant Dr. Tatjana Djokic for their valuable assistance, leadership, mentorship, and

guidance while taking part of a project funded by the U.S. Department of Energy (DOE) to improving the stability of electrical grids.

I thank my fellow lab mates who have made my journey memorable, enjoyable, and have always been there for me at the Center for Data Analytics and Biomedical Informatics, listed here in alphabetical order: Abdulrahman Alharbi, Branimir Ljubic, Daniel Polimac, Daniel Saranovic, Hussain Otudi, Jovan Andjelkovic, Jumanah Alshehri, Martin Pavlovski, Marija Stanojevic, Mohammad Alqudah, Nima Asadi, Nouf Albarakati, Rafaa Aljarbua, Shoumik Roychoudhury, Shelly Gupta, Saman Enayati, Sidra Hanif, Wilson Diaz and many of which I collaborated and published joint papers with. Special thanks to my dear friend Rosana Rosas for valuable help and support throughout this journey.

Finally, I thank the Department of Computer & Information Sciences for creating engaging and supportive academic environment and for graduate courses that I learned a lot from. Special thanks to Dr. Ola Ajaj for his immense and invaluable support, guidance during my teaching assistantship assignments and throughout my course of study.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Motivation

A multitude of high-impact applications involve predictive modeling of real-world field-recorded temporal data. In various domains, from hospital readmission prediction to improve patients' healthcare, through event detection in power systems to aid stabilize electrical grids, up to prediction in online businesses to improve customers' experience, developing performant traditional machine learning models require large, high-quality training datasets, and a sufficient amount of labeled data to perform optimally. However, such approaches might be impracticable since such assumptions seem to be violated for a plethora of real-world problems, where labeled data is costly or infeasible to obtain, data attributes are of low-quality, and marginal distributions of the source and target domains are dissimilar (covariate shift), or the conditional distributions are different (concept shift). Such problems are found in diverse domains.

In [1], a healthcare domain application for hospital readmission prediction was developed based on real-world electronic health record (EHR) data. In spite of enormous efforts of advanced data cleaning, preprocessing, data representation techniques, and optimal architecture of a deep learning method, the performance of the readmission prediction model was found to be degraded due to low-quality of EHR data consisting of a large number of erroneous and missing data due to the manual data entry scheme. This

method is limited to learning from EHR data drawn from the same distribution and is incapable of utilizing higher-quality EHR data from a source hospital system to enhance prediction on a target hospital since using traditional approaches might bias the dataset if distributions of the source and target data are different. In other words, traditional methods fail to perform cross-domain transfer to enhance prediction on the target task using a related source domain data from a different distribution. Furthermore, collecting data from multiple hospitals and developing performant readmission models for every site may not be feasible for many institutions. Another potential limitation is having sufficient historical data. Developing a readmission risk model based on data from source hospitals to predict readmission on an unseen test hospital using our previous and other conventional approaches is not effective because these methods are not capable of generalizing well to unseen test domains when training and testing distributions are different [6, 79]. Transfer learning methodologies have been investigated in the context of hospital readmission, aiming to enhance the learning of the target population by leveraging insights from a related source population. The studies reported in [80, 81], transfer learning is successfully applied to mitigate the challenges of limited data by utilizing a relevant source dataset. Conversely, the study reported in [82], potential benefits of transfer learning are investigated by assessing the fine-tuning capabilities of pre-trained models within the healthcare domain. Nonetheless, none of the aforementioned methods are capable of generalizing to unseen data collected from multiple hospital systems drawn from different distributions. Hence, there remains a demand for end-to-end models that perform cross-domain knowledge transfer capable of learning from varied source datasets collected from different hospital systems with different distributions and generalizing on previously

unencountered domains in a unified framework, while capturing and maintaining long-term temporal dependencies for hospital readmission.

Event detection in power systems is critical to aid stabilize electrical grids by detecting various types of events using field-recorded Phasor Measurement Units (PMUs) data. This task is a challenging problem for machine learning methods due to scarce and temporally imprecise labels and the inability to automate event labeling in high-volume data such as PMU measurements. Extending event logs to obtain a sufficient number of labeled data is costly and often infeasible to obtain since it requires manual observations by a domain expert in the field. Developing performant traditional machine learning detectors based on fully supervised approaches might be infeasible since such detectors rely heavily on labeled data, which when done manually may be labor-intensive and hence prohibitively expensive. Moreover, supervised learning methods assume that the marginal distribution of the source training data and target test data are identical (no covariate shift assumption), which PMU data violate. Event detection can also be deemed as an unsupervised learning task, however unsupervised approaches fail to correct mistakes made by using labeled data and utilize the underlying assumption that events occur infrequently, meaning they fall in low-density regions of the instance space, which PMU data regularly violate this assumption. Supervised, and unsupervised approaches are infeasible for detecting events from PMU data [2-5].

Therefore, building predictive models from such data calls for devising advanced knowledge transfer-based methods to enhance predictions in domains where data covariate and concept shifts are a challenge, labels are scarce and difficult to obtain, and data are of low-quality.

1.2 Contributions

This dissertation revolves around proposing knowledge transfer-based frameworks to address inherent challenges in real-world, high-impact applications in two different domains, healthcare and power systems. The main contributions of the conducted studies are presented in the following.

Healthcare. Hospital readmission risk prediction methods for patients with diabetes based on electronic health record (EHR) data were developed to address gaps in literature. Advanced data preprocessing and representation techniques were utilized for the heterogenous temporal EHR data collected from five academic health systems, encompassing urban, suburban, and rural areas in Pennsylvania or Maryland. Models were developed using EHR data as defined by the Patient-Centered Clinical Research Network (PCORnet®) Common Data Model (CDM), which standardized EHR data across sites [1].

The extracted heterogenous EHR data encompass inherent challenges that have modelled difficulties for machine learning algorithms to effectively learn. This was primarily because data drawn from varied distributions and included low-quality elements. Challenges include non-uniform number of recordings of diagnoses, procedures, and vitals, in addition to missingness and erroneousness (i.e., outliers) owing to the manual entry scheme and the process of data extraction and merging from different databases. Moreover, the data exhibited distinct characteristics. For instance, in terms of sociodemographic, 4.9% of patients at a suburban site used in this study were Hispanic, in contrast to 22% at an urban site. The mean number of diagnostic recordings at one site was just 2.5, while other sites reported between 10 to 19 recordings. The non-uniform complicated the

preprocessing stage. Further differences were observed in categories such as race and tobacco use.

Contributions were as follows:

1) To develop deep learning models for the prediction of unplanned, all-cause 30-day readmission.
2) To compare the performance of the deep learning models to traditional machine learning models.
3) To explore model performance across a range of prior EHR encounters from 1 to 100 being included in model development.
4) To compare a deep learning model developed using a subset of a laboratory tests selected by domain knowledge with a deep learning model developed using all available laboratory test.

Upon conducting extensive experiments to evaluate the model, data representation techniques and the utilization of sequential deep learning models were optimal to model EHR data. Deep learning models achieved 0.80 evaluated using F-1 score metric. However, building a model using the same techniques used on higher quality data collected from a rural academic hospital system, Penn State University Hospital System resulted in 91% F1-score. The degradation in F-1 score on Temple University Hospital System was due to low-quality data.

Utilizing EHR data from a source hospital system to enhance prediction on a target hospital using traditional approaches enlarge dataset bias which might deteriorate performance due to distributional difference of the source and target dataset, resulting in statistically unbounded risk for the target task. This was confirmed by training a model

using both source and target data which resulted in a lower F-1 score tested on Temple University Hospital (0.79% F-1 score). Traditional approaches are designed for a specific data type, and not capable of generalizing to other temporal data. Due to the need for an end-to-end model to perform cross-domain spatial knowledge transfer and predictive learning in a unified learning framework while capturing temporal dependencies for hospital readmission, we propose the following.

To address the aforementioned limitations and challenges, we propose an early readmission risk temporal deep adaptation network, ERR-TDAN, to perform cross-domain spatial knowledge transfer from EHR data of different sites and perform predictive learning. Motivated by the success of the Deep Adaptation Network (DAN) in numerous transfer learning tasks which utilizes convolutional neural network (CNN) for computed vision tasks, we employed the idea of learning transferable features of temporal data by matching the source and target domain distributions in the latent feature space. We tailored it for the hospital readmission using EHR data and optimized for the target task. Results conducted show that ERR-TDAN might enhance hospital readmission prediction by performing cross-domain knowledge transfer utilizing higher-quality data from a related source domain [6]. The aims of this study were as follows:

1) To develop a hospital readmission framework using EHR data that transfers knowledge between a rural academic hospital and an urban academic hospital to enhance predictions on the urban academic hospital.
2) To study the optimal amount of retrospective EHR data needed for future predictions.
3) To study the duration of optimal performance.

Given the success of our method in learning from data sourced from multiple hospital systems with different distributions, we further addressed the challenge and infeasibility of developing hospital-specific readmission risk prediction models using data from individual hospital systems. Herein, we devised an additional readmission risk prediction model to address the aforementioned limitations and challenges. We propose an early readmission risk domain generalization network, ERR-DGN, to perform cross domain knowledge transfer from electronic health record (EHR) data of different health systems to facilitate predictive learning. Motivated by the success of our previous study aforementioned and reported in [6], we employed the idea of learning transferable features of the EHR data by matching multiple source distributions in the latent space to generalize and enhance predictions on an unseen target task. ERR-TDAN [6] takes as an input two sites (i.e., source and target) and requires historical training data from both. In contrast, we tailored ERR-TDAN to learn transferable features of multiple source datasets to predict rehospitalization risk on an unseen target hospital where data distribution might be significantly different from data at previously observer hospitals. We hypothesized that this novel approach would improve hospital readmission risk predictions among people with diabetes for a previously unobserved target domain. We further supplemented our experimental findings by studying the number of source sites needed to enhance predictions on an unseen target domain and examined model performance over time to avoid performance degradation due to data drift over time.

Power systems. Event detection methods to detect line fault, transformer outages, and frequency events from PMU data collected from the Western and Eastern Interconnections of the United States were developed to address the following challenges: labels scarcity;

and spatial and temporal data drift. A multitude of models were developed based on transfer learning techniques in conjunction with semi-supervised detector to avoid and mitigate the labor-intensive manual labeling. Transfer learning methods were developed that leverages a small number of well-defined instances from one task to another within the Western Interconnection. Conducted experiments demonstrate that transfer learning methods are applicable for PMU data and can detect events without having to rely on event logs or extensive number of labels of PMU data. Further, the applicability of utilizing transfer learning techniques on PMU data were validated by utilizing statistical methods to compare the similarity between two continuous distribution functions. Results indicate that the source and target distributions are different which makes it applicable for transfer learning and violates the assumptions of fully supervised methods [2]. In this study, the following research questions were addressed:

1) The effect of varying percentages of labeled data were studies to determine the minimum number of labeled data needed. The proposed transfer learning method found to be effective and can detect events based on only 20 labeled data instances while state-of-the-art supervised and unsupervised approaches performed poorly.
2) A comparative analysis was performed to compare the transfer learning method versus baseline anomaly detectors.
3) The effect of varying time window sizes was studies to determine the optimal window dimension to detect events from PMU data. 4) Varying ways of leveraging knowledge from a source task to detect events from the target task were studies, including leveraging knowledge temporally where related instances from the past were leveraged to detect events from the future, and using leveraging labeled data

from a selected set of PMUs by domain knowledge to detect events from another set of PMUs. This experiment demonstrates and validates data drift over time degrading the performance of traditional machine learning approaches.

The aforementioned study is limited to transferring related labeled instances within the same interconnection since the Western and Eastern Interconnections comprise of different number of PMUs resulting in a non-uniform dimensions of feature vectors. Thus, we extended this study to leverage related data instances from one interconnection to detect events in another. Experiments conducted show that the proposed transfer learning method is more feasible than alternative baselines since distributions of the Western and Easter Interconnections data differ, which degraded the performance of traditional Machine learning methods. Moreover, conducted experiments demonstrate superior performance over various state-of-the-art ML algorithms (unsupervised, semi-supervised, and supervised) when leveraging labeled data from one power system to detect events in another, which mitigate the labor-intensive and costly manual labeling efforts [3]. In this study, the following research questions were addressed:

1) Performed distribution comparison of both interconnections and validated transfer learning assumptions.
2) Performed a comparative study to compare the transfer learning method versus baseline event detectors.
3) Studied the effect of using various quantities of labeled data to find the minimum number of labeled data instanced needed from the one Interconnection to detect events from another.

1.3 Journal and Conference Publications

1.3.1 Healthcare

- Abdel Hai, A., et al. (in-review). "Domain Generalization for Enhanced Predictions of Hospital Readmission on Unseen Domains." *Artificial Intelligence in Medicine.*
- Abdel Hai, A., Weiner, M.G., Livshits, A., Brown, J.R., Paranjape, A.P., Obradovic, Z., & Rubin, D.J. (2023). "Spatial Knowledge Transfer with Deep Adaptation Network for Predicting Hospital Readmission." In *Proceedings of the 21st International Conference on Artificial Intelligence in Medicine*, Portoroz, Slovenia, June 2023.
- Abdel Hai, A., Weiner, M.G., Paranjape, A.P., Livshits, A., Brown, J.R., Obradovic, Z., & Rubin, D.J. (2022). "Deep Learning vs Traditional Models for Predicting Hospital Readmission among Patients with Diabetes." In *Proceedings of the AMIA 2022 Annual Symposium,* Washington, DC, Nov. 2022.

1.3.2 Power Systems

- Abdel Hai, A., et al. (2021). "Transfer Learning for Event Detection from PMU Measurements with Scarce Labels." in *IEEE Access*, 9, 127420-127432.
- Abdel Hai, A., Mohamed, T., Pavlovski, M., Kezunovic, M., & Obradovic, Z. (2022). "Transfer Learning on Phasor Measurement Data from a Power System to Detect Events in Another System." In *Proceedings of the 21st IEEE International Conference on Machine Learning and Applications (ICMLA),* Nassau, Bahamas, 1567-1572.

1.4 Organization

The remainder of this report is organized as follows. Chapter 2 discusses the challenges of labels scarcity and distributional differences between source and target domains in power system application domain and proposes transfer learning methods to avoid and mitigate the labor-intensive manual labeling efforts. Chapter 3 discusses a hospital readmission risk prediction method for patients with diabetes. This section provides advanced data preprocessing and representation techniques and serve as foundation for study reported in Chapter 4 and Chapter 5. Chapter 4 discusses the proposed early readmission risk temporal domain adaptation network, ERR-TDAN that is tailored for EHR data and optimized on the target domain; it performs cross-domain knowledge transfer while capturing temporal dependencies of EHR data. Chapter 5 presents the proposed early readmission risk domain generalization network, ERR-DGN, a domain generalization method, tailored for hospital readmission. ERR-DGN enhances readmission risk predictions when applied on an unseen target domain using EHR data collected from varied hospital systems with different distributions. Chapter 6 concludes the dissertation and outlines possible directions for future research.

CHAPTER 2

TRANSFER LEARNING FOR EVENT DETECTION FROM PMU MEASUREMENTS WITH SCARCE LABELS

2.1 Introduction

The stored data collected by the Phasor Measurement Units (PMUs) at the electric utilities in the USA has increased to hundreds of terabytes in the last few years [7]. In the past decade, PMU data have been used extensively for post- mortem analysis in case of system-wide disturbances. In recent years, utilities have been interested in investigating ways to increase the value of the stored PMU data through novel applications of the machine learning models for improved situational awareness and predictive decision-making capabilities [8].

Event detection is an essential task that involves detecting instances in a dataset that significantly deviate from the norm [9]. The increase in the volume of PMU data is making it more challenging to quickly analyze a large number of historical recordings.

Event detection can be deemed as an unsupervised learning task [10]. Usually, unsupervised approaches utilize the underlying assumption that events occur infrequently, meaning they fall in low-density regions of the instance space, or they are distant from normal events to identify them. However, PMU data regularly violate this assumption, affecting the performance of unsupervised approaches (e.g., maintenance events can occur infrequently and irregularly, but are considered normal). Labeled data allow detectors to

correct the errors made by unsupervised approaches. Unfortunately, a fully supervised learning approach to event detection relies heavily on labeled data, which when done manually may be labor-intensive and hence prohibitively expensive.

Contribution. To avoid the expense of extensive manual labeling, semi- supervised approaches to event detection are often used in conjunction with active learning to efficiently collect labels [11]. However, when dealing with a large amount of PMU data, utilizing active learning to assign labels for each individual instance might be infeasible. To reduce the required labeling effort, we employ transfer learning techniques to leverage a small number of well-labeled instances from one task to another without additional labeling effort. The contribution is in the enhancement of the transfer learning method using a non-redundant approach that does not select duplicates/similar instances in order to improve computation efficiency. Additionally, we improved the semi-supervised detector by using an alternative similarity measure that is more applicable to the dimensionality of the PMU data. We demonstrate that a transfer learning method is applicable for PMU data and can detect events without having to rely on an extensive number of labels or event logs of PMU data. This technique may be propagated to other situations where some of the events' data from one power system may be applied to enhance learning in another. Our approach outperforms state-of-the-art machine learning algorithms from varying learning types (unsupervised, semi- supervised, and supervised) on a large benchmark when developing the model from a large dataset that requires intensive event labeling effort. Experiments conducted show that the employed transfer learning method may mitigate the need of manual labeling and is capable of detecting events with minimal labeled data instances.

Organization. The remainder of this paper is organized as follows. Section II provides and describes the related work. Section III describes and provides preliminaries on transfer learning for event detection. Section IV describes the methodology used to conduct experiments, evaluate and elucidate the proposed transfer learning method, and provides insights into the comparison with a variety of learning types of algorithms used in the literature. Section V describes the data preprocessing techniques used in the experiments. The experimental setup is outlined in Section VI. Section VII presents the experimental results and discussion. Finally, Section VIII concludes the paper. Section IX discusses future work. References are provided at the end.

2.2 Literature Review

A variety of studies have investigated ways to reduce the size of the PMU dataset by different means of dimensionality reduction and feature engineering to address the increase in the volume of PMU data. The dimensionality reduction method based on Principal Component Analysis was used in [12] for early online event detection, and in [13] to detect and analyze complex cascading events. The feature engineering method based on the Minimum Volume Enclosing Ellipsoid was reported in [14]. Several studies have used signal transform methods, such as the fast variant of Discrete S-Transform [15- 17], or wavelet analysis [16-19]. Fast event detection based on Detrended Fluctuation Analysis on Big PMU Data was developed in [20]. Domain-specific shapelets were investigated for event detection and classification in [16,17]. In [21] the Dynamic Programming based Swinging Door Trending was used. Signal Energy Transform was used to detect and classify faults in [22]. Several machine learning models were tested in these studies:

Agglomerative Hierarchical Clustering [14], Extreme Learning Machine classifier [15], K-Nearest Neighbor [16,17,23], Support Vector Machine (SVM) [16,17,23], Decision Tree [23], Convolutional Neural Network [19]. Transfer learning has been applied to several power systems applications in recent years, such as transient stability prediction in [24], detection of oscillation events in [25], and detection of high-impedance faults in distribution systems in [26]. Studies [24-26] demonstrate the applicability of transfer learning to a variety of power system problems. Our study extends the benefits of using transfer learning to solve the problem of transmission system event detection from an exceedingly small number of labeled events based on PMU data.

2.3 Transfer Learning for Event Detection from a Small Number of Labels

While event detection tasks would benefit from labeled data, it is often done using an unsupervised approach since assigning labels across all the events manually can be time-consuming and hence costly. The downside is that the unsupervised detectors do not benefit from labeled data that provide the possibility of correcting errors made by the unsupervised detectors. On the other hand, supervised learning algorithms rely on a sufficient number of labeled data. Thus, supervised, and unsupervised learning algorithms are infeasible for event detection tasks when labels are scarce and temporally imprecise. Transfer learning can be utilized to leverage a small number of related labeled data instances from a related task to the target task. Related instances can aid semi-supervised learning algorithms to detect events based on minimal labeled data, since it only selects and transfers tasks that are similar to instances in the target set.

Often, transfer learning is used in conjunction with semi- supervised learning algorithms, since semi-supervised algorithms assume only a limited amount of labeled data instances for training are available. Hence, semi-supervised learning algorithms are employed when labeled data instances are scarce and difficult to obtain. Semi-supervised learning algorithms aim to train a classifier from both the labeled and unlabeled data samples in order to achieve better performance than supervised learning algorithms trained on labeled data only.

The aim of transfer learning is to learn a model for the unlabeled dataset of the target domain given labeled data from a related dataset of the source domain [27]. Since this study concerns event detection, the task is to compute and assign an anomaly score to each time window (data instance) in the target dataset that quantifies how anomalous the time window is based on similarity measures; assigning an anomaly score to a time window can be compared with a predefined threshold to classify whether an anomalous event exists within a given time window [27]. We use D_s to denote the source dataset, which contains labeled time windows, and D_t to denote the target dataset, which contains unlabeled time windows of events to be classified as either normal or anomalous events. We use x_s to refer to a time window from the source dataset, and x_t to refer to a time window from the target dataset.

There are three important assumptions for transfer learning techniques to be considered when applied to event detection tasks [28]. First, the source and target datasets were obtained from the same m-dimensional feature space. Second, the marginal distributions of the source and target datasets differ (covariate shift assumption). A covariate shift assumption occurs when dissimilar behaviors are observed in either domain. Third, the

conditional distributions can differ due to changes in context, meaning the same behavior might have a different meaning in the two domains (concept shift assumption). Assumptions two and three complicate the transfer task.

2.4 Methodology

To demonstrate the performance of the utilized transfer learning method, a comparative analysis with a multitude of event detection algorithms with varying learning types used as a baseline was performed. Additionally, different datasets were used for experiments with varying splits of the data and window dimensions.

2.4.1 Unsupervised Learning

Unsupervised learning algorithms aim to identify hidden patterns without using any labeled data samples. Thus, unsupervised learning algorithms are capable of learning without an error signal to assess and evaluate the performance of the model. Since unsupervised learning algorithms do not require any labels during learning and identifying hidden patterns, event detection tasks using this method can be beneficial when labels are not available [29]. However, unsupervised learning algorithms utilize the fundamental assumption that events occur infrequently, and PMU data often violate this assumption [11]. The lack of labeled data instances that provide the option to correct the errors made by unsupervised detectors degrades the performance of the algorithms.

As a part of the comparison study, an event detection experiment was performed using two unsupervised learning algorithms, namely: 1) the k-nearest neighbor outlier (kNNO) detection algorithm that computes for each data point the anomaly score as the distance to its k-nearest neighbors in the dataset [30], and 2) the isolation nearest neighbor ensembles

(iNNE) algorithm that computes for each data point the anomaly score roughly based on how isolated the point is from the rest of the data [31]. They learn a structure on the training data without incorporating any labels into the models. Event detection is performed on the test dataset to classify data samples as anomalous or normal events.

In order to assess the performance of the algorithms, the predicted labels were compared to the ground truth (actual) labels obtained by visual inspection by a domain expert.

2.4.2 Supervised Learning

Supervised learning is based on training a model using previously observed labeled data samples and assuming that the marginal distribution of the source training data and the target test data are identical (no covariate shift assumption). Supervised learning algorithms tend to rely heavily on learning data samples and require a sufficient amount of training data before performing classification, which can be infeasible in event detection tasks [29]. The more complex the problem and the models are the more training data is required.

We employed state-of-the-art and most common conventional supervised learning algorithms to compare with other learning types. We used scikit-learn library for Machine Learning in Python [32]. A variety of classification algorithms from this library were utilized, including Multilayer Perceptron (MLP), Logistic Regression (LR), K-Nearest Neighbor (KNN), Support Vector Machine (SVM).

2.4.3 Semi-Supervised Learning

The semi-supervised learning concept is in between unsupervised and supervised learning. Semi-supervised classification algorithms aim to train a classifier from both the labeled and unlabeled data samples, such that they achieve better performance than the supervised or unsupervised learning algorithms. There are many practical benefits in using semi-supervised learning, especially, when labeled data instances are scarce and difficult to obtain, since such algorithms assume only a limited amount of labeled data instances for training are available. Semi-supervised learning algorithms might perform as well as supervised learning algorithms, but with much fewer time-labeled data instances, which is beneficial in event detection tasks to reduce annotation effort resulting in reduced implementation costs [33].

Two semi-supervised learning algorithms that do not rely entirely on labels obtained from event logs or by visual inspection to classify data samples as normal or anomalous events were utilized:

1) the semi-supervised k-nearest neighbor anomaly (SSKNNO) detection algorithm, which is a combination of the well-known kNN (i.e., unsupervised learning) classifier and the kNNO (k-nearest neighbor outlier detection) (i.e., supervised learning) method [11]. Since SSKNNO is a distance-based method that relies on Euclidean similarity measure, the number of labeled instances does not affect the learning process. Having as minimum as one labeled data instance from each pattern of signals or type of event should be sufficient for the algorithm to detect events. The algorithm uses an unsupervised setting when a similar labeled data instance is not available in the training data.

2) the semi-supervised detection of outliers (SSDO) algorithm, which computes an unsupervised prior anomaly score, and then, corrects this score with the known label information. It is based on constrained k-means clustering [34]. These algorithms take a partially labeled dataset that consists of three labels: unknown (0), event (1), normal (-1), and assigns a binary label (-1, 1) to each unknown instance in the dataset.

The performance of the algorithm was assessed by comparing the predicted labels to the ground truth labels. Small proportions of labeled data samples combined with unlabeled data samples were used during training.

2.4.4 Transfer Learning + Semi-Supervised Learning

We formulate the event detection task using transfer learning technique as:

- **Input:** D_s and D_t from the same feature space. Where D_s denotes a source dataset containing labeled time windows and D_t denotes a target dataset containing unlabeled time windows $D_{t,}$;
- **Do:** Compute an anomaly score for every time window in D_t based on D_t and a subset of the related time windows in D_s;
- **Output:** y labels (predictions) indicating whether a time window in D_t contains normal or anomalous behavior.

A two-step transfer learning approach used in our study is based on a recently introduced LocIT algorithm [11], that was not yet applied on PMU data. First, the algorithm takes as an input a labeled source dataset D_s. Then, it selects a subset from the labeled

source time windows to transfer to the unlabeled target dataset D_t. If the local data distribution of a certain time window is similar in both the source and target datasets, the algorithm transfers the time window from the source to the target domain. LocIT utilizes unsupervised learning techniques since labels for time windows in the target dataset are not available and the labeled time windows in the source dataset should not influence the transfer decision. Second, the algorithm computes an anomaly score using a semi-supervised learning algorithm based upon nearest-neighbor techniques that consider both the related time windows that were selected and transferred from the source D_s and the unlabeled target time windows [11].

LocIT selects and transfers similar time windows from D_s to D_{trans}, where D_{trans} is a subset that contains the selected labeled time windows for transfer [11]. Let $D^* = D_t \cup D_{trans}$, where D^* is a dataset containing the transferred time windows combined with the target unlabeled time windows. D^* is a partially labeled dataset, where time windows from D_{trans} are labeled as an event (1) or normal (-1), and D_t time windows are labeled as unknown (0). Then, a semi-supervised SSKNNO algorithm takes D^* as input and classifies each unknown time window as an anomalous event or normal, indicating whether a given event occurred in a given time window or healthy signal respectively. This process is further illustrated in the flowchart in Figure 2.1

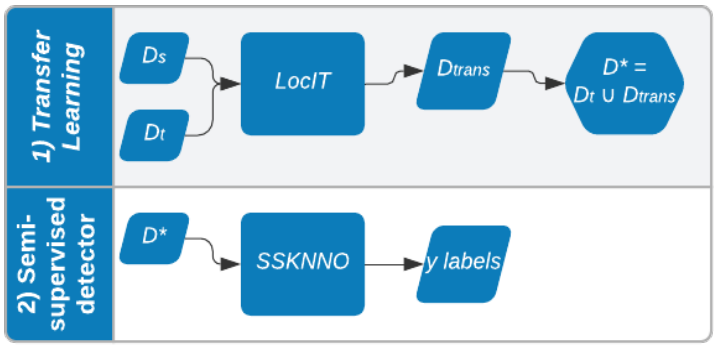

Figure 2.1: Flowchart that illustrates the two-step process of event detection using transfer learning + semi-supervised detector.

Local Structure of time windows. LocIT algorithm defines the localized source distribution for a given source time window x_s using the subset $N_\psi(x_s, D_s)$ of the nearest neighbor ψ of x_s in D_s; and defines the localized target distribution based upon the subset $N_\psi(x_s, D_t)$ of x_s 's ψ nearest neighbor in D_t. Where ψ controls the strictness of the transfer.

The higher the value of ψ is (i.e., 1.0), the stricter the transfer is. If ψ is 0, the algorithm ignores the differences of local distribution and considers the complete global structure of D_s and D_t to determine the transfer.

The algorithm transfers a time window from the source subset to the target subset if the distributions of both subsets are sufficiently identical. The similarity measure (i.e., *location distance*) used to compare the first and second order statistics of $N_\psi(x_s, D_s)$ and $N_\psi(x_s, D_t)$ is defined as:

$$d_1(N_1, N_2) = \left\| \frac{1}{k} \left(\sum_{x_i \in N_1} x_i - \sum_{x_j \in N_2} x_j \right) \right\|_2 . \qquad (2.1)$$

The *location distance* used in equation (2.1) is the l2-norm of the difference of the arithmetic mean (i.e., centroids) between two neighborhood subsets $N_1$and N_2. Large values of d_1 reduce the chance of meaningful transfer.

The distance between the covariance matrices of two neighborhood subsets (i.e., *correlation distance*) is defined as:

$$d_2(N_1, N_2) = \frac{\|C_{N_1} - C_{N_2}\|_F}{\|C_{N_1}\|_F} \qquad (2.2)$$

Where $\|\cdot\|_F$ is the Frobenius norm and C is the covariance matrix. The Frobenius norm was considered since N_1 and N_2 are matrices. Large values of d_2 indicate that the localized distributions of the source and target subsets are different, which decreases the chance of a meaningful transfer.

Learning the Transfer Function. In order to transfer a time window from the source D_s to target subset D_t, the transfer function decides whether to transfer the time window based upon combining the values of d_1 and d_2. LocIT utilizes an SVM classifier that learns on the target distribution using the target data only to serve as the transfer function. SVM predicts whether a time window in the source instance fits in the target domain by leveraging the smoothness assumption, having the meaning that neighboring target time windows have similar localized distributions while the farthest time windows have dissimilar localized distributions. Hence, the negative training instances are generated by computing for every time window in the target subset a feature vector consisting of the distances between the neighborhood subsets of x_t and its farthest neighbor. The one positive

training instance is generated for each instance x_t by finding its nearest neighbor in the target subset and computing d_1 and d_2 on the target subset. Finally, once the SVM classifier is trained on the target subset using both the negative and positive training instances, each instance from the source subset can be predicted to check whether it belongs to the target. If it belongs, the algorithm transfers it and adds it to D_{trans}.

2. 5 Data Processing

2.5.1 PMU Data

The PMU dataset used for testing was provided in the Apache Parquet database. The original dataset contains measurements from 38 PMUs from the Western Interconnection of the U.S.A. captured over a period of two years (2016-2017). The dataset was anonymized by the provider. Geographical locations of the PMUs and the network topology information are not made available. The data are collected with two frame reporting rates per second (fps), 30 fps, and 60 fps, and contain measurements from PMUs located at several voltage levels in the transmission network. This dataset corresponds to a variety of event types, including line fault transformer outages, and frequency events. Some data quality issues, such as missing data, data duplicates, and outliers were observed but did not have a significant impact on our method.

2.5.2 Event Log

The event log received from the data provider contains manually created labels with only an approximation of the event start time with a precision of 1 minute. We referred to these labels as temporally imprecise. The time labels were not created based on the PMU time reference; thus, some events were mislabeled and did not occur at the location of the

PMUs used in this study. Using such limited labels makes it challenging to temporally extract more precise PMU labels from the event log.

To extract temporally more precise labels, we considered a set of labels created based upon visual inspection of the PMU-recorded signals by a domain expert on our team. The domain expert on our team relabeled the data to ensure that the labels were accurate and precise, since the initial labels (event log) received were inaccurate. The different sets of labels (1-minute, 30-seconds, 10-seconds, 5-seconds, 2-seconds) for event and normal operations identified through this study are presented in Table 2.1.

Table 2.1: Number of labels per category and window selection method.

Event Log	**# Event Labels**	**# Normal Labels**	**Event Start Time**	**Event End Time**
1-min Labels	1033	923	ST_{VI} – 5 sec	ST_{VI} + 55 sec
30-sec Labels	1038	1846	ST_{VI} – 2 sec	ST_{VI} + 28 sec
10-sec Labels	1038	1846	ST_{VI} – 1 sec	ST_{VI} + 9 sec
5-sec Labels	1038	1846	ST_{VI}	ST_{VI} + 5 sec
2-sec Labels	1038	1846	ST_{VI}	ST_{VI} + 2 sec
ST_{VI} - *start time of the event based on visual inspection.*				

2.5.3 Feature Extraction

We defined the *Rectangle Area* (*RA*) features extracted per PMU for each time window. No data cleansing was performed on the PMU dataset from the chosen 38 PMUs prior to the feature extraction. The *RA* feature, created using the frequency and positive-sequence voltage magnitude measurements, is defined as:

$$RA_{PMU,TW} = (f_{max} - f_{min}) * (V_{max} - V_{min}) \quad (2.3)$$

where f_{max} and f_{min} are the maximum and minimum frequency values, and V_{max} and V_{min} are the maximum and minimum positive sequence voltage magnitude recorded by the selected *PMU* device, inside the selected time window *TW*.

After feature extraction, only minor cleansing of outliers was performed by removing *RA* values that were too large to be possible. Only 11 *RA* values were discarded. They were replaced with zeros. The impact of missing data is negligible. If at least two data points were present inside a time window, the RA was calculated. For example, in the case of the 1-minute window on a 30 fps PMU, we only need 2 out of 1800 ($30\ fps * 60\ sec$) points to be able to calculate the *RA*. In case there is only one data point within a time window, *RA* is set to zero. Data duplicates do not have any impact on this method since the minimum and maximum values of voltage and frequency are not affected by the duplicates.

The RA feature is sufficient to capture whether an event has occurred within a time window. The RA feature is limited to detecting events and is not suitable for classifying event types. The RA feature was used since it yielded the best performances among multiple data processing techniques that were tested. Furthermore, aggregated RA features allow the utilization of simple and efficient similarity measures to compute distances between time windows to find the nearest and farthest neighbors.

Data processed based on the rectangle area were standardized using *StandardScaler,* which subtracts the mean, and then scales each feature to unit variance.

2.5.4 Temporal Split

A set of 38 PMUs that contain time windows collected over a span of two years, 2016 and 2017 was split into two subsets, where the first subset was used as a source dataset for transfer learning, D_s, and the second subset is the target for transfer learning, D_t. The split

between two the subsets was based on the temporal split between the years 2016 and 2017. Knowledge was leveraged and transferred from the year of 2016, D_s, to the target subset D_t, which contains time windows collected from the year of 2017. D_t is a fixed dataset that contains all windows from 2017 in all the experiments conducted. Proportions of labeled time windows were randomly selected from D_s, and combined with target time windows, D_t, in a dataset D^*, which is a partially labeled dataset that contains the transferred related labeled windows from D_s and windows to be classified as anomalous or normal event, D_t. While proportions of labeled time windows were randomly selected, it was ensured that the selected time windows result in a balanced subset containing both anomalous and normal events.

For the unsupervised, semi-supervised, and supervised classifiers, the set of 38 PMUs was also split temporally, hence, training data containing data time windows from the year of 2016, and test data containing data time windows from the year of 2017. Since these classifiers do not transfer related time windows, classifiers were trained on entire time windows from 2016, and tested/classified time windows from the future, hence, time windows collected from the entire 2017.

Features for a certain time window were combined into a feature vector that contains 38 RA features, one feature for each observed PMU. Labels y are created for each time window as (‘1’ – in case of an event reported, ‘-1’ – in case of a normal operation) for transfer learning and semi-supervised learning classifiers. Whereas, unsupervised and supervised learning classifiers, windows are labeled as (‘1’ – in case of an event reported, ‘0’ – in case of a normal operation). When performance measures were applied to assess

the performance of the transfer learning and semi-supervised classifiers, predicted labels '-1' were transformed to '0' to match with ground truth labels.

2.5.5 PMUs Split

Similarly, a set of 38 PMUs was split into two subsets, the source subsets, D_s, and the target subset, D_t. The split between two subsets was made using *RA* feature based on the following procedure. First, a set of 35 events was selected randomly. For each of the 35 events, the *RA* feature was extracted on each PMU. For each of the 35 events, top 3 PMUs with the greatest *RA* were selected. Different subsets of PMUs were iterated until the smallest subset was found that had at least one of top three PMUs in each of the 35 events. This resulted in 12 chosen PMUs that combined have a representative in the top three *RA* in all 35 events. The procedure is outlined in Table 2.2 using a simplified example with 7 PMUs and 4 events.

Table 2.2: Split into two subsets of PMUs for transfer learning based on calculated *Rectangle Area* during events.

	Event 1	**Event 2**	**Event 3**	**Event 4**	**Comment**
Top 1	***RA*$_{PMU1}$=56**	*RA*$_{PMU5}$=32	***RA*$_{PMU2}$=48**	*RA*$_{PMU4}$=17	Only the top 3 PMUs with largest RA for an event are considered as candidates for the Source Subset
Top 2	*RA*$_{PMU3}$=54	***RA*$_{PMU2}$=31**	*RA*$_{PMU7}$=32	*RA*$_{PMU6}$=16	
Top 3	*RA*$_{PMU7}$=44	*RA*$_{PMU4}$=28	*RA*$_{PMU3}$=27	***RA*$_{PMU1}$=12**	
Top 4	*RA*$_{PMU2}$=42	*RA*$_{PMU1}$=27	*RA*$_{PMU1}$=24	*RA*$_{PMU7}$=8	The rest of the PMUs with lower RA are not considered as candidates for the Source Subset
…	…	…	…	…	
Final split with minimum elements in *PMU Source Subset*					
PMU Source Subset, D_s = {PMU1, PMU2} Each event has at least one of these two PMUs in the Top 3 based on the *RA*					
PMU Target Subset, D_t = {PMU3, PMU4, PMU5, PMU6, PMU7}					

Additional 7 PMUs were selected randomly from the remaining set of PMUs, totaling the final 19 PMUs in the *PMU Source Subset*. The remaining 19 PMUs were placed in the *PMU* target subset, D_t. A proportion of labeled time windows from D_s were randomly selected; selected time windows from D_s were leveraged and knowledge was transferred to D_t. Then, related time windows selected for transfer from D_s were combined with D_t in a dataset D^*.

Similarly, for the unsupervised, semi-supervised, and supervised classifiers, D_s was used as the training subset and D_t was used as the test subset. Since the aforementioned classifiers do not transfer related time windows to the target domain, all windows from the set of PMUs in D_s were used for the prediction task.

Features for a certain time window are combined into a feature vector that contains 19 RA features. The process of creating labels y is identical to the process of the *Temporal Split* experiment.

2.6 Experimental Setup

Extensive experiments conducted in our study are described in this section. Using limited proportions of labeled data incorporated into the models we assessed and compared the capabilities of our method to alternative models (unsupervised, semi-supervised, and supervised) to detect events based on a limited proportion of labels, or without any labels used. Experiments conducted included 2%, 5%, 10%, 25%, 40%, 55%, and 70% of available labeled data, corresponding to 20, 51, 103, 259, 415, 570, and 726 available labeled data instances respectively. Available data instances were randomly selected from the source dataset D_s, whereas target dataset D_t was fixed among all experiments. This does

not apply to unsupervised learning algorithms since they do not incorporate any labels during learning. The performance of the classifiers was evaluated using the area under the receiver operating characteristic (AUROC) since this metric is the standard in event detection tasks [35]. Other relevant performance measures including Precision, Recall, F-1 score, and Matthews Correlation Coefficient (MCC) (also known as phi coefficient) were also reported. The formal definitions of these metrics are very common and can be easily found [35, 36, 37].

The different uses of leveraging knowledge from source to target domain are illustrated in Section V-D and Section V-E. A variety of experiments were conducted to address the following comparative questions:

- How do window sizes (time intervals) over which features were computed affect the performance of the algorithms? Different window sizes varying from 2 seconds to 1 minute were experimented to determine the best choice for the event detection task.
- How does the percentage of labeled time windows in the source data affect the performance of the models? Varying percentages of labeled source data ranging from 2% to 70% were experimented to analyze the performance of the models and analyze what percentage of labeled source data is sufficient for the models to detect events.

2.6.1 Hyperparameter Tuning

Hyperparameter tuning using cross-validation is infeasible since labels of time windows in the target domain are not available, and the distributions of the source data D_s

and target data D_t are dissimilar [38]. Instead, the baseline and recommended hyperparameters in comparative studies were used. LocIT has three significant hyperparameters that need to be set. We used a transfer threshold ψ of 0.7, which indicates how closely related time windows to be transferred are and scaling that determines whether to scale the source and target domain before transfer using StandardScaler. In the final classifier, SSKNNO, the three significant hyperparameters were set as contamination of 0.34, k of 1, and strict supervision. The contamination is the threshold of anomaly score, k is the number of nearest neighbors, and supervision indicates whether to use all time windows in the set of nearest neighbors (loose) or use only windows that count the window among their neighbors. Hyperparameters that were set for all classifiers are listed in Table 2.3, categorized by a learning type.

Table 2.3: Selected hyperparameters for the binary classifiers categorized by learning type.

Unsupervised	**kNNO** (weighted=True, k=10, contamination=0.34)
	iNNE (n_members=1000, sample_size=16, contamination=0.25)
Supervised	**MLPClassifier** (alpha=0.3)
	LogisticRegression (C=0.9)
	KNeighborsClassifier (n_neighbors=10, weights='distance')
	NuSVC (kernel='rbf')
Semi-supervised	**SSkNNO** (metric='euclidean', k=1, supervision='strict', weighted=True, contamination=0.34)
	SSDO (metric='euclidean', k=10, contamination=0.39, alpha=0.2)
Transfer Learning	**LocIT** (transfer_threshold=0.7, scaling='none', metric=’euclidean’)

2.7 Experimental Results and Discussion

2.7.1 Distributional Difference between Source and Target Datasets

To demonstrate the applicability of utilizing transfer learning techniques on PMU measurements data for event detection, the three assumptions explained in Section III had to be validated. Transfer learning is typically applied on datasets where traditional machine learning modeling assumptions are violated since the marginal distributions of the source and target subsets are dissimilar (covariate shift assumption), or the conditional distributions are different owing changes in context, in which the meaning of the same behavior might be different in both the source and target domains (concept shift assumption). Therefore, in the initial experiment of our study Kolmogorov-Smirnov (KS) test for comparing the similarity between two continuous distribution functions G and F, was used to check whether the source and target distributions are identical by comparing the underlying distributions $F(x)$ and $G(x)$ of two independent samples [38], where x denotes to the RA features for a certain PMU. The null hypothesis was $F = G$, indicating that the distributions of the source and target are identical.

We applied the KS test metric on the source and target subsets, where the source is a 1-dimensional array containing the RA features collected over the year 2016 for a single PMU, and the target was a 1-dimensional array containing the RA features collected over 2017 for the same PMU. This process was repeated for each PMU. Furthermore, we confirmed the results using the Empirical Distribution Function (EDF) by modeling and sampling the cumulative probabilities for a data sample that does not fit standard probability distribution.

We obtained the p-values from the KS test metric for all PMUs. The maximum p-value was $3.9e^{-15}$, hence, due to a very small p-value (i.e., < 0.01) we can safely reject the null hypothesis, indicating the source and target distributions are different. Figure. 2.2 shows an example for one PMU to illustrate the distributional difference between the source and target subsets. The top and bottom figures show the distribution of the same PMU over two consecutive years.

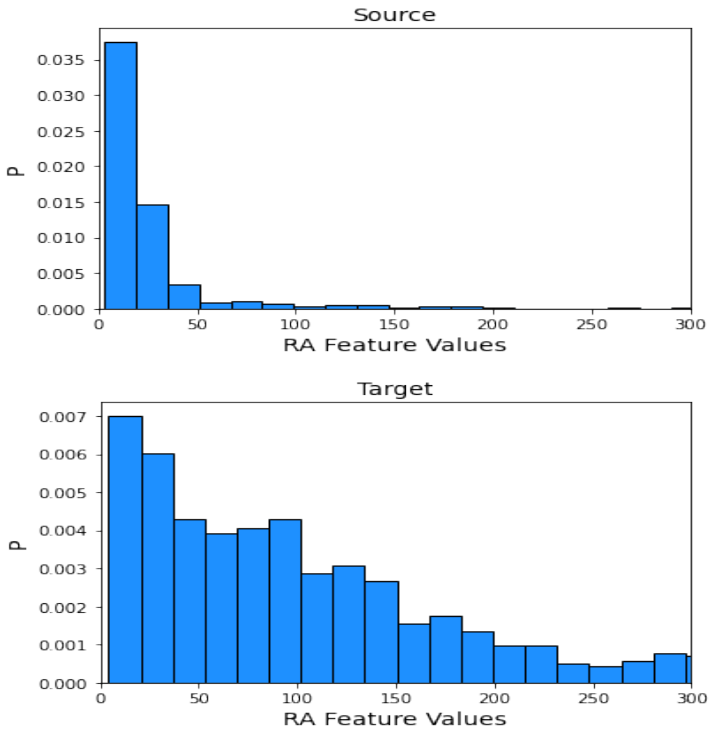

Figure 2.2: An example to illustrate the distributional difference between the source (2016) and target (2017). X-axis is the time window RA values for a certain PMU, and y-axis, *P* denotes to the probability that a certain RA feature will belong to a certain pin. Both top and bottom figures show the distribution of the same PMU over two years, where source contain RA features collected over 2016 and target contain RA features collected over 2017.

Many reasons could lead to a distributional difference of PMU data collected from the future. That might occur since power systems experience randomness of occurrence of events depending on the circumstances, including but not limited to, weather, equipment

failures, wear and tear, and the fact that operating conditions differ every year. Thus, this experiment suggests that transfer learning could be more applicable than supervised learning alternatives that assume the same distribution.

2.7.2 The Effect of Varying Percentages of Labeled Data

In order to study the effect of the amount of labeled source data on the performance of our model versus other models, a variety of percentages of labeled time windows were analyzed. Often, it is non-trivial to acquire labeled data or event logs for event detection tasks since label extraction can be expensive and sometimes impossible to obtain. Thus, this experiment is relevant and provides insights on what percentage of labeled data is sufficient for the models to detect anomalous events. We randomly sampled 2%, 5%, 10%, 25%, 40%, 55%, and 70%, corresponding to 20, 51, 103, 259, 415, 570, and 726 of labeled source data D_s and only considered these labeled time windows when performing a transfer. Experiments were repeated five times and the results were averaged. This experiment was conducted based on *Temporal Split* of the data described in Section V-D. The best performing methods from three alternative learning types (i.e., unsupervised, semi-supervised, fully supervised) were chosen and compared to the transfer learning method.

Figure 2.3 shows that the AUROC improves with more labeled data added to the source subset on three datasets with different window dimensions, listed in Table I. The utilized transfer learning method LocIT, outperforms unsupervised kNNO, semi-supervised SSKNNO, fully supervised MLP, learning algorithms with limited or no labels used in the source data. With only 2% of labeled source data used, corresponding to ~20 characteristic events the transfer learning algorithm performed generally well, while the fully supervised algorithm performed poorly. The gap widened between supervised learning and transfer

learning algorithms as the labeled source data decreased. However, with >60% of labels, supervised learning outperformed transfer learning with a slight increase in AUROC. There were considerable discrepancies between supervised and transfer learning algorithms with <10% of labels used in all experiments conducted. The unsupervised kNNOs curves are straight lines because it is trained without using any labels and it only considers the target data. This was included to visualize and compare with other algorithms. The unsupervised algorithm was trained without any labels, outperformed supervised learning algorithm with <5% of labels in 2-seconds time windows. With 1-minute and 30-seconds time windows, unsupervised outperformed supervised learning with approximately <30% of labels used. Unsupervised learning performed poorly compared to transfer learning and semi-supervised algorithms with a small percentage of labeled data used, since labeled data can assist with correcting the errors made by unsupervised detectors. The semi-supervised algorithm's performance was adjacent to transfer learning's performance with >10% of labeled data. Transfer learning's performance was greater than semi-supervised learning with <10% of labeled data since only related time windows were used to guide with the event detection task. On average transfer learning yields an average increase in AUROC of approximately 13% compared to supervised learning, and 5% compared to unsupervised learning. This provides evidence that the proposed transfer learning approach can help with PMU event detection tasks when labels are not available or are expensive to obtain. Additionally, this shows that supervised learning algorithms rely heavily on labels and are infeasible for detecting events with limited labeled data.

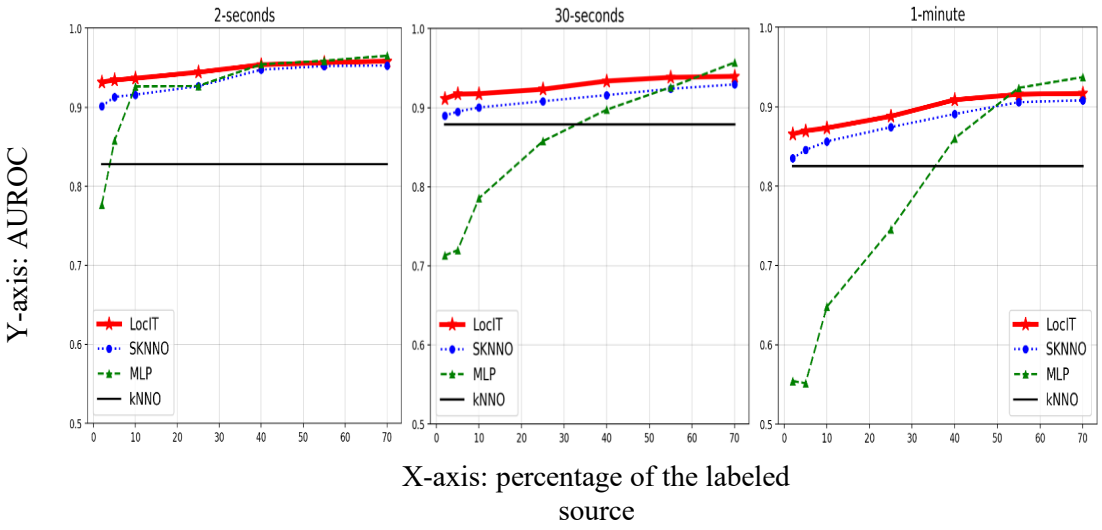

Figure 2.3: Comparing performances of the proposed transfer learning algorithm LocIT based on varying percentages of labeled source data on different window dimensions to three alternative learning types SKNNO, MLP, and kNNO, based on AUROC metric. Performances improve with more labeled data added to the source. Results are consistent and show that LocIT, always performs better with limited labels used. This experiment was conducted based on a temporal split.

2.7.3 Transfer Learning vs. Baseline Anomaly Detectors

Table 2.4 compares the proposed transfer learning algorithm, LocIT to baseline algorithms with different learning types based upon the best performing length of time windows (i.e., 2-seconds) with only 2% of labeled data used to challenge the event detection task based on *Temporal Split* and *PMUs Split* of the data. In both experiments, transfer learning outperformed unsupervised, semi-supervised, and fully supervised algorithms. Results were consistent among all experiments conducted on all datasets. With sufficient amounts of labeled data available, supervised algorithms perform well. However, when limited or no labels are available, unsupervised algorithms, semi-supervised, and transfer learning with semi-supervised, outperform supervised learning algorithms. Additionally, Table IV shows consistency of results where LocIT outperforms other

models with limited labeled data and shows significant improvement over supervised algorithms (MLP, LR, KNN, SVM). Experiments conducted provide evidence that Transfer Learning and Semi-supervised algorithms are more feasible than supervised algorithms for event detection tasks when labels are scarce.

Table 2.4: Performance of various models trained using only 20 labeled events based on *Temporal and PMUs Split*

Experiment	Model	AUROC	Precision	Recall	F1-score	MCC
2-sec. Temporal Split	LocIT	**0.93**	0.93	0.93	0.93	0.86
	SSKNNO	0.90	0.92	0.92	0.92	0.83
	SSDO	0.82	0.84	0.84	0.84	0.67
	MLP	0.77	0.89	0.77	0.79	0.64
	LR	0.68	0.86	0.68	0.69	0.52
	KNN	0.67	0.85	0.67	0.68	0.50
	SVM	0.66	0.85	0.66	0.66	0.48
	kNNO	0.82	0.84	0.84	0.84	0.67
	iNNE	0.81	0.86	0.85	0.84	0.69
2 sec. PMUs Split	LocIT	**0.94**	0.95	0.95	0.95	0.89
	SSKNNO	0.90	0.93	0.92	0.92	0.84
	SSDO	0.80	0.85	0.84	0.84	0.68
	MLP	0.74	0.84	0.81	0.79	0.59
	LR	0.71	0.87	0.71	0.72	0.55
	KNN	0.76	0.85	0.82	0.81	0.61
	SVM	0.72	0.87	0.72	0.74	0.57
	kNNO	0.76	0.83	0.82	0.80	0.62
	iNNE	0.86	0.90	0.89	0.89	0.84
Transfer Learning: LocIT; Semi-supervised: SKNNO, SSDO; Supervised: MLP, LR, KNN, SVM; Unsupervised: kNNO, iNNE						

2.7.4 The Effect of Varying Time Window Sizes

In order to explore how the window size affects the performance of the models, a variety of window sizes were analyzed, including 2-seconds, 5-seconds, 10-seconds, 30-seconds, and 1-minute window sizes. Figure 2.4 shows that the AUROC improves with shorter window sizes among both experiments *Temporal Split* and *PMUs Split*. Applying

transfer learning algorithm on 2-seconds window size yields an approximately 7% increase compared with 1-minute windows based on *Temporal Split* of the data, and 5% increase based on *PMUs Split* of the data. The increase in AUROC is due to the nature of the distance-based classifier. Each time window was manually inspected to make sure that the start of the anomalous event fell within of the selected time window. Anomalous events result in fluctuations (i.e., abnormal behavior) in the signal. As such, when detecting events, distance metrics in shorter time windows highlight the deviation from normal operation more than when longer time windows are used, since the anomalous event corresponds to a shorter timeframe, whereas the rest of the signal corresponds to normal operation. Hence, having a longer time window can dilute the event effect in the window. Fluctuations impact the RA feature values owing to the difference between the minimum and maximum values of positive sequence voltage magnitude and frequency. There was no significant increase in AUROC between the 2-second and 5-second windows. There was a significant improvement in AUROC when detecting events based on 2-second windows compared to 1-minute windows. The increase in AUROC observed when comparing 1-minute labels to 2-second labels can be explained by the smaller fluctuation of normal operation within a shorter time window. Experiments conducted show that shorter time windows result in a higher AUROC. Thus, the size of the time windows was determined based upon the size that exhibited the best performances formed on the results obtained from the conducted experiments.

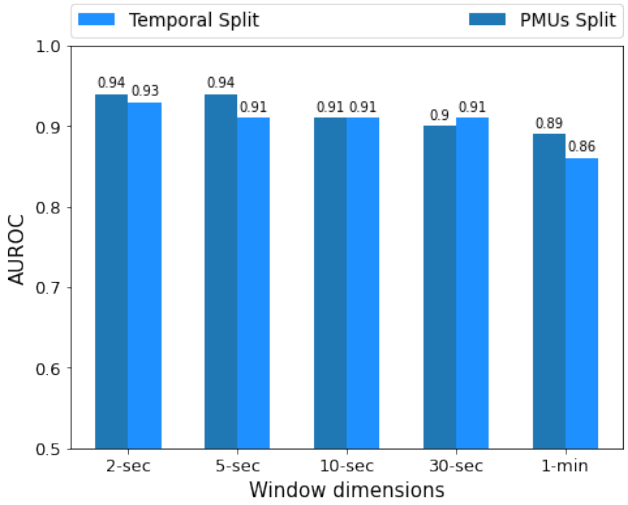

Figure 2.4: LocIT's performance based on AUROC by varying window sizes based on temporal and PMUs data split.

Figure 2.5 demonstrates the fluctuations caused by longer time windows based upon two normal operation instances captured in a 2-second and 1-minute time windows. 1-minute time window shows more fluctuations occurred during the normal operation, and the 2-second time window showed slight fluctuations occurred during the normal operation. Thus, the use of a shorter time window exhibits less fluctuation resulting in a better performance.

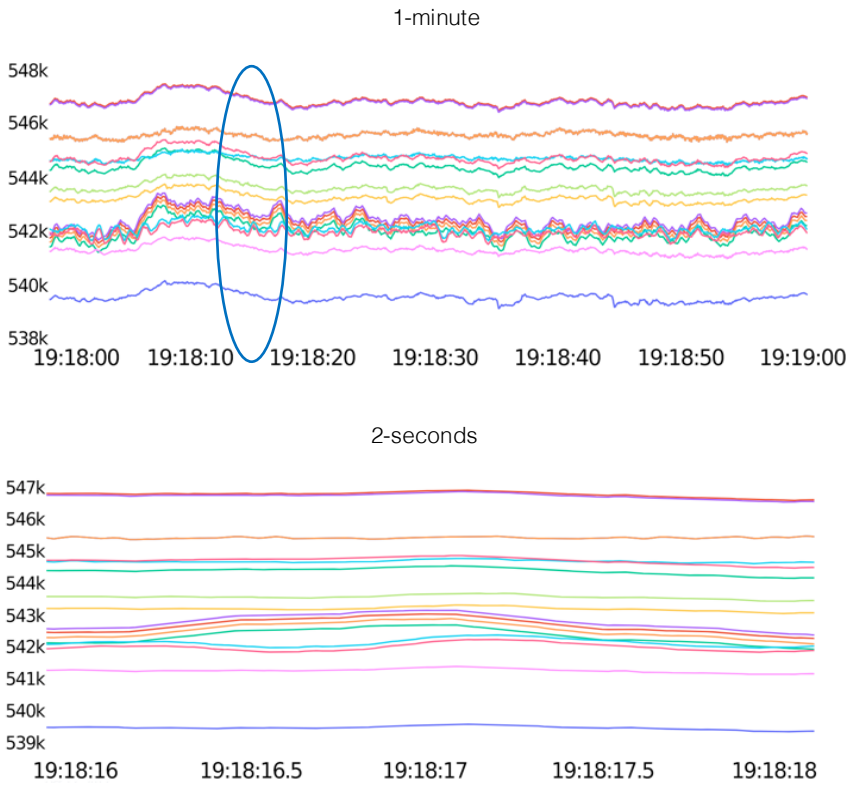

Figure 2.5: The top figure shows 1-minute time window of normal operation, and the bottom figure shows 2-seconds time window of normal operation. Each colored line corresponds to a certain PMU. Shorter time windows produce less fluctuations, resulting in better performance in detecting events.

2.7.5 Leveraging Knowledge on Temporal and PMU Splits

Two experiments were conducted based on different ways of leveraging the data from source to target to test and ensure the model's robustness. Different data splits are introduced in Section V-D and Section V-E. Figure 2.4 shows the performance of transfer learning algorithm, LocIT, based on *Temporal Split* and *PMUs Split* of the data. As can be seen, leveraging limited knowledge temporally, from the year of 2016 and transfer to 2017 results in a slight decrease in AUROC, which can be explained by the randomness of occurrence of events depending on the circumstances that might differ from a year to

another. Leveraging knowledge from a set of PMUs to another set of PMUs shows an increase in AUROC, but the difference is not significant. Splitting the source and target datasets temporally yields a decrease of approximately 1.5% compared with the PMUs split of the data. Thus, this shows that it is feasible to transfer knowledge from historical data and apply it on time windows from the future, and/or time windows from a specific set of PMUs to another set of PMUs.

2.7.6 Misclassified Time Windows

We examined the time windows that were misclassified by the transfer learning approach to event detection to develop a better understanding of the nature of events that led to errors in detecting events. Both false positives (FPs) and false negatives (FNs) were observed. FPs are events that were misclassified as anomalous operation, but in fact, they were normal operations. FNs are the events that were misclassified as normal operation, but in fact, they were anomalous events. A pattern was observed based upon visually inspecting the misclassified events. These events were local events, meaning that they were not observed by most of the PMUs in the interconnect. Moreover, their impact on local PMUs in terms of prominent changes in voltage or in frequency is weak as compared to major events that might precede them. Recall that the input to the algorithm is a vector of RA features from all the PMUs. The difference between the maximum and minimum voltage and frequency in these instances was not as substantial, resulting in a smaller value of RA. Thus, since most PMUs did not observe these changes, false classification (i.e., errors) occurred. Additionally, since the employed semi-supervised detector is distance-based, the weak changes in voltage or in frequency affected the distance metric due to fluctuations in the time window, hence, the detector misclassified these instances.

Furthermore, upon visual inspection of the misclassified events, we observed unrealistic values of frequency in the order of thousands of Hz or sudden drops to zero in the value of voltage and frequency, since no data cleansing was performed prior to the extraction of the RA features. Hence, this led to false classifications as well.

Figure 2.6 provides two 2-second time windows that contain events. The top figure shows an apparent event with a significant drop in voltage and was observed by most PMUs, hence, it was classified correctly as anomalous event. The bottom figure shows a misclassified time window since there was a very minor drop in voltage and did not affect most of the PMUs (i.e., local event).

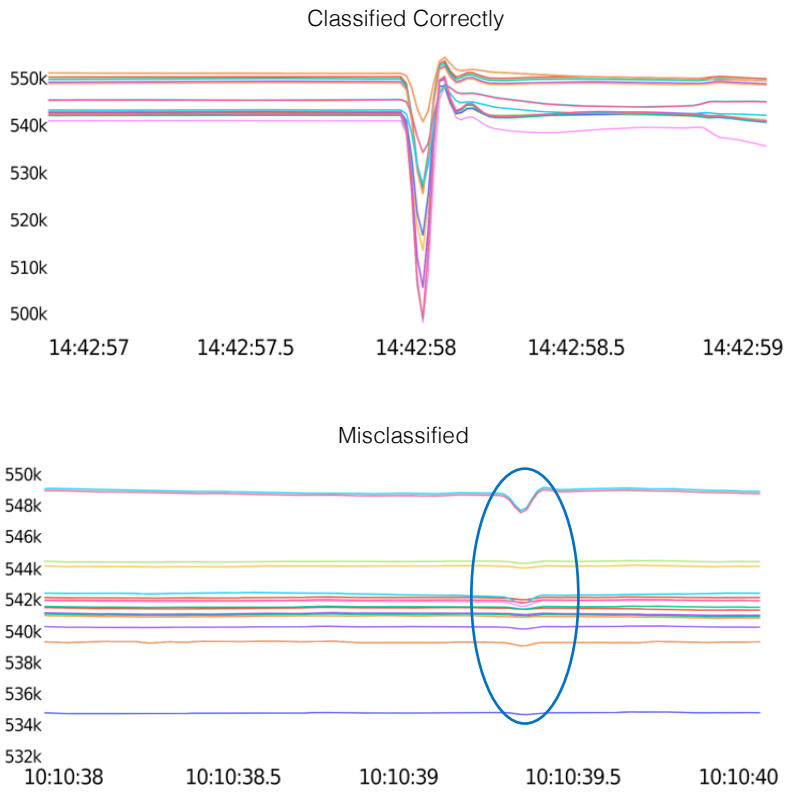

Figure 2.6: Both the top and bottom figures show 2-second time windows that contain events. The top figure shows an obvious event that was observed by most PMUs and was classified correctly as anomalous event. The bottom figure shows a very minor dip in voltage that did not affect most of the PMUs; hence, it was not classified correctly. The bottom time window was classified as normal event.

2.7.7 Statistical Significance Analysis

Statistical analysis was performed to assess the significance and stability of the proposed method's performance. To address how frequently we can expect the proposed algorithm LocIT to obtain the same performance measured based upon the AUROC metric under different conditions, we randomly selected 19 PMUs for the source data and the remaining 19 PMUs were the target data. For consistency with experiments conducted in this study, we leveraged only 20 labeled time windows from the source data to predict the target domain. We repeated the random selection of PMUs 10 times and obtained results from all algorithms used in this study. Then, we employed a t-test with a significance level of 0.1 to obtain confidence intervals with 90% confidence level for the average AUROC for each algorithm individually. Table 2.5 summarizes the average AUROCs and their corresponding two-sided confidence intervals. In general, confidence intervals obtained for the algorithms are very small (< 0.05), hence, it is possible to rely on these algorithms to obtain a similar AUROC with 90% confidence level. LocIT obtained an average AUROC of 0.94 with a confidence interval width of 0.0032, meaning that the average may vary up to ±0.0032 outperforming baselines with high confidence.

Moreover, an additional analysis was performed to assess how statistically more significant the performance of LocIT is, compared to the other algorithms. We calculated the differences between LocIT's AUROC and baseline methods and employed a t-test. The p-values obtained were very small (< 0.05). The p-value obtained by comparing LocIT to the second best-performing method SSKNNO was $1.4e^{-8}$, indicating that LocIT's performance is significantly better than other baselines.

Table 2.5: Summarizes the average AUROC and their corresponding two-sided confidence interval, calculated at 90% confidence level.

Learning Type	Model	Average AUROC and Confidence Interval
Transfer Learning	LocIT	**0.94 ± 0.0032**
Semi-supervised	SSKNNO	0.90 ± 0.0036
	SSDO	0.81 ± 0.0078
Supervised	MLP	0.74 ± 0.0058
	LR	0.72 ± 0.0074
	KNN	0.76 ± 0.0042
	SVM	0.72 ± 0.0199
Unsupervised	kNNO	0.75 ± 0.0037
	iNNE	0.85 ± 0.0113

2.8 Extension and Enhancement to Perform Transfer Learning on PMU Data from a Power System to Detect Events in Another System

The experiments conducted show that the proposed transfer learning approach is capable of detecting events by leveraging minimal labeled time windows from a related task within PMU data of the Western Interconnection of the U.S.A. Upon promising results reported in the previous study, the proposed technique could be extended to leverage labeled time windows from the Western Interconnection of the U.S.A. and transfer learning to the Eastern Interconnection of the U.S.A. for event classification task. The challenge of this task revolves around different number of PMUs at two interconnections, resulting in different dimensions of feature vectors. Moreover, the distance metric used in the semi-supervised detector (SSKNNO) might be less effective for the Eastern Interconnection where a much larger number of PMUs were observed. Hence, implementation of an

appropriate distance measure that is suitable for high dimensional feature vectors might be required to enhance the event detection task.

Therefore, we extended and enhanced the transfer learning method to leverage related labeled instances from one power system to detect events from another. Set of experiments conducted show that events in one power system can be accurately detected by reusing a small number of carefully selected labeled PMU data from another without the need for additional labeling. We demonstrate this approach with a use of detecting events from historical PMU data recorded in the Eastern Interconnection in the USA by using similar labeled PMU data from the Western Interconnection. This technique may be propagated to other situations where some of the events' data from one power system may be applied to enhance learning in another. This enhancement addresses the nonunified dimensions of feature vectors when leveraging data from a power system to detect events from another, and enhanced the semi-supervised detector by using a distance metric that is more applicable for the dimensions of PMU data. Our approach demonstrates superior performance over various state-of-the-art machine learning algorithms (i.e, unsupervised, semi-supervised, and supervised).

2.8.1 Related Work

In [2], TL was applied to detect events using PMU measurements by transferring relevant labeled data from a power system collected in one year (2016) to detect events from future instances (2017) in the same power system. In [39] TL technique in conjunction with deep learning model was utilized to enhance the detection of events in one power system using a model pre-trained on another. The use case of using PMU recorded data from the Western and Eastern Interconnections (WI and EI) in the US demonstrates that

the use of TL enhances the performance by leveraging labeled data from both WI and EI to enhance the detection on WI. This model transfers parameters of the pre-trained model, trained on EI to be used as the initial parameters of the model trained on the data of the WI. As illustrated in Sec. IV, the quality of data of the EI is poor compared to data from the WI, so detecting events from WI based on EI only might be insufficient. There are some limitations of the study reported in [39]: a) its proposed model does not detect events from one power system based on another without utilizing labeled data from both power systems, b) it utilizes a fully supervised learning estimator and considers only line, generator, oscillation events, and normal/healthy signals.

2.8.2 New Event Detection Approach

To address the mentioned gaps, our paper extends and enhances studies reported in [2, 39] by exploring the benefits of knowledge transfer between two independent power systems, such as the WI and EI in the USA using a transfer function combined with a semi-supervised detector to identify events based on minimal labeled data of the source task only, and it downgrades to unsupervised mode if no related labeled data instances were available in the source power system.

To address the mentioned issues, we propose the following two methods based on TL techniques:

1) Spatial transfer, sLocITR (spatial localized instance transfer reduced), which leverages labeled data from one power system to detect events in another system. Our approach does not require target labels, since it relies only on related instances from the source power system to detect events from the target power system, while

the study reported in [39] requires target labels since it leverages labeled data from both power systems (source and target) to detect events from the target power system.

2) Spatiotemporal transfer, stLocITR, based on leveraging labeled data from one power system integrated with a small number of labeled data from another power system to detect future events. Table 2.6 summarizes the major differences between the proposed approach and studies reported in [2, 39].

Table 2.6: Comparing the proposed method to transfer learning alternatives.

Study	Source	Target	Transfers	Detector	Target Labels
[2]	WI_{past}	WI_{future}	Temporally Related data	Semi-supervised	Not Required
[39]	EI & WI	WI	Parameters	Supervised	Required
sLocITR	WI	EI	Spatially Related data	Semi-supervised	Not Required
stLocITR	WI & EI_{past}	EI_{future}	Spatio-Temporally Related data	Semi-supervised	Not Required

2.8.3 Compression and Unification of Data Dimension

To transfer labeled instances from one power system to another, we project time windows (*TWs)* from the source and target datasets of the two power systems with different numbers of PMUs to latent spaces of unified dimensions while preserving the properties of the original data. This is achieved by an Autoencoder, i.e., an unsupervised Neural Network (NN) for dimensionality reduction. Autoencoders utilize multiple neural computing layers to learn non-linear transformations of data to a latent space [40]. Other dimensionality reduction techniques such as Principal Component Analysis (PCA) were

also considered but failed to learn a representation that preserves the properties of the original data since such techniques are limited to linear transformations only [12]. The feature vectors (*TWs*) from both datasets were extended to 200 dimensions by padding with zeros, thus standardizing the number of dimensions in the two datasets. In the use case with the data from WI and EI, two fully connected layers with batch normalization were used to learn how to unify the 35-dimensional feature vectors from both WI and EI datasets. To enhance the performance of the ML models, the unified data were scaled to a standard range using Standard Scaler [2], defined as $z = \frac{x-\mu}{\sigma}$.

2.8.4 Data Processing

PMU Data. We utilize historical field measurements collected over two years, 2016-2017 from 38 PMUs placed in the WI, and from 178 PMUs placed in the EI in the U.S. electric power system. The measurements from EI are collected at 30 frames per second (fps), while measurements from WI are collected at 30 fps or 60 fps. Locations of PMUs and the system topology are not provided to us. Some outliers, data duplicates and missing data are observed in both datasets but do not affect our method significantly [41]. Non-uniform number of PMUs and data quality issues make this event detection task complex. WI dataset contained higher quality measurements than the EI dataset, since EI contains missing data ranging from ~1% to ~70%, whereas missing data of WI ranges from ~1% to ~30%. Thus, we utilize labeled data from WI to detect events from EI, without using any labeled data from EI.

Event Log. Similarly, both WI and EI datasets contain phasor measurements associated with line outages, transformer outages, and fundamental frequency deviations that are

labeled in the event log. Visual inspection of these events revealed that some events evolve from one type to another, hence, they were considered "complex" events. Complex events include events labeled generator, capacitor, bus, and oscillation. The provided event log most likely was obtained from the SCADA data, and therefore it contains temporally imprecise event labels (start time with a precision of 1-minute). In addition, due to the sparsity of PMU locations in the network, log events did not necessarily occur in the vicinity of the PMUs used in this study. To improve the temporal precision of the log events, visual inspection was performed by the domain expert on our team. Then, we used a more precise start time of the events confirmed through visual inspection. The study reported in [2] experimented various dimensions of *TWs;* 2-second *TWs* resulted in performant classification results; hence, the dimension of 2-seconds was used. Table 2.7 presents the number of labels used for each proposed method.

Feature Extraction for EI and WI data was performed as described in Section 2.5.3.

Table 2.7: Number of labels per category from both WI and EI datasets.

Method	# Event Labels from WI	# Normal Labels from WI	# Event Labels from EI	# Normal Labels from EI
sLocITR	1038	1846	0	0
stLocITR	1038	1846	849	762

2.8.5 Experimental Setup

We propose two TL methods based on different splits of the source and target datasets.

1) *Spatial transfer*, sLocITR, where labeled *TWs* were selected from D_s which consisted of *TWs* from WI and were used to detect events in D_t, which consisted of

unlabeled *TWs* from EI. In this experiment, D_s contained the entire data of the WI, while D_t contained the entire data of the EI.

2) *Spatiotemporal,* stLocITR, where $D_s = WI \cup EI_{2016}$; $D_t = \mathrm{EI}_{2017}$; where *WI* denotes the entire *TWs* of the WI, EI_{2016} denotes the *TWs* of the EI collected from 2016, used to detect events in EI_{2017} which denotes the *TWs* of the EI from 2017.

We answer the following empirical questions:

1) How does the proposed TL method perform compared to alternative baselines?
2) How does the number of labeled source data selected from D_s affect the classification accuracy for events in the target domain D_t?

The results validate our hypothesis and illustrate the benefits of employing TL techniques in conjunction with a semi-supervised detector to leverage knowledge and detect events based on minimal labeled data. To address question 2, we selected the top p related instances excluding redundant/similar instances to experiment how the proportion of labeled data affects the performance; where $p \in \{20, 51, 103, 259, 415, 570, 726\}$ corresponding to 1% to 25% of labeled source data instances.

The performance of the TL algorithm was evaluated by comparing it to common conventional ML algorithms of varying learning types described in Sec. III (i.e., unsupervised, supervised, and semi-supervised). The following metrics were used to evaluate the algorithms: The area under the receiver operating characteristic (AUROC), Precision, Recall, and F-1 score [35].

2.8.6 Results and Discussion

WI and EI Distribution Comparison. To validate the applicability of the TL on PMU data, we utilized Kolmogorov-Smirnov (KS) metric to test if the cumulative distribution functions of the source WI and target EI datasets are similar. KS metric was applied to compare two independent samples on the source and target system, where the source is represented as a 1-dimensional array that contains features from the WI and the target is a 1-dimensional array that contains features from the EI. We obtained p-values by iteratively computing similarities between two independent samples. The maximum p-value was $2.7e^{-13}$, thus, since the obtained p-value is very small, we can safely reject the null hypothesis, implying distributions of WI and EI are different.

2.8.7 Transfer Learning versus Baseline Event Detection

Table 2.8 presents and compares the performance of the proposed TL methods stLocITR and sLocITR to alternative baselines of various learning types. Consistent results demonstrate the effectiveness of the proposed methods and show that both methods outperformed fully supervised, semi-supervised, and unsupervised algorithms. The sLocITR method selected and transferred 570 (out of 2,884) related data instances (543 abnormal events + 27 normal) excluding redundant instances from WI to detect events from EI. stLocITR transferred additional 362 (out of 1,611) temporally disjoint related cases from EI, resulting in increased AUROC by 11% when compared to the best performing supervised and unsupervised learning algorithms, and a 5% improvement when compared to the best performing semi-supervised learning algorithms. sLocITR increases the AUROC by 12% when compared with the best supervised learning algorithm, 10% improvement when

compared with the best unsupervised learning algorithm, and 3% improvement when compared with the best semi-supervised learning algorithm. Unsupervised learning algorithm, kNNO outperformed supervised variant using the Spatial split, indicating that the source and target label sets differ significantly. In other words, there are many input-output relationships in the target domain that do not have similar counterparts in the source. However, the underlying anomaly patterns remain similar. Unsupervised learning is based on detecting anomaly patterns only from the input signals, whereas supervised algorithms attempt to learn the relationship between the input signals and the output labels, which might be misinforming for some cases due to the distributional difference (label sets) of both interconnections.

Table 2.8: Comparative analysis of the utilized transfer learning methods to various baselines using the selected labeled *tws* from d_s.

Method	Learning Type	Model	AUC	Precision	Recall	F1
Spatio-temporal	Transfer Learning	stLocITR	**0.90**	0.90	0.90	0.90
	Semi-supervised	SSKNNO	0.85	0.86	0.86	0.86
		SSDO	0.84	0.86	0.85	0.85
	Supervised	RF	0.79	0.79	0.79	0.79
		KNN	0.79	0.80	0.78	0.79
		MLP	0.74	0.82	0.73	0.77
		SVM	0.72	0.81	0.70	0.75
	Unsupervised	kNNO	0.79	0.80	0.79	0.79
		iNNE	0.74	0.75	0.73	0.74
Spatial	Transfer Learning	sLocITR	**0.87**	0.87	0.87	0.87
	Semi-supervised	SSKNNO	0.84	0.84	0.84	0.84
		SSDO	0.83	0.85	0.84	0.84
	Supervised	RF	0.75	0.77	0.74	0.75
		KNN	0.72	0.75	0.71	0.73
		MLP	0.68	0.77	0.66	0.71
		SVM	0.65	0.77	0.63	0.69
	Unsupervised	kNNO	0.77	0.79	0.76	0.77
		iNNE	0.74	0.76	0.73	0.74

Experiments provide evidence that TL-based methods are more accurate than unsupervised, supervised, and semi-supervised alternatives for detecting events from one power system based on labeled data of another.

2.8.8 The Effect of Using Various Quantities of Labeled Data

Often, obtaining event logs or labeled data for event detection tasks is non-trivial or costly. Thus, we studied the effect of using various amounts of labeled source data to assess what number of labeled data is adequate to detect events from the EI of the U.S.A. based on minimal labeled data from the WI of the U.S.A. (Spatial Split). We selected from D_s 20, 51, 103, 259, 415, 570, and 726 events to detect events from the target data D_t. We repeated the experiments 10 times and reported AUROCs, and their corresponding two-sided confidence interval calculated at 95% confidence level, presented in the shaded area of Figure 2.7 We selected the best methods from various learning types (i.e., fully supervised, semi-supervised, and unsupervised) and compared them with the proposed TL method stLocITR.

Figure 2.7 shows that the TL method outperforms supervised learning on a large benchmark since there is a distributional difference between the D_s and D_t. Results show that the TL method outperforms baselines with varying quantities of labeled data incorporated. The straight line of the unsupervised learning algorithm kNNO with no labels incorporated is included for comparison. When sufficient labeled data are incorporated, semi-supervised SSKNNO outperforms unsupervised learning. The increase in labeled source data is not found to increase the performance of the supervised algorithm, since the source and target label sets differ greatly. This study demonstrates that transferring 570 labeled data instances from the WI are sufficient to detect events from the 3,085 instances

of the EI PMU data. We randomly select a proportion of labeled data from D_s to train supervised and semi-supervised learning algorithms, whereas the TL algorithm uses the most relevant instances from D_s. When comparing sLocITR with a supervised learning algorithm, Figure 2.7 shows that selecting the top relevant instances results in not only better performance, but a more stable model since sLocITR has a significantly lower two-sided confidence interval than RF. Table 2.9 illustrates event types when transferring the top selected 100, 300, and 500 instances.

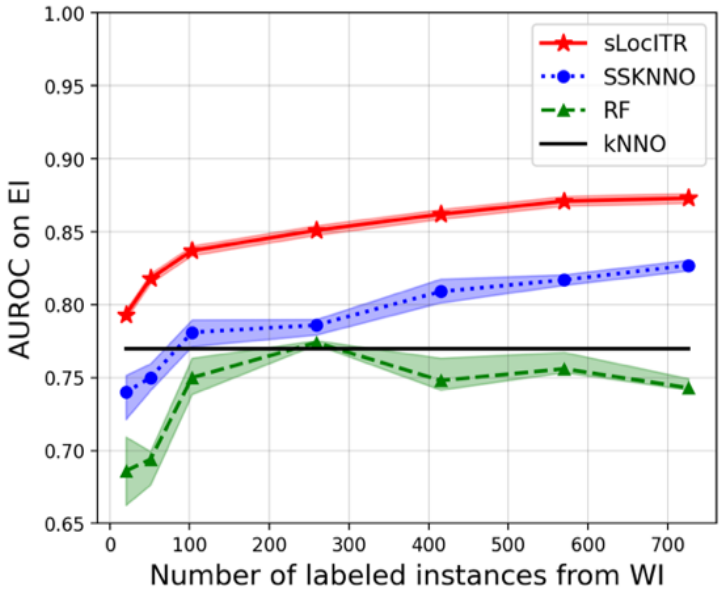

Figure 2.7: Comparing the performance of the proposed method sLocITR to baselines based on varying number of labeled source data evaluated using AUROC and their corresponding two-sided confidence interval calculated at 95% confidence level.

This experiment shows that supervised learning algorithms are infeasible when leveraging knowledge from one interconnection to another due to covariate and concept shift assumptions and when labels are scarce and difficult to obtain.

Table 2.9: Events transferred per category among Top 100, 300, and 500.

# Labeled Events	Line	Frequency	Transformer	Complex	Normal
100	68	15	3	6	8
300	165	71	3	45	16
500	269	103	3	103	22

2.8.9 Misclassified Events

To further comprehend the errors made by the TL method, a domain expert visually inspected the misclassified *TWs.* The most common occurrence of these *TWs* is the presence of low-frequency *oscillations* that the algorithm was unreliable in detecting as only 0.3% of all events in WI were labeled as oscillations even though these events are more common. Low-frequency oscillation events are difficult to capture because their impact is most obvious after performing modal analysis.

2.9 Conclusion

Since obtaining extensively labeled data can be labor intensive, and requires domain knowledge, it may be too costly, especially, when working with big datasets. The results of our study show several benefits of the transfer learning approach utilized for event detection tasks:

- It yields an average increase in AUROC of approximately 13% compared to the best performing supervised learning algorithm, and 5% compared to the best performing unsupervised learning algorithm based on *Temporal Split* experiment. The significant accuracy improvements were evident when relying on only 2% of

labeled data corresponding to 20 characteristic events. Further, it yields an 8% increase in AUROC compared with alternative state-of-the-art algorithms based on leveraging data from WI to detect events from EI, sLocITR.

- The performance is less affected by the decrease in the number of available labels, and algorithm provides high performance even with only 20 representative labeled events used. In comparison, the supervised learning algorithms are infeasible for event detection in this domain when labels are very limited.
- The proposed approach is robust to temporal and locational options for splitting the PMU data. Consequently, it is feasible to leverage and transfer knowledge from historical PMU data to improve learning on future unlabeled instances, and to transfer selected labeled events from a specific set of PMUs to another set of PMUs. The reported results provide evidence that identifying a variety of event types, including line faults, transformer outage, and frequency events by a model that can be deployed to detect events in future PMU data while avoiding challenges faced by online learning. Furthermore, reported results show that the proposed transfer learning method is more applicable than alternative baselines when reusing labeled instances from one power system to detect events from another.

2.10 Disclaimer

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information,

apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

CHAPTER 3

DEEP LEARNING VS TRADITIONAL MODELS FOR PREDICTING HOSPITAL READMISSION AMONG PATIENTS WITH DIABETES

3.1 Introduction

Hospital readmission is an undesirable, costly outcome for both patients and hospitals [42]. Patients with diabetes are at higher risk of readmission within 30 days of hospital discharge (30-day readmission) than patients without diabetes [43-45]. Of the nearly 9 million discharges of diabetes patients annually in the US, [46] almost 2 million are 30-day readmissions, corresponding to at least $20 billion in hospital costs [47-48]. Identifying higher risk patients with diabetes would enable the targeting of interventions to those at greatest need, optimizing the cost-benefit ratio.

We previously published on the development and validation of the Diabetes Early Readmission Risk Indicator (DERRI^{TM}), a logistic regression (LR) model that predicts the risk of all-cause 30-day readmission among patients with diabetes [49]. The DERRI^{TM} was designed for use at the point of care based on user input of 10 factors. In split-sample internal validation, performance was modest (Area Under the Receiver Operating Characteristic Curve, AUROC 0.69). In external validation studies, the DERRI^{TM} AUROC was 0.63 and 0.80 [50,51]. In addition to variable predictive performance, application of

the DERRI™ requires manual data collection and entry, which are major barriers to its use in clinical practice.

In other published work, we showed that adding variables to the DERRI™ substantially improves predictive accuracy to an AUROC of 0.82 [52]. This expanded model (DERRIplus), however, is not feasible for use at the point of care and included employment status, which is not routinely documented in Electronic Health Records (EHRs). Therefore, this model cannot be directly translated to an automated, EHR-integrated tool. There is an unmet need for a readmission risk prediction tool for patients with diabetes that is both accurate and easy to use.

Over the past few years, multiple machine learning (ML) models for predicting 30-day readmission risk of diabetes patients have been published. Several traditional ML modeling approaches have been explored, including random forest (RF), k-nearest nearest neighbor, naïve Bayes, support vector machine (SVM), AdaBoost, and multilayer perceptron (MLP), with a wide range of performance (AUROC 0.53-0.99, accuracy 0.54-0.99) [52-63]. Deep learning (DL) models have also been developed for predicting readmission risk of diabetes patients, also with variable performance (AUROC 0.61-0.97, accuracy 0.69-0.95), none of which exceeded that of the best traditional ML models [64-68]. Two of these studies demonstrated a clear advantage of DL approaches over traditional ML models [64,65], and two studies found marginal benefit with DL approaches [66,68]. Comparisons of model performance across all these studies, however, is limited by the lack of standardized reporting of performance characteristics and variable approaches to testing. Therefore, it remains unclear if DL models outperform traditional ML models at predicting readmission risk for patients with diabetes.

Interestingly, all these prior models were developed on the same dataset [69], except for the DERRI™ and DERRIplus. This publicly available dataset contains hospital encounters with a diagnosis of diabetes and a length of stay between 1 and 14 days at one of 130 US hospitals between 1999 and 2008. Only 3 International Classification of Diseases, Ninth Revision (ICD-9) diagnostic codes per encounter, and only 2 laboratory values (blood glucose and HbA1c) were recorded. Lastly, there is no distinction made between planned and unplanned readmissions. Thus, even the best of these models may not perform as well in patients today. More current, generalizable models are needed.

Therefore, to address these gaps, the aims of the current study were as follows: 1) To develop DL models for the prediction of unplanned, all-cause 30-day readmission, 2) To compare performance of the DL models to traditional ML models, 3) To explore model performance across a range of prior EHR encounters from 1 to 100 being included in model development, and 4) To compare a DL model developed using a subset of laboratory tests selected by domain knowledge with a DL model developed using all available laboratory tests. All models were developed and tested in a dataset of 2,836,569 encounters of 36,641 patients with diabetes using demographics, vital signs, diagnostic and procedure codes, medications, laboratory tests, and administrative data as defined by the National Patient-Centered Clinical Research Network (PCORnet) Common Data Model (CDM) [70]. This study esablishes the foundational framework and baseline for the subsequent chapters.

3.2 Materials and Methods

3.2.1 Definition of Patient Cohort

Inclusion criteria were patients with at least one discharge from any of the three Temple University Health System hospitals in Philadelphia, PA, between July 1st, 2010, and December 31st, 2020, and diabetes defined by at least one of the following: a diagnosis of diabetes (ICD-9: 249.xx or 250.xx or ICD-10: E08.xxx through E13.xxx); a Hemoglobin A1c (HbA1c) level ≥6.5%, or an order for a diabetes specific medication. Encounters were excluded for patient age <18 years, discharge by transfer to another hospital, inpatient death, a diagnosis of gestational diabetes (ICD-9: 648.0x or ICD-10: O24.4x), a diagnosis of prediabetes (ICD-9: 790.29 or ICD-10: R73.03), or pregnancy (positive beta human chorionic gonadotropin laboratory test within 90 days before or after the encounter).

Patients were sorted into one of 2 groups by readmission status: those who had at least one 30-day readmission and those who did not. Among the patients who had a readmission, one admission-readmission pair was randomly selected for analysis. Among the patients who did not have a readmission, one admission was selected randomly for analysis.

3.2.2 Definition of Variables and Data Processing

Tables were extracted from the CDM for each of the following domains: encounters, demographics, diagnoses, laboratory tests, medication orders, procedures, and vital signs. Because features of a given encounter existed in multiple tables, tables were merged by a unique identifier. Merging extracted tables resulted in a sample containing all records for a given encounter. This resulted in substantial missingness. Thus, missingness was used as

a separate feature. For continuous features, missing data were replaced with 0, while categorical features were replaced with a unique category.

A total of 23 features were used as input to the models: 14 were extracted from the CDM and 9 were aggregated. Extracted features were: 1) Encounter type (Inpatient, Emergency Department, Observation Stay, Ambulatory Visit, Other Ambulatory Visit, Telehealth and Other; 2) Discharge Status (Assisted Living Facility, Against Medical Advice, Expired, Home Health, Home/Self Care, Hospice, Nursing Home, Rehabilitation Facility, Skilled Nursing Facility; 3) Sex; 4) Hispanic; 5) Race (American Indian/Alaska Native, Asian, Black, Pacific Islander, White, other/no information); 6) Tobacco (Current user, never user, former user, passive exposure, other/no information); 7) age; 8) Diagnosis Clinical Classification System (CCS) codes [70]; 9) Procedure CCS codes [70]; 10) Laboratory results; 11) Medication orders within 1 year before each encounter; 12) Diastolic blood pressure; 13) Systolic blood pressure; and 14) Body mass index (BMI). Aggregated features were: 1) Elixhauser conditions: a binary feature indicating the presence or absence of each condition [71]; 2) Duration of admission (length of stay in days); 3) number of procedure codes before conversion to CCS code; 4) number of diagnosis codes before conversion to CCS code; 5) number of days since the prior encounter regardless of encounter type; 6) number of days since the prior inpatient, observation or emergency department encounter; 7) number of days since the prior encounter of other (non-hospital) encounter types; 8) number of inpatient, observation and emergency department encounters before the current encounter; and 9) number of other (non-hospital) encounters before the current encounter. ICD-9 codes were converted to ICD-10 codes to unify the code format. ICD-10 codes and procedure codes were converted

to CCS codes. Based on domain knowledge, medications relevant to diabetes were categorized as follows: diabetes medications by class, cholesterol, corticosteroids, renin-angiotensin system (RAAS) blood pressure agents, and non-RAS blood pressure agents. Other medications were ignored. Features found not to be reliable, mostly missing, or correlated were removed. Outliers in features such as dates, results, height, weight, BMI, and blood pressure (systolic and diastolic) were removed by observing the data distributions, percentiles, and domain knowledge. Missing values were treated as another category that indicates that a parameter was not collected in relation to the encounter. The primary outcome for model prediction ($\boldsymbol{y}$) was unplanned, all-cause inpatient readmission within 30 days of an inpatient encounter discharge as defined by the Centers for Medicare & Medicaid Services (CMS) [72]. Based on the CMS definition, only the first readmission within 30 days was analyzed.

To prepare the data for machine learning models the following data preprocessing techniques were performed. Categorical features were one hot encoded; continuous and discrete features were normalized using min-max normalization techniques [73], defined as:

$$\mathrm{x}' = \frac{x - min(x)}{max(x) - min(x)} \tag{3.1}$$

There were different numbers of recordings in each encounter for each of the following features. Thus, the following statistical values were computed instead. For diastolic and systolic blood pressures, we calculated minimum, maximum, and mean values. For BMI, minimum, maximum, mean, and coefficient of variance were used. These statistical values where normalized and used as features. Moreover, the number of features differed at

encounters due to the different number of laboratory tests, diagnoses, and procedures because an encounter could have multiple diagnosis and/or procedure codes, or none. To remedy this and unify the dimensionality of feature vectors, the following data representation techniques were used to enhance the learning of the models. For diagnosis and procedure codes, we used the representation of one-hot encodings, where each value was set to 0 or 1, indicating whether a diagnosis/procedure code existed or not for each encounter. We modified this data representation technique slightly for laboratory tests because each test had an associated result. Hence, we replaced 1, which indicated a code exists, with the laboratory result. Laboratory results were normalized using Equation 1. Because results were of different units and measures, when normalizing laboratory results, we considered the minimum and maximum for each laboratory code separately. This technique created a high dimensional sparse array due to the many unique codes. Then, we utilized Singular Value Decomposition (SVD) algorithm to learn an embedding and reduced dimensionality. SVD was used since it does not assume a square matrix as an input and better for sparse data [74]. Laboratory tests were reduced to 50 components, procedure codes were reduced to 45 components, and diagnosis codes were reduced to 25 components. Different numbers of components were explored, and the sum of variance ratio was observed to determine the optimal number of components to reduce dimensionality. All features were concatenated in a feature vector for each encounter. SVD was applied on each encounter separately to reduce and unify dimensions; dimension of encounters was reduced to 50 features per encounter. Then, we concatenated all encounters for a given patient in a feature vector ordered sequentially by admission date. The class

distribution was 27,511 patients without readmission (negative class) and 9,130 patients who were readmitted (positive class).

3.2.3 Experimental Approaches

We conducted extensive experiments using the EHR data to address the following objectives:

- Predict whether patients with diabetes will be readmitted within 30 days.
- Compare the performance of the utilized DL methods with several traditional models.
- Analyze how many prior encounters (i.e., historical data) within 2 years is optimal to predict readmission.
- Evaluate the effects of incorporating all laboratory tests in the data versus learning from a subset of tests chosen by a domain expert.

In this study, DL models take as an input a 3-dimensional tensor p x e x f to represent f features for each of e encounters for p patients. In contrast, in traditional models, data is typically represented as a 2-dimensional matrix, with all features of all encounters corresponding to a single patient concatenated in a long feature vector. The dimensionality of each encounter was reduced and unified to 50 features, hence, in a deep model f is of size 50. In a traditional model feature vector consists of all encounters and therefore is of size e x 50. Patients have different numbers of encounters resulting in a nonuniform dimensions; hence, feature vectors were padded with 0s to achieve a unified form. Data representation used as input for DL and traditional models is illustrated at the left and right panels of Figure 3.1, respectively.

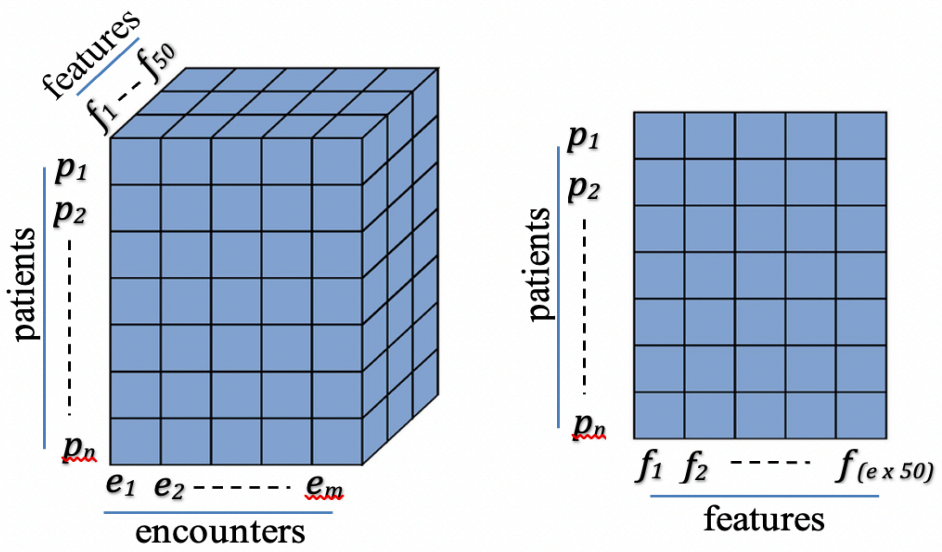

Figure 3.1: Representation of data input to the deep learning models (left), and traditional models (right).

To model heterogenous sequential data, we developed 2 variants of DL models and compared both versus several traditional models used as baselines. DL models used in our study were: 1) 1-way Long Short-Term Memory (LSTM) networks, which are a variant of Recurrent Neural Network (RNN) that is capable of learning order dependence in sequential data [73]; and 2) Bidirectional Gated Recurrent Unit (GRU), which is another variant of RNN. Traditional models used as baselines were: 1) Random Forest (RF), an ensemble method for classification and regression; during training, it constructs multiple decision trees [71]; RF frequently achieves the state-of-the-art performance in existing literature on predictions using medical data. 2) Multi-layer Perceptron (MLP), a simple neural network model that does not account for temporal information. MLP consists of multiple layers of perceptron and performs backpropagation learning and utilizes a non-linear activation function [72]. 3) Logistic Regression (LR), an interpretable model used

frequently in existing literature of readmission predictions and applied on medical data; and 4) AdaBoost, which is less prone to overfitting as its input parameters are not jointly optimized. The DL models were implemented using “Keras” Python libraries, a high-level API of “TensorFlow”. “Scikit-learn” library was utilized to implement traditional models in Python.

The architecture of the proposed model, LSTM, comprises 128 neurons, a sequential layer, a reshape layer that was used to reshape the input to 3-dimensional tensor, and a masking layer with a mask value of 0 used to skip the timesteps for which the data were missing. Since padding with 0s was performed to unify dimensions, the masking layer was utilized to avoid any computation with the missing values in all layers following the masking layer, hence, missing values were not accounted for during learning. Additionally, a dropout was added between hidden and output layers. Utilizing this technique to randomly select a given percentage to drop, which is a common regularization technique that assist the model learn general pattern in data.

RNN is a variant of neural networks, which consist of hidden neurons that are capable of analyzing temporal EHR data [73]. RNN comprises of the same structure as the basic neural network, but neurons in the same layer are connected, allowing a neuron to learn from the same neighboring layers, in addition to learning from outputs of the previous layers and the input data. Thus, RNN neurons include two sources of inputs, the present and the recent past. The process of learning is defined as:

$$\boldsymbol{b^t = \mathbf{ReLU}\,(b + Wh^{t-1} + Ux^t)} \tag{3.2}$$

$$\boldsymbol{\bar{y} = \mathbf{sigmoid}\left(b + \sum_t Vb^t\right).} \tag{3.3}$$

To compute value b^t of a hidden neuron, t, a non-linear transformation function, ReLU, is applied to weighted W value of its left hidden neuron b^{t-1} and the weighted U value of its input x^t. Predictions are computed using a sigmoid function of weighted V sum of all hidden neurons with added bias b. The drawback of RNN is that it suffers from the vanishing gradient problem, meaning that weights remain unchanged making it difficult for the model to convergence, hence, the model struggles to learn. To solve this, an LSTM layer was introduced in which sigmoid neurons of RNN are replaced with more complex short-term memory structure. LSTM shares the same weights across layers, which reduces the numbers of parameters that the network compute. The GRU is an alternate solution for a vanishing gradient problem. It substitutes the simple neuron with a gated unit, which has fewer parameters than the LSTM neurons, because it lacks an output gate [74].

In this study, extensive experiments were conducted to determine how many prior encounters is optimal to predict readmission. We conducted experiments by considering x encounters within the prior 2 years, where $x \in \{1, 2, 4, 8, 15, 30, 60, 80, 100\}$. The average number of encounters per patient in this period was 21, and the 90th percentile was 56. The variation in encounter number resulted in a non-unified length of feature vectors. Thus, in an experiment that considers up to 60 encounters, feature vectors lacking data were padded with 0s to ensure that feature vectors for all patients represent 60 encounters. The hypothesis of this study was that DL models outperform traditional models on a large

benchmark, hence, a comparative analysis with a variety of evaluation metrics was performed to evaluate and compare the DL algorithms to the baseline traditional models. Moreover, to examine the importance of domain knowledge, we trained and tested the models on data with all laboratory studies included in the EHR dataset and compared with models trained and tested with a subset of laboratory studies based on prior papers reporting association with readmission (serum albumin, anion gap, arterial pH, bilirubin, blood urea nitrogen, carbon dioxide, serum creatinine, blood glucose, hematocrit, lactate, PaCO2, PaO2, serum sodium, troponin-I, venous pH, and white blood cell count) [52,75]. Using only a subset of laboratory studies may be beneficial by reducing dimensionality.

Patients were sorted randomly into 3 nonoverlapping subsets, where 70% were used for training, 10% for validation, and 20% for testing. We employed cross-validation techniques to find the hyperparameters that yield the best performances. For LSTM and GRU, we varied the number of neurons, dropout, batch size, and the number of epochs using a grid search. Following the literature, in conducted experiments dropout percentage varied from 0 to 50, and the number of neurons varied from 32, to 512. We selected a dropout of 0.1, 128 neurons, a batch size of 512, and 16 epochs for LSTM, and 12 epochs for GRU, since bidirectional GRU converges faster than 1-way LSTM. Sigmoid activation function and Adam optimizer were used. Traditional models were fine-tuned as well and the hyperparameters that yielded best performances were chosen.

3.2.4 Performance Metrics and Analysis

The performance of the methods used in our study was evaluated by five common metrics: Area Under the Receiver Operating Characteristic Curve (AUROC), Recall (also

known as Sensitivity), Specificity, F1-score, and Accuracy. The formal definitions of these evaluation metrics are common and can be easily found [75].

Statistical significance analysis was performed to evaluate the stability and significance of the proposed model's performance. We randomly selected different patients for training and testing and repeated the random selection 10 times to generate mean performance measures and 95% confidence intervals. LSTM was compared to the best performing traditional model (RF) by t-test. A p-value <0.05 was considered statistically significant. The Temple University Institutional Review Board approved the protocol.

3.3 Results

A total of 36,641 patients with 2,836,569 encounters were analyzed. There were 9,130 patients with at least one readmission and 27,511 without a readmission. Influence of the number of encounters within the prior 2 years was evaluated for five prediction models where $\boldsymbol{x}$ encounters were considered for each model, and experiments were repeated for $x \in \{1, 2, 4, 8, 15, 30, 60, 80, 100\}$. Figure 3.2 presents the AUROC of the proposed model, LSTM, versus traditional models across various numbers of encounters. Bidirectional GRU was also performed but omitted because it achieved an identical AUROC to LSTM. LSTM outperformed traditional models on a large benchmark across all experiments with different number of encounters. On average, the LSTM models yielded an increase in AUROC of 0.06 when compared to the best performing traditional models, RF. Experiments show that predicting readmission based on a single prior encounter is not sufficient and yielded much lower performance (0.7 using the DL models and 0.68 using the best performing traditional model). DL models reached a plateau when trained using data from 30 encounters with

minimal improvement thereafter. The DL algorithm yielded 0.07 increase in AUROC versus best performing traditional model RF when using the optimal number of encounters, 80.

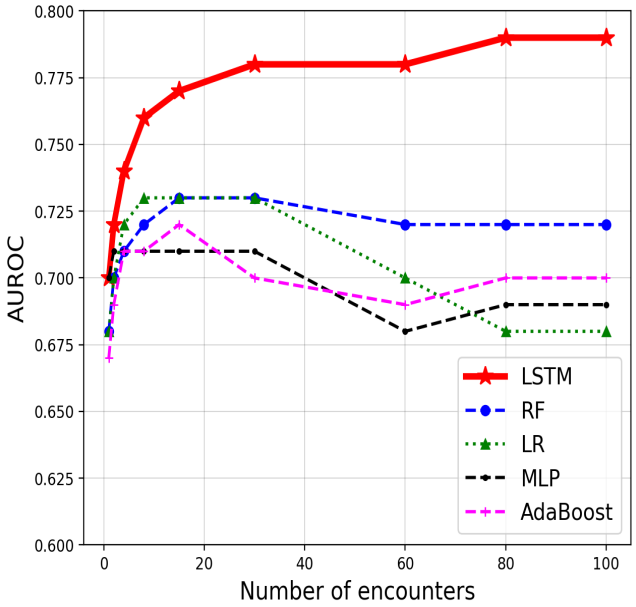

Figure 3.2: The proposed Deep Learning (DL) method's performance compared to baselines evaluated using the area under receiver operating characteristic (AUROC) metric across varying numbers of prior encounters. Results show that DL methods outperform traditional methods on a large benchmark, and improvement in AUROC reaches a plateau at 80 encounters. Results show that using historical data of up to 30 encounters might be sufficient, 80 is optimal, and ≤15 yields poor performance.

LSTM - Long Short-Term Memory; RF - Random Forest; LR - Logistic Regression; MLP - Multi-Layer Perceptron.

Table 3.1 shows the performance of LSTM and traditional models using all laboratory tests from up to 80 of the most recent encounters in the prior 2 years. Overall, the confidence intervals were very small (<0.02), indicating a high degree of precision around the means. The proposed method, LSTM, obtained an average AUROC of 0.79 with a 95% CI of 0.001. The p-value obtained by comparing the LSTM AUROC to the second best-

performing model (RF) was <0.0001, hence, LSTM performance was significantly greater than the traditional models.

Table 3.1: Performance of LSTM and traditional models using all laboratory tests from up to 80 of the most recent encounters in testing cohort of 7,329 patients with diabetes mean +/- 95% confidence interval (CI) are based on 10 runs.

Model N=7,329	**AUROC +/-CI**	**Recall**	**Specificity**	**F1-score**	**Accuracy**
LSTM	**0.79 ±0.001**	**0.81 ±0.002**	**0.94 ±0.010**	**0.80 ±0.003**	**0.81 ±0.002**
RF	0.72 ±0.004	0.76 ±0.001	0.97 ±0.019	0.71 ±0.002	0.76 ±0.001
AdaBoost	0.70 ±0.000	0.76 ±0.000	0.94 ±0.000	0.73 ±0.000	0.77 ±0.000
LR	0.69 ±0.000	0.77 ±0.000	0.91 ±0.000	0.75 ±0.000	0.77 ±0.000
MLP	0.69 ±0.006	0.75 ±0.009	0.87 ±0.018	0.74 ±0.006	0.75 ±0.009

LSTM - Long Short-Term Memory; RF - Random Forest; LR - Logistic Regression; MLP - Multi-Layer Perceptron.

LSTM models achieved a Recall/Sensitivity of 0.81, indicating that performance was fairly strong at predicting true positives, (i.e., correctly classifying patients with readmissions). All models used in our study achieved a very good specificity, (i.e., the true negative rate). Thus, the trained models performed well at predicting patients who are not likely to be readmitted. LSTM achieved an F1-score of 0.80, indicating very good ability to distinguish between patients who will be readmitted or not.

To determine whether domain knowledge about laboratory studies is helpful, we conducted two different experiments where we trained and tested the model based on a subset of 16 unique laboratory studies selected by domain knowledge versus using all 981 unique laboratory studies included in the data. One Hot encoding techniques were utilized and modified to associate the laboratory result with each laboratory code. A long unique array of laboratory codes $unique\_lab\_codes$ was created. For each encounter, an array of

zeros l of the same length as $unique\_lab\_codes$ was created. l consisted of the result at the same index of each laboratory test in $unique\_lab\_results$, to associate the result to a given laboratory code. An encounter without laboratory results would have an l of zeros, indicating that no laboratory test was conducted for a given encounter. Since most encounters contained <3 laboratory codes, this resulted in a sparse array. SVD was therefore utilized to learn an embedding of a sparse feature vector and reduce dimensionality. The Receiver Operating Characteristic (ROC) Curves of the LSTM models based on all laboratory studies or selected laboratory studies were identical (0.79, Figure 3.3).

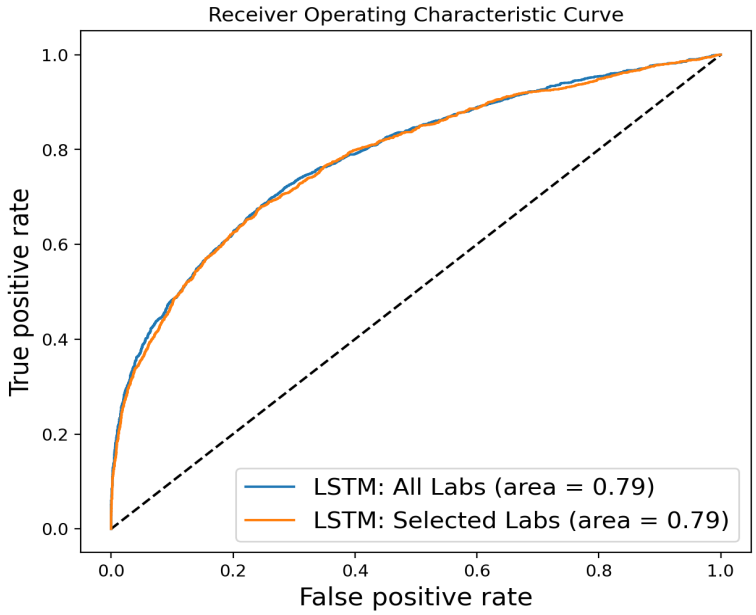

Figure 3.3: Receiver Operating Characteristic (ROC) curves of the LSTM models using all laboratory studies or 16 selected laboratory studies.

3.4 Discussion

In this retrospective cohort of 36,563 patients with diabetes, DL models outperformed RF, MLP, AdaBoost, and LR models at predicting unplanned, all-cause 30-day readmission. The optimal LSTM model yielded an AUROC of 0.79 and accuracy of 0.81, indicating very good performance. Experiments designed to reveal the relationship between the number of prior encounters and model performance show that AUROC of the LSTM models increased as encounter number increased and plateaued at 30 encounters. Performance of the traditional models increased to a lesser extent up to prior encounter numbers of 15 or 30, then either plateaued (RF) or declined (MLP, AdaBoost, LR) as encounter number increased. Finally, an LSTM model that included a set of 16 laboratory tests selected by domain knowledge yielded equivalent performance to an LSTM model that included all available laboratory tests.

In our study, the DL models performed better than the traditional models. We are aware of 4 studies that compared DL models to traditional models for predicting readmission risk of patients with diabetes. Two of these studies demonstrated a clear advantage of DL approaches over traditional ML models [64,65], while two studies found marginal benefit with DL approaches [66,68]. Performance of these DL models was variable with AUROC 0.61-0.97 and accuracy 0.69-0.95, none of which exceeded that of the best traditional ML models, which reported AUROC as high as 0.99 and accuracy of 0.99 [64,68]. Comparisons of model performance across all these studies, however, is limited by the lack of standardized reporting of performance characteristics and variable approaches to testing. Our study considered with the prior studies that directly compared DL to traditional ML models, suggests that DL approaches usually yield better performance in this population.

We are unaware of other papers that have explored the relationship between the number of prior encounters and readmission risk model performance in patients with diabetes. In related work, however, one paper examined how the performance of models for predicting readmission risk in morbidly obese patients varied as the number of hospitalizations increased from a minimum of 2 up to 5 [76]. AUROC increased from 2 to 3 hospitalizations then plateaued. In another broadly related study, we found that the performance of LR models for predicting readmission risk of patients with diabetes tended to increase as sample size increased from 2,000 up to 6,000, then plateaued [77]. This body of research suggests that experimentation across a range of encounter number and sample size may reveal thresholds that could optimize data analysis, balancing information quantity with dimensionality.

We are also unaware of other studies that have compared readmission risk models using laboratory data selected by domain knowledge with all laboratory data available in patients with diabetes. There is a tradeoff between including all laboratory data, which results in higher dimensionality and more computationally expensive models and involving a domain expert to select a subset of laboratory data, which can be costly and less feasible. Like the number of prior encounters beyond which model performance did not improve, the finding that performance of the model with the laboratory data subset was equivalent to the model with all laboratory data suggests that there is a similar plateau for this domain. Whether or not this phenomenon generalizes to other patient populations should be investigated.

The presented LSTM models, which we are calling *e*DERRI™, are an extension of our prior models, the DERRI™ and DERRIplus [49,52]. In terms of AUROC, the *e*DERRI™ model performed better than the DERRI™ but worse than the DERRIplus. Unfortunately,

the performance of 3 models cannot be directly compared in the current study because the dataset does not include zip code, employment status, or payer information. Unlike the DERRI™ and DERRIplus, the *e*DERRI™ models are developed with generally available EHR data such as demographics, vital signs, diagnostic and procedure codes, medications, laboratory tests, and administrative data as defined by the PCORnet CDM [70]. The CDM standardizes the abstraction of EHR data, enhancing the generalizability and scalability of models utilizing it. We plan to translate the *e*DERRI™ into an application embedded in an EHR system that will automatically generate readmission risk predictions for hospitalized patients with diabetes.

In addition to the generalizability of the CDM-based dataset, the current study has other notable strengths. The dataset is sampled from patients with a hospitalization between 7/1/2010 and 12/31/2020, which is much more recent than the datasets used for other currently published readmission risk models in diabetes patients. Also, in contrast to the most used dataset, which only included hospital encounters with an associated diagnosis of diabetes and a length of stay less than 15 days, the current dataset included all encounter types regardless of the associated diagnosis, capturing both inpatient and outpatient data. Lastly, the sample size of 36,563 patients with 2,836,569 encounters provided ample data to develop DL models and conduct experiments up to 100 prior encounters.

There are some limitations worth acknowledging. The data were sampled from a single urban, academic health system. Therefore, generalizability of the models to other populations is unknown and requires testing. The lack of both patient and hospital zip code precludes estimating the distance between a patient's home zip code and the hospital,

which is known to be associated with readmission risk [49,52]. Lastly, readmissions to other hospitals were not captured.

3.5 Conclusion

An LSTM model with very good performance predicting unplanned, all-cause 30-day readmission among patients with diabetes was developed and internally tested. LSTM models outperform traditional models at predicting readmission in this population. LSTM model performance initially increases as the number of prior encounters increases then plateaus. Carefully selected laboratory features can yield predictive models with performance equal to that of models based all available laboratory studies. Additional study is needed to externally validate the model.

This chapter establishes the foundational framework for the subsequent chapters. The primary metric employed was the AUROC, known for its effectiveness in datasets with balanced class distribution. However, given its limitations in scenarios of significant class imbalance, the focus is redirected towards the F1 score in the later chapters of the study.

3.6 Acknowledgements

This research was supported by the National Health Institute (NIH) under grant number R01DK122073.

CHAPTER 4

KNOWLEDGE TRANSFER WITH DEEP ADAPTATION NETWORK FOR PREDICTING HOSPITAL READMISSION

4.1 Introduction

Previously, we published a risk deep learning (DL) model based on electronic health records (EHR) data collected from an urban academic hospital that predicts the risk of unplanned, 30-day readmission among patients with diabetes. We used a sequential model, long short-term memory (LSTM). Performance was adequate (F-1 Score 0.80), and results showed that this LSTM model can capture temporal dependencies of the EHR data [1].

Performant readmission models based on DL techniques require large, high-quality training data to perform optimally. Utilizing EHR data from a source hospital system to enhance prediction on a target hospital using traditional approaches enlarge dataset bias which might deteriorate performance due to distributional difference of the source and target datasets, resulting in statistically unbounded risk for the target tasks [79]. Traditional approaches are designed for a specific data type, and not capable of generalizing to other temporal data.

Transfer learning approaches have been explored for hospital readmission with the objective to improve learning at the target population by exploiting information from a

related source population. In [80, 81], classical transfer learning was employed to address data scarcity using a relevant source dataset. In [82], classical transfer learning techniques were explored as to what extent can transfer learning benefit learning on target tasks by fine-tuning pre-trained models in the healthcare domain. However, there is still a need for an end-to-end model to perform cross-domain spatial knowledge transfer and predictive learning in a unified learning framework while capturing temporal dependencies for hospital readmissions.

In this paper, we propose an early readmission risk temporal deep adaptation network, ERR-TDAN, to perform cross-domain spatial knowledge transfer from EHR data of different sites and perform predictive learning. Deep Adaption Network (DAN) utilizes deep convolutional neural network (CNN) and generalizes it to the domain adaptation setting through learning transferable latent features between source and target domains for computer vision tasks [79]. Motivated by the success of DAN in numerous transfer learning tasks in computer vision, we employed the idea of learning transferable features of temporal data by matching the source and target domain distributions in the latent feature space. We tailored it for hospital readmission using EHR data and optimized for the target task.

The aims of this study were as follows: 1) To develop a hospital readmission framework using EHR data that transfers knowledge between a rural academic hospital and an urban academic hospital to enhance predictions on the urban academic hospital. 2) To study the optimal amount of retrospective EHR data needed for future predictions. 3) To study the duration of optimal performance. Experiments conducted show that ERR-TDAN can enhance hospital readmission prediction.

4.2 Deep Adaptation Network

Domain adaptation is a form of transfer learning commonly used in computer vision to address the problem of learning using data from two related domains but under different distributions [79, 83]. Domain adaptation can help improve the performance of a model by learning transferable features to minimize the gap between the source and target domains in an isomorphic latent feature space. DAN generalizes deep CNN for computer vision applications to utilize domain adaptation techniques to learn transferable feature representation in the latent embedding space [79]. Motivated by the success of DAN in various computer vision tasks [84-86], we utilized the idea of DAN for transferring cross-domain spatial knowledge tailored for predicting hospital readmission on EHR data and optimized to enhance predictions on the target, rather than generalizing on both domains. A direct comparison to DAN is not applicable since DAN is modified for computer vision tasks using CNN. CNNs capture spatial correlations and are unable to capture temporal correlations of EHR data [79]. Thus, we employed the idea of DAN and tailored it for hospital readmission on EHR data to capture temporal dependencies using LSTM layers, establish cross-domain knowledge transfer, and optimized it for the target task using a customized loss function.

4.3 The Proposed ERR-TDAN Framework

An early readmission risk framework based on temporal deep adaptation network was developed to enhance prediction on the target data collected from Temple University Hospital System (TUHS) by establishing spatial knowledge transfer from a source data with higher quality features collected from Penn State University Hospital System

(PSUHS). The model was developed using data as defined by the National Patient-Centered Clinical Research Network (PCORnet) Common Data Model (CDM) [70].

We applied a hospital readmission LSTM model that we previously published using EHR data collected from TUHS [1]. When trained on TUHS data and tested on the following year TUHS data this model F-1 score was 0.80. We trained and tested the same method on EHR data collected from PSUHS, where performance was better (F1-score 0.91). The 11% increase in F-1 score was achieved since EHR data from PSUHS contained fewer missing data, denser features, and less erroneous data. However, training and evaluating the same method on data from both domains affected the performance (F-1 score 0.79) since the model struggled to generalize and converge due to training data drawn from different distributions. To address this limitation, we employed the idea of DAN, tailored for hospital readmission on EHR data that captures temporal correlations and enhances target prediction through learning transferable features via domain-specific fully connected linear layers to explicitly reduce the domain discrepancy. DAN generalizes on both domains for computer vision tasks, whereas in our study we tailored this technique for hospital readmission using EHR data and optimized on the target task, instead of generalizing on both domains. To accomplish this, the hidden embeddings of the domain-specific layers are embedded to a reproducing kernel Hilbert space through maximum mean discrepancy (MMD), to match the mean embeddings of two domain distributions. The model was optimized via a customized loss function.

Figure 4.1 presents the proposed framework, ERR-TDAN which consists of the following main processes:

1) LSTM's input was data from both source and target to learn hidden representation to map source and target data to a common embedding while capturing temporal dependencies of the EHR data.
2) To match the embedding distributions of the source and target domains, deep adaptation network scenario is established through fully connected linear layers constructed to match the mean embeddings of different domain distributions. The hidden representation is embedded through a reproducing kernel Hilbert space to transfer knowledge and bridge the gap between two distributions via MMD to reduce domain discrepancy.
3) The matched embeddings are then passed to a fully connected layer with a sigmoid function to classify if a patient is likely to be readmitted or not. In backpropagation, we optimize the model on the target domain using a customized loss function that combines the domain discrepancy loss and binary cross entropy loss. The following sections illustrate the framework in more detail.

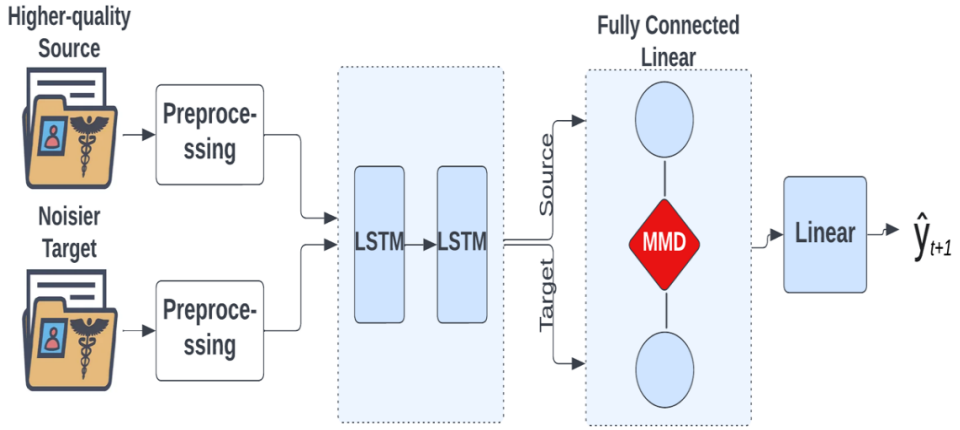

Figure 4.1: The proposed method framework, ERR-TDAN. It comprises of three main processes. 1) LSTM layers learn hidden representation of the input of both source and target domains. 2) Deep adaption network structure with fully connected layers is constructed to match the mean embeddings of different domains drawn from different distributions. 3) The matched embeddings are then passed to a fully connected layer with sigmoid function for binary classifications. The model is optimized through a customized loss function that penalizes on domain discrepancy of both source and target, and task loss to optimize for the target task.

4.3.1 Representation Learning of Temporal EHR Data with LSTM

Initially, we utilized LSTM with two recurrent layers to form a stacked LSTM to learn hidden data representation embedded to a common latent feature space of the temporal EHR data of the source and target domains. LSTM, a sequential model capable of capturing temporal correlations, is commonly used for sequential tasks and is proven to be effective for hospital readmission using EHR data [1, 87]. LSTM takes as an input a 3-dimensional tensor of stacked source and target data. LSTM is structured based on basic neural network, but neurons of the same layer are connected, enabling a neuron to learn from adjacent layers, in addition to learning from outputs of the previous layers and the input data. Hence, neurons include two sources of inputs, the recent past and the present. A dropout of 0.1

was applied between the first and second LSTM layers. To add nonlinearity, we utilized ReLU activation function on the output of LSTM (embeddings), formulated as follows:

$$\boldsymbol{b^t = \mathbf{ReLU}\,(b + Wh^{t-1} + Ux^t)} \tag{4.1}$$

4.3.2 Learning Transferable Features and Predictions

The output b^t is then fed to domain-specific fully connected linear layers with deep adaptation network setting. Domain discrepancy is reduced by matching the mean embeddings of the source and target distributions. Hidden representation of the linear layers embedded through a reproducing kernel Hilbert space to bridge the gap between two distributions and transfer knowledge via MMD. MMD measures the distance of the source and target distributions in the embedding space. MMD distance measure was originally used to determine whether two samples are drawn from the same distribution and measures how distant the samples are [88]. In this study, we utilized MMD to learn transferable features between source and target domains to enhance prediction on the target. MMD was utilized as one of the two components of the loss function to minimize the domain discrepancy. The loss function is explained in more detail in the next section. MMD is defined as:

$$\boldsymbol{MMD_{loss}(\mathcal{D}^S, \mathcal{D}^T) = \left\| \frac{1}{n}\sum_{i=1}^{n} \phi\left(d_i^S\right) - \frac{1}{m}\sum_{j=1}^{m} \phi\left(d_j^T\right) \right\|_{\mathcal{H}}^2,} \tag{4.2}$$

where $\mathcal{D}^S$ and $\mathcal{D}^T$ denote source and target data respectively, ϕ denotes the Gaussian kernel function, $\mathcal{H}$ denotes the Hilbert space, and n and m denote the number of observations of

the source and target sets, respectively. The temporal embeddings of the LSTM are then fed into fully connected layers with MMD loss to measure the distance between two distributions and reduce domain discrepancy.

Prediction. The matched embeddings are then fed into a linear layer with output of 1 with sigmoid activation function for predictions $\hat{y}$ [1].

4.3.3 Model Optimization via a Customized Loss Function

We tailored the loss function for hospital readmission on the target domain by combining Binary Cross Entropy (BCE) loss to measure the error of reconstruction, applied on the target task only, and MMD loss applied on both source and target to reduce domain discrepancy. Since the aim of this study is to enhance prediction on the target domain using higher-quality source data, we reduced the weight of the MMD loss via the penalty parameter γ and optimized the loss on the target domain. Loss function used in the proposed ERR-TDAN model is defined as follows:

$$\begin{aligned} \boldsymbol{BCE_{loss}} = (\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{L} = \{\boldsymbol{l_1}, \dots, \boldsymbol{l_N}\}^\top, \boldsymbol{l_n} \\ = -\boldsymbol{w_n}[\boldsymbol{y_n} \cdot \mathbf{log}\, \boldsymbol{x_n} + (\mathbf{1} - \boldsymbol{y_n}) \cdot \mathbf{log}(\mathbf{1} - \boldsymbol{x_n})], \\ \boldsymbol{TOTAL_{loss}} = \frac{\mathbf{1}}{\boldsymbol{L^N}} \sum_{\boldsymbol{t=1}}^{\boldsymbol{L^N}} (\boldsymbol{BCE_{loss}}(\boldsymbol{x}, \boldsymbol{y}) + \boldsymbol{\gamma}\, \boldsymbol{MMD_{loss}}(\boldsymbol{d^S}, \boldsymbol{d^T})), \end{aligned} \tag{4.3}$$

where x and y are the predictions and ground truth for a given batch respectively. L denotes loss. N is the batch size, w is a rescaling weight given to the loss of each batch element, γ is the penalty parameter of domain discrepancy. To optimize for the target task, we determined empirically that 0.5 value of γ is appropriate.

4.4 Data

We collected data from an urban academic hospital, TUHS, and a rural academic hospital, PSUHS, between July 1st, 2010, and December 31st, 2020. We extracted data on encounters, demographics, diagnosis, laboratory tests, medication orders, procedures, and vital signs. in the cohort of patients with diabetes was defined as previously described [1]. Data preprocessing, handling of missingness of data, different number of recordings per encounter, learning embeddings to reduce dimensionality, address sparse feature vectors, and data representation were performed as [1] and as presented in Chapter 3. Additional features were aggregated to assist with learning temporal dependencies, including duration of stay in days, and number of days since the prior encounter. In addition to the 23 features outlined in Section 3.2.2, two more features were derived and incorporated into the model, bringing the total number of features utilized to 25. The two features are income quintile, and distance between the hospital and patient address.

We obtained a total of 1,421,992 encounters corresponding to 20,471 patients for PSUHS, and a total of 3,023,267 encounters corresponding to 37,091 patients for TUHS. The class distributions were as follows. TUHS: 28,107 for the negative class (no readmission), and 8,984 for the positive class (readmitted within 30-days); PSUHS: 18,775 for the negative class and 1,696 for the positive class.

The characteristics of the samples from the two sites were different. For instance, 4.9% of patients were Hispanic at PSUHS, whereas TUHS contained large Hispanic population of 22%. Other differences included race and tobacco use. The numbers of unique ICD-9 and ICD-10 codes, and vital recordings at PSUHS were larger than that at TUHS.

Patient encounters were sequentially ordered by admission date and represented in a 3-dimensional tensor for the LSTM model, where each patient's data is represented as a 2-dimensional matrix in which features of each encounter are represented in a 1-dimensional array while a second dimension represents different hospitalizations of that patient. The third dimension is used to encode hospitalization information of different patients.

4.5 Experimental Setup and Results

We hypothesize that it is feasible to enhance readmission predictions on target data of TUHS using a source data from a relevant domain under different distribution. In this section, we conduct extensive experiments to evaluate the performance of the proposed model, ERR-TDAN and compare it to baselines. F-1 score, precision, recall (sensitivity), specificity, and accuracy were used to evaluate the model's performance [75]. We randomly selected different patients for training and testing. Experiments were iterated 10 times; results were presented based on the mean and two-sided 95% confidence interval (CI). Moreover, we address the following research questions to evaluate optimal performance of the model.

4.5.1 Can we enhance readmission risk prediction for a target hospital by utilizing data from another hospital?

We randomly split TUHS and PSUHS data to 70% training, 10% validation, and 20% testing. Then, we concatenated training data of both domains, and fed to the ERR-TDAN. We tested the model on TUHS using 7,418 patients, of whom 1,557 had a readmission.

Table 1 presents a comparative analysis to evaluate the proposed method, ERR-TDAN compared to alternative baselines. Table 1 shows that ERR-TDAN yielded a 5% increase

in F1-score when compared to a model we previously published for hospital readmission on EHR data collected from TUHS, and 3% increase using a generalized version of ERR-TDAN (G-ERR-TDAN) of the domain adaptation framework with MMD loss without optimizing on the target task. G-ERR-TDAN results provide evidence that optimizing on the target task enhances target's predictions is superior to generalizing on both domains.

Table 4.1: Performance of the proposed method, ERR-TDAN and three alternatives tested on the target domain (TUHS) enhanced by a related source data (PSUHS). The average F1, recall/sensitivity, specificity, and accuracy and their corresponding two-sides 95% confidence interval (CI) on 10 experiments on training and testing patients' data selected completely at random.

Model	Train	F1-score	Recall	Specificity	Accuracy
[1]	TUHS	0.80 ±0.003	0.81 ±0.002	0.94 ±0.010	0.81 ±0.002
LSTM	TUHS + PSUH	0.79 ±0.007	0.81 ±0.006	0.95 ±0.008	0.81 ±0.005
G-ERR-TDAN	TUHS + PSUH	0.82 ±0.001	0.81±0.001	0.92 ±0.002	0.81 ±0.001
ERR-TDAN	TUHS + PSUH	**0.85** ±0.002	0.84 ±0.002	0.91 ±0.003	0.84 ±0.001

4.5.2 What is the retrospective optimal amount of EHR data needed for future predictions?

We conducted extensive experiments to find the optimal amounts of patient's historical data needed for the model to perform optimally. Our objective was to determine a size of training data so that further enlargements do not improve predictions of hospitalization risk. The model was trained on varying t and tested on $t + x$, where t denotes a period in the past and $t + x$ denotes a period in the future. For a fair comparison, $t + x$ was a fixed test dataset of 2020, and trained on varying training sets of t, including 6 months (July-December of 2019), 1 year (2019), 2 years (2018-2019), 3 years (2017-2019), 4 years (2016-2019), and 5 years (2015-2019) look-back time. For instance, training on 2019, and

testing on 2020 (1 year look-back) to test if learning on 1 year of historical EHR data from the past is sufficient to perform optimally.

Figure 2 (left) shows that 1 year of historical data are optimal to predict readmission since it yielded the highest F1-score with least amounts of data required.

4.5.3 How often do we need to retrain the model to achieve optimal performance?

Concept and covariate shifts are one of the major reasons model performances degrade overtime. Monitoring data drift helps avoid performance degradation. Thus, we conducted experiments to study the lifetime of the proposed model. Based on the optimal look-back time of question 2, we trained the model on EHR data collected in 2015 and tested it with 1, 2, 3, 4, and 5 future gaps. For instance, training on data collected in 2015 and testing in 2020 to experiment if the model's performance would degrade after 5 years. We iterated this over various models trained on 1 year of data collected in 2015, 2016, and 2017 and tested for readmissions on future instances.

Figure 2 (right) shows that F1-score decreased over time due to data drift. Performance was relatively stable when tested on 1 and 2 years in the future. There was a significant decrease in F1-score when used to predict readmissions with 3 years gap between training and testing. On overage, F1-score degraded 0.6% when used 3 years later, 3.5% when used 4 years later, and 7% when used 5 years later. Therefore, to maintain optimal performance of hospital readmission models on EHR data, retraining the model every 3 years may avoid model degradation and maintain optimal performance.

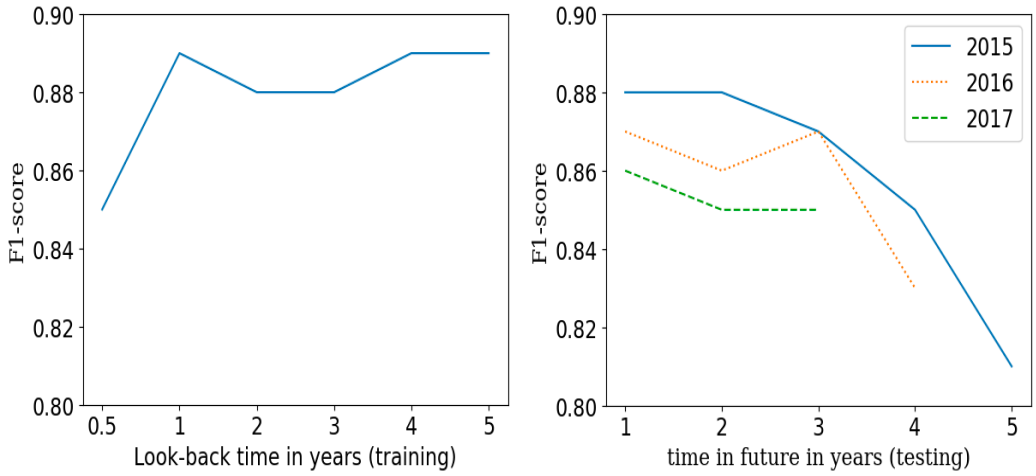

Figure 4.2: (left) presents the retrospective optimal amount of EHR data needed for future predictions. Results show that 1 year of historical data are sufficient to predict hospital readmission from the leading year. (right) presents the lifetime of the model to maintain and achieve optimal performance. Three different models were developed on data collected from 2015, 2016, and 2017 to predict future instances with 1 to 5 years gap. Results show that to maintain optimal performance, the proposed framework may benefit from retraining every three years.

4.6 Discussion and Conclusion

We examined the hypothesis that it is feasible to enhance hospital readmission risk predictions on EHR data using data collected from a related source domain. ERR-TDAN model trained on joint TUHS, and PSUHS data yielded a 5% increase in F1-score when compared to an LSTM model trained on TUHS only, 6% increase in F1-score when compared to LSTM model trained on both TUHS and PUSH, and 3% increase when compared to a generalized version of ERR-TDAN (G-ERR-TDAN) aimed to generalize on both domains. Furthermore, conducted experiments showed that one year of historical data is sufficient to predict readmission. We studied the lifetime of the model to avoid performance degradation due to data drift over time. Experiments suggest that retraining the ERR-TDAN framework every three years avoids performance degradation.

We propose a framework, ERR-TDAN that establishes spatial knowledge transfer based on a temporal deep adaptation network tailored for hospital readmission on EHR data and optimized for the target task. ERR-TDAN can enhance readmission predictions of the target task using higher quality data from a related source domain under different distributions by matching the mean embeddings to reduce domain discrepancy. This is the first end-to-end transfer learning framework based on domain adaptation for hospital readmission. A deployment challenge for the proposed framework is that it requires training data from both source and target domains which might be difficult to obtain. In a planned follow up study we will evaluate applicability of the proposed method for prospective applications. In addition, we will compare the proposed hospital readmission method to alternatives aimed to learn from integrated data with explanatory variables of various quality.

4.7 Acknowledgements

This research was supported by the National Health Institute (NIH) under grant number R01DK122073.

CHAPTER 5

DOMAIN GENERALIZATION FOR ENHANCED PREDICTIONS OF HOSPITAL READMISSION ON UNSEEN DOMAINS

5.1 Introduction

We previously published multiple early readmission risk prediction models based on deep learning (DL) among people with diabetes. Our long short-term memory (LSTM) model was trained on data collected from an urban hospital to identify patients at high risk of readmission within the same hospital, presented in Chapter 3. This model exhibited good performance with an F-1 Score of 0.8, indicating that this LSTM model can recognize temporal dependencies [1]. We also developed a novel temporal deep adaptation network-based model for early readmission risk, called ERR-TDAN, to enhance predictions on the target task by transferring knowledge from high-quality source data, presented in Chapter 4 [6]. Baseline results of this study provided evidence that traditional approaches fail to generalize on unseen target domains when source and target data have different distributions. Utilizing domain adaptation techniques to enhance predictions on a seen target domain was shown to be effective in reducing domain discrepancy of data drawn from different distributions collected from different hospitals. ERR-TDAN is designed to

accept source and target data in training as input, thus, it is not capable of generalizing on unseen target domains.

Collecting data from multiple hospitals and developing performant readmission models for every site may not be feasible for many institutions. Another potential limitation is having sufficient historical data. Developing a readmission risk model based on data from source hospitals to predict readmission on an unseen test hospital using our previous and other conventional approaches is not effective because these methods are not capable of generalizing well to unseen test domains when training and testing distributions are different [6, 79].

Transfer learning methodologies have been investigated in the context of hospital readmission, aiming to enhance the learning of the target population by leveraging insights from a related source population. The studies reported in [80, 81], transfer learning is successfully applied to mitigate the challenges of limited data by utilizing a relevant source dataset. Conversely, the study reported in [82], potential benefits of transfer learning are investigated by assessing the fine-tuning capabilities of pre-trained models within the healthcare domain. Nonetheless, none of the aforementioned methods are capable of generalizing to unseen data collected from multiple hospital systems drawn from different distributions. Therefore, there remains a demand for an end-to-end model that performs cross-domain knowledge transfer that is capable of generalizing on previously unencountered domains in a unified framework, while capturing and maintaining long-term temporal dependencies for hospital readmissions.

In this paper, we propose an early readmission risk domain generalization network, ERR-DGN, to perform cross-domain knowledge transfer from electronic health record

(EHR) data of different health systems to facilitate predictive learning. Motivated by the success of our previous study reported [6] that leverages Deep Adaptation Network (DAN) techniques to enhance readmission risk predictions on a target hospital by leveraging high-quality source data and matching mean embeddings of source and target distributions, we employed the idea of learning transferable features of the EHR data by matching multiple source distributions in the latent space to generalize and enhance predictions on an unseen target task. ERR-TDAN [6] takes as an input two sites (i.e., source and target) and requires historical training data from both. In contrast, we tailored ERR-TDAN to learn transferable features of multiple source datasets to predict rehospitalization risk on an unseen target hospital where data distribution might be significantly different from data at all previously observed hospitals. We hypothesized that this novel approach would improve hospital readmission risk predictions among people with diabetes for a previously unobserved target domain.

5.2 Methods

5.2.1 Domain Generalization

Domain generalization (DG) is a form of knowledge transfer (i.e., transfer learning) commonly used in computer vision tasks to address the problem of learning from related source domains but under different distributions. DG pertains to the ability of models to generalize knowledge from multiple source domains to a previously unobserved target domain. The main goal is to transfer knowledge from the known source domains to be effective on the unknown target domains. The challenge is to ensure that the learned model does not overfit to the idiosyncrasies of the source domains but captures more general

patterns or invariants that are effective across all domains. Unlike traditional transfer learning, which typically focuses on adapting a model trained on a source domain to a specific known target domain, domain generalization aims to equip the model with the capability to perform effectively across any potential target domain without prior exposure to the model. This is achieved by emphasizing the learning of transferable and domain-invariant features from the source domains. By training on multiple source domains, the model ideally internalizes the shared patterns and characteristics that are consistent across those domains, making it more robust and adaptable when confronted with data from a new domain. While domain adaptation focuses on adjusting a model trained on one domain to perform well on a specific different domain using target domain data, domain generalization aims to train a model on multiple domains to perform well on any new domain without accessing data from the unseen new domain. Both are types of transfer learning, but their objectives can differ. DG thus addresses the challenges of domain shift and dataset bias, making it especially valuable in fields where collecting data for every possible domain is infeasible or costly. DG can be effective for hospital readmission risk estimation since the process of collecting EHR data may be infeasible in some settings. Furthermore, DG can be effective in hospital systems where historical data is not available and a readmission risk model is needed [6, 79, 89, 83].

5.2.2 Formalization

1) *Domain.* Let $\mathcal{X}$ denote an input space and $\mathcal{Y}$ an output space. A domain is composed from a data sampled from a distribution, denoted as $\mathcal{S} = \{(\mathrm{x}_i, y_i)\}_{i=1}^{n} \sim P_{XY}$, where $\mathrm{x} \in \mathcal{X} \subset \mathbb{R}^d$, $y \in \mathcal{Y} \subset \mathbb{R}$ denotes the label, and P_{XY} denotes the joint distribution of the input sample and output label.

2) *Domain Generalization.* An input that consists of M source domains (training set) we denote as $\mathcal{S}_{\text{train}} = \{\mathcal{S}^i \mid i = 1, \cdots, M\}$, where $\mathcal{S}^i = \{(\mathrm{x}_j^i, y_j^i)\}_{j=1}^{n_i}$ denotes the $i-th$ domain. The joint distributions between each pair of domains are different, $P_{XY}^i \neq P_{XY}^J, 1 \leq i \neq j \leq M$. The aim of DG is to learn a robust and generalizable predictive function from the M training source domains to achieve a minimum prediction error on an unseen test domain $\mathcal{S}_{\text{test}}$ that cannot be seen during training of the model and $P_{XY}^{\text{test}} \neq P_{XY}^i$ for $i \in \{1, \cdots, M\}$.

5.3 Methodology

5.3.1 Distributional Difference and Selection of Unseen Target Domain

This section details the methodology employed to validate the applicability of utilizing domain generalization on EHR data and identify a particularly challenging site to evaluate the proposed ERR-DGN method. We conducted pairwise comparisons, analyzing the data distributions from the five sites, to select the test site that exhibited the most divergence from the others. We utilized Kolmogorov-Smirnov (KS) statistic test for pairwise comparisons of similarity between two probability distributions functions G and F to check if two distributions from M domains are identical. This is achieved by comparing the underlying distributions F(x) and G(x) of two intendent samples [38], where x denotes a sample consisting of patient data. The null hypothesis was F= G, indicating that the distributions are identical. We applied KS test on two independent samples from different domains, with each sample being a 1-dimensional array representing a patient's data. From each domain, we randomly selected 2000 patients/sample in which 1000 patients had readmission (positive class) and 1000 patients are without readmission (negative class).

Each sample was then compared with every sample from the other domain. This was reiterated to assess the underlying distributions between every pair of domains. We obtained p-values for each pairwise comparison and computed the mean p-value.

5.3.2 The proposed ERR-DGN Framework

An early readmission risk framework based on domain generalization techniques was developed to generalize and enhance predictions on an unseen target academic hospital system by establishing cross-domain knowledge transfer from four different academic hospital systems based on EHR data. Motivated by the success of ERR-TDAN on EHR data [6], we employed the concept of DAN for transferring cross-domain knowledge tailored for predicting hospital readmission on EHR data through learning transferable features via domain-specific fully connected linear layers to reduce domain discrepancy and optimized to generalize on several domains to enhance predictions on an unseen target domain. To accomplish this, the hidden embeddings of the domain-specific layers are embedded to a reproducing kernel Hilbert space through maximum mean discrepancy (MMD), to match the mean embeddings of domain distributions. We customized the loss function to optimize ERR-DGN to learn general patterns and characteristics from several source domains.

Fig. 5.1 shows the proposed ERR-DGN framework, encompassing the following main processes:

1) ERR-DGN accepts EHR data from multiple source domains as input. These datasets are then channeled through domain-specific LSTM layers, aiming to learn hidden representations that map source data to a common embedding while

capturing the temporal dependencies of the EHR data. Then, attention mechanism is applied to weigh the importance of different parts of the sequence differently, allowing the network to decide which parts of the input sequence to focus on.

2) In order to match the embedding distributions of the source domains, deep adaptation network setting is formulated through domain-specific fully connected linear layers designed to match the mean embeddings across varying distributions. The hidden representations subsequently embedded through a reproducing kernel Hilbert space, utilizing pairwise computations to compare embedding distributions among all pairs. This process aids in transferring knowledge and mitigating the differences between distinct distributions by employing MMD to reduce domain variance.

3) The matched embeddings are then concatenated and passed to a fully connected linear layer with a sigmoid activation function for binary classification to pinpoint patients that are likely to be readmitted. In evaluating the framework's performance, a sample is processed via a singular domain-specific layer set, rather than all four. In backpropagation, we optimize the model to generalize on multiple source domains via a customized loss function that combines the sum of pairwise MMD loss calculations leveraged for domain discrepancy and binary cross entropy loss. The following sections elucidate the main processes of the framework in more detail.

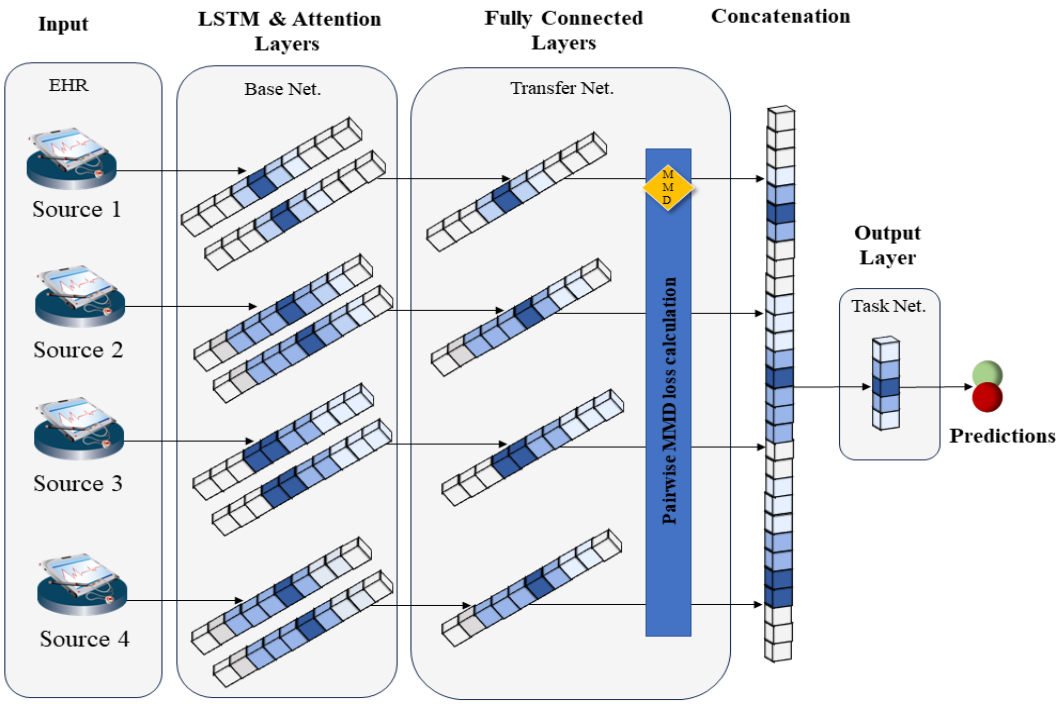

Figure 5.1. The proposed ERR-DGN framework. ERR-DGN consists of three primary processes. Firstly, domain-specific LSTM layers derive hidden representations from the input of source domains, subsequently Attention mechanism is applied to extract relevant information. Secondly, a domain generalization network techniques utilized with fully connected layers designed to match the mean embeddings from domains drawn from different distributions. Lastly, the matched embeddings are then concatenated and passed through a fully connected layer with sigmoid activation function for binary classification. The optimization of this model is facilitated through a customized loss function that penalizes on domain discrepancy of source domains to aid in learning general patterns and characteristics that are shared across source domains.

5.3.2.1 Learning Representations of Temporal EHR Data via LSTM and Attention.

Initially, we employed a stacked LSTM consisting of two recurrent layers. This aimed to learn hidden data representations embedded to a common latent feature space of the temporal EHR data of source domains. LSTM, a sequence-based model capable of capturing temporal dependencies, has been widely employed for sequential tasks and has demonstrated efficacy in predicting hospital readmission using the sequential and temporal

EHR data [1, 6, 87]. LSTM layers accept 3-dimensional tensors of source data as input. It is built upon the foundation of neural networks, in LSTM, neurons within the same layer are interconnected. This allows a neuron to learn not only from the outputs of preceding layers and the current input but also from its neighboring layers. Consequently, each neuron has dual input sources, the immediate past and the present. A dropout rate of 0.1 was applied between the first and second LSTM layers. To introduce nonlinearity, the ReLU activation function was applied to the LSTM output (embeddings), formulated as follows:

$$\boldsymbol{b^t} = \mathbf{ReLU}(\boldsymbol{b} + \boldsymbol{W h^{t-1}} + \boldsymbol{U x^t}). \tag{5.1}$$

Here, $\boldsymbol{b^t}$, the output, is then fed to domain-specific attention layer which serves to weigh the importance of different of certain features in the sequence differently, allowing the network to focus on relevant features of the sequence. The attention mechanism computes a weighted sum of all the hidden states of the LSTM, based on their relevance to the current timestep of the output sequence. The weights for this weighted sum are dynamically computed for each output timestep. Adding attention mechanism enhances LSTM layers by allowing the network to decide which features to focus on, leading to improved performance. This process is illustrated at a high level as the "base net" process in Fig. 5.1.

5.3.2.2 Learning Transferable Features and Enhancing Predictive Performance. The hidden embeddings are subsequently passed to domain-specific fully connected layers within a deep adaptation network framework. To minimize domain discrepancy, the mean embeddings from multiple source distributions are matched. The representations from these linear layers are projected into a reproducing kernel Hilbert space, facilitating the

bridging of gaps distributions by utilizing pairwise computations to compare embedding distributions among all pairs, then the sum is computed quantifying the distance between distributions within the embedded space. and enabling knowledge transfer using MMD. The MMD distance metric was originally employed to determine if two samples originate from identical distributions, gauging the extent of their divergence [93]. In this study, MMD was utilized to generalize on multiple source domains by learning shared patterns and characteristics that are consistent across source domains, making it more robust and adaptable when applied with data from a new unseen domain. MMD served as the key component of the loss function, aiming to minimize domain discrepancies. A comprehensive explanation of the loss function is provided in the subsequent section. MMD can be defined as:

$$\boldsymbol{MMD_{loss}}(\boldsymbol{S^1}, \boldsymbol{S^2}) = \left\| \frac{\mathbf{1}}{\boldsymbol{n}} \sum_{\boldsymbol{i=1}}^{\boldsymbol{n}} \boldsymbol{\phi}\left(\boldsymbol{s_i^1}\right) - \frac{\mathbf{1}}{\boldsymbol{m}} \sum_{\boldsymbol{j=1}}^{\boldsymbol{m}} \boldsymbol{\phi}\left(\boldsymbol{s_j^2}\right) \right\|_{\mathcal{H}}^{\mathbf{2}}, \tag{5.2}$$

where S^1 and S^2denote two distinct source domains data, ϕ denotes the Gaussian kernel function, $\mathcal{H}$ denotes the Hilbert space, and n denote the number of samples in $S^1$and m denote the number of samples in S^2. MMD is iteratively applied utilizing pairwise calculation setting to compare distributions of all source domains among all pairs, where $\mathcal{S}^i = \{i = 1, \cdots, M\}$ and the joint distributions between each pair of domains are different, $P_{XY}^i \neq P_{XY}^J, 1 \leq i \neq j \leq M$. The temporal embeddings of the LSTM are then fed into fully connected layers with MMD loss to measure the distance between every pair of

distributions and minimize domain discrepancy. This process is illustrated at a high level as the "transfer net" process in Fig. 5.1.

Output. The matched embeddings are subsequently inputted into a linear layer with a single output, coupled with a sigmoid activation function, to derive probabilities for pinpointing patients at high risk of readmission. Utilizing the default threshold of 0.5, high risk patients are distinguished, and predictions are generated $\hat{y}$ [1].

5.3.2.3 Customized Loss Function for Model Optimization. To achieve generalization across multiple source domains, we modified the loss function by integrating MMD loss and Binary Cross Entropy (BCE) loss. MMD was utilized using pairwise calculations to assess the discrepancies among distributions, thereby internalizing patterns and characteristics that are consistent across source domains. BCE was utilized to measure the error of reconstruction, applied on each source domain. The sum of BCE on each source domain and MMD loss among source distributions was computed as the loss. Loss function used in the proposed ERR-DGN model is defined as follows:

$$\begin{aligned} \boldsymbol{BCE_{loss}} &= (\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{L} = \{\boldsymbol{l_1}, \dots, \boldsymbol{l_N}\}^{\top}, \boldsymbol{l_n} \\ &= -\boldsymbol{w_n}[\boldsymbol{y_n} \cdot \log \boldsymbol{x_n} + (\mathbf{1} - \boldsymbol{y_n}) \cdot \log(\mathbf{1} - \boldsymbol{x_n})], \\ \boldsymbol{TOTAL_{loss}} &= \frac{\mathbf{1}}{\boldsymbol{L^N}} \sum_{\boldsymbol{t=1}}^{\boldsymbol{L^N}} \left(\left(\sum_{\boldsymbol{i=1}}^{\boldsymbol{M}} \boldsymbol{BCE_{loss}}(\boldsymbol{x_i}, \boldsymbol{y_i}) \right) + \left(\sum_{\boldsymbol{i=1}}^{\boldsymbol{P}} \boldsymbol{\gamma MMD_{loss}}(\boldsymbol{s_i^1}, \boldsymbol{s_i^2}) \right) \right), \end{aligned} \tag{5.3}$$

where x and y are the predictions and ground truth for a given batch respectively. L denotes loss. N is the batch size, M is the number of source domains, P is the number of pairwise calculations among distributions, w is a rescaling weight given to the loss of each batch element, γ is the penalty parameter of domain discrepancy.

5.3.3 Experimental Setup

This section outlines various research questions formulated to evaluate the performance of the framework. The primary measure of model performance was F1-score. Secondary measures of model performance were precision, recall (sensitivity), and accuracy [75]. The AUROC (C-statistic) was not employed as an evaluation metric in this study due to unreliability in the context of class imbalance, rendering the metric less indicative of actual performance [96]. Experiments were conducted 10 times, with results presented as the mean along with a two-sided 95% confidence interval (CI).

Upper Bound Results. We developed several hospital readmission LSTM models based on a model that we previously published using EHR data collected from an urban academic hospital [1], To determine the maximal prediction capabilities of readmission risk models, we trained and tested the published LSTM model on each hospital site individually. Two different experiments were conducted. Experiment 1 was conducted by training the model on training samples of the same size (15,966 patients) and experiment 2 was conducted by adopting a data split of 70% for training, 10% for validation, and for 20% testing.

Generalization on Unseen Target Domain. To evaluate the generalization capabilities of the proposed framework, ERR-DGN, experiments were conducted by leveraging knowledge from multiple source domains for training (Sites A-D) and tested on the unseen target domain with the most different data distributions (Site E), encompassing 67,066 patients, of whom 9,553 had a readmission. Several baseline methods were developed for comparison. The baseline LSTM model was trained using data from Sites A-D by concatenating data from these multiple sites, since this baseline model is designed to accept

data from a single source. It was then tested on the unseen target domain, Site E. Additional experiments were conducted where the baseline LSTM model was trained on each source site to assess the performance of each source domain when applied to the unseen target. Of note, a direct comparison to ERR-TDAN is not applicable because ERR-TDAN requires source and target data in training, whereas our objective is to generalize a model and apply it on an unseen hospital that has not been seen in training.

Exploring the Number of Source Sites. We studied the optimal number of source sites needed to enhance predictions on an unseen target domain, Site E, using a greedy approach. We trained the proposed method, ERR-DGN, using all combinations of 2, 3, and 4 sites. We methodically iterated this process, each time selecting the site that most enhanced predictions on the target site, and subsequently evaluating it in conjunction with other sites. The proposed domain Generalization method was evaluated and compared to the baseline LSTM to assess the effectiveness of domain generalization on EHR data collected from multiple hospital systems.

Model Performance Over Time. To investigate the longevity of our proposed model, we conducted experiments by training the model on EHR data from 2016 and subsequently testing it with time gaps of 1, 2, 3, and 4 years into the future. For example, we trained on data from 2016 and tested its performance in 2020 to determine if there would be a decline in accuracy after a span of 4 years. The same process was conducted using training data from 2017 and testing data from 1, 2, and 3 years in the future. Data were not available from all sites before 2016, therefore data from 2016 – 2020 were used for these experiments.

5.4 Data Description and Study Design

This is a retrospective cohort study that used EHR data from five academic health systems, encompassing urban, suburban, and rural areas in Pennsylvania or Maryland, spanning from July 1st, 2010, to December 31st, 2020. All 5 sites are members of the PaTH Network, itself a member of the National Patient-Centered Clinical Research Network (PCORnet). Models were developed using EHR data defined by and standardized according to the PCORnet Common Data Model (CDM) [70]. We extracted data on encounters, demographics, diagnoses, procedures, laboratory tests, medication orders, and vital signs. Eligibility criteria included patients who had at least one hospital discharge and a diabetes diagnosis, based on an *International Classification of Diseases* (ICD) diagnostic code for diabetes (ICD-9: 249.xx or 250.xx or ICD-10: E08.xxx through E13.xxx), a Hemoglobin A1c (HbA1c) level ≥6.5%, or a prescription for a diabetes-specific medication. We excluded encounters for patients aged <18 years, those transferred to another hospital upon discharge, cases of inpatient mortality, instances of gestational diabetes (ICD-9: 648.0x or ICD-10: O24.4x), prediabetes diagnoses (ICD-9: 790.29 or ICD-10: R73.03), or pregnancy (indicated by a positive beta human chorionic gonadotropin lab test within 90 days before or after the encounter) [1]. A total of 25 features extracted from CDM derived were incorporated into the models as presented in Sections 3.2.2 and 4.4.

Patients were categorized based on their hospital readmission status into two groups: those with at least one readmission within 30 days of discharge and those without any 30-day readmissions, as performed in the previously developed readmission models, presented in Chapters 3 and 4. For the readmission group, a random admission-readmission pair was

selected for predictive modeling. Similarly, for patients without readmissions, a single random admission was selected for the same purpose.

To model heterogenous sequential data, we concatenated all encounters for a given patient in a feature vector ordered sequentially by admission date. The primary outcome for modeling was unplanned, all-cause readmission within 30 days after an inpatient discharge, as defined by the Centers for Medicare & Medicaid Services (CMS) [72]. In line with the CMS guidelines, only the first readmission within the 30-day period was considered for modeling. Data were represented in 3-dimensional tensors $p\ x\ e\ x\ f$ that represent f features for each e encounters for p patients. Data preprocessing, managing missing data, accommodating the challenge of handling varying numbers of recordings per encounter, learning embeddings for dimensionality reduction, addressing sparse feature vectors, and data representations were performed as presented in [1]. Added features, such as length of hospital stay (in days) and the number of days since the previous encounter, were aggregated to aid in learning temporal dependencies. [1] We previously conducted experiments to determine how many prior encounters are optimal to predict readmission, finding that 80 or 100 most recent encounters in the prior 2 years yielded the best performance. Thus, we utilized up to 100 most recent encounters in the prior 2 years for each patient. For patients who had less than 100 encounters in the 2-year window, feature vectors were padded with 0's to unify dimensions. Singular Value Decomposition (SVD) was utilized to learn embeddings and reduce dimensionality of encounters to 150 features per encounter, while preserving explained variance of >0.95. This yielded high-dimensional feature vectors, consisting of 15,000 features per patient's sample (150 features per encounter X 100 encounters).

The study protocol was approved by the Johns Hopkins Medicine (JHM) and PennState Health Institutional Review Boards (IRB). The other three participating institutions ceded oversight to the JHM single IRB.

5.5 Results

5.5.1 Data Characteristics

Table 5.1 presents data collected from the 268,754 eligible patients within the five hospital systems, detailing the number of encounters (any EHR interaction: visit, call, medication, or lab order, et al.), number of patients, and class distributions.

Table 5.1. Key site characteristics of 268,754 patients with diabetes by site. Any EHR interaction (ambulatory visit, hospitalization, phone call, order, et al), and 30-day readmission (positive class).

Site	No. Encounters	No. Patients	Positive Class	Area
Site A	4,599,933	54,316	11,442 (21%)	Urban
Site B	2,934,532	37,091	9,014 (24%)	Urban
Site C	12,804,784	90,323	11,172 (12%)	Mixed
Site D	1,363,413	19,958	1,585 (8%)	Suburban
Site E	13,396,308	67,066	9,553 (14%)	Rural

Table 5.2 illustrates key characteristics of the data and shows that each site was statistically different from Site E for every pair-wise comparison except the proportion of females at Site A. *Statistical Significance.* Chi-square and t-test were used to test the statistical significance between variables of the unseen target domain, Site E, and other source sites. T-test was utilized for continuous variables to compare if the means of two groups are statistically different. Chi-square test of independence was employed to test the association between categorical variables [94, 95]. *P*-values obtained show that there is a

significant difference between variables (*p<0.001*) of the unseen target domain, Site E, and source sites, with the exception of sex between Site A and Site E.

Table 5.2. Key characteristics of 268,754 patients with diabetes by site. *p* denotes the *p*-value, μ denotes the mean, and σ denotes the standard deviation.

Variable		**Site A N=54,316**	**Site B N=37,091**	**Site C N=90,323**	**Site D N=19,958**	**Site E N=67,066**
Age (years), (μ & σ)		61.8 ±14.9 *p<0.001*	60.3 ±13.6 *p<0.001*	64.3 ±14.9 *p<0.001*	61.0 ±15.3 *p<0.001*	62.7 ±15.2
Sex (Female)		26,988 (50%) *p=0.24*	18,789 (51%) *p<0.001*	45,844 (51%) *p<0.001*	9,268 (46%) *p<0.001*	33,094 (49%)
Race	White	28,319 (52%)	8,707 (23%)	77,432 (86%)	17,305 (87%)	64,330 (96%)
	Black	18,215 (34%)	17,732 (48%)	8,388 (9%)	1,170 (6%)	1,882 (3%)
	Asian	1,977 (3%)	569 (2%)	630 (<1%)	160 (<1%)	336 (<1%)
	Other	5,805 (11%)	10,083 (27%)	3,873 (4%)	1,323 (7%)	518 (<1%)
	p	*p<0.001*	*p<0.001*	*p<0.001*	*p<0.001*	-
Hispanic (Yes)		2,058 (4%) *p<0.001*	8,495 (23%) *p<0.001*	534 (<1%) *p<0.001*	965 (5%) *p<0.001*	1,833 (3%)
Tobacco	Current	738 (1%)	8,010 (22%)	17,147 (19%)	2,440 (12%)	8,432 (13%)
	Never	2,837 (5%)	12,798 (34%)	49,594 (55%)	10,490 (53%)	18,256 (27%)
	Quit	1,561 (3%)	12,199 (33%)	16,795 (19%)	5,543 (28%)	20,359 (30%)
	Other	49,180 (91%)	4,084 (11%)	6,787 (7%)	1485 (7%)	20,019 (30%)
	p	*p<0.001*	*p<0.001*	*p<0.001*	*p<0.001*	-
No. diagnostic codes, (μ & σ)		2.5 ±2.7 *p<0.001*	19.1 ±11.0 *p<0.001*	15.3 ±7.5 *p<0.001*	18.1 ±13.5 *p<0.001*	10.0 ±5.1
No. procedure codes, (μ & σ)		6.3 ±4.5 *p<0.001*	18.7 ±11.8 *p<0.001*	2.4 ±2.84 *p<0.001*	22.3 ±23.2 *p<0.001*	18.6 ±15.1
Admission duration (days), (μ & σ)		6.8 ±9.5 *p<0.001*	6.2 ±9.8 *p<0.001*	5.5 ±7.1 *p<0.001*	6.4 ±9.7 *p<0.001*	3.5 ±5.3
Prior encounters, (μ & σ)		114.1 ±126.5 *p<0.001*	97.9 ±111.7 *p<0.001*	222.6 ±200.6 *p<0.001*	88.9 ±108.5 *p<0.001*	199.6 ±205.5

Table 5.2. (Continued)

Prior IP, OS, ED encounters, (μ & σ)	3.1 ±6.8 *p<0.001*	3.4 ±7.5 *p<0.001*	7.5 ±15.4 *p<0.001*	3.9 ±6.9 *p<0.001*	4.6 ±9.4
Days since prior encounter, (μ & σ)	11.4 ±54.5 *p<0.001*	25.5 ±92.1 *p<0.001*	12.2 ±54.9 *p<0.001*	25.3 ±90.4 *p<0.001*	12.3 ±51.8
Days since prior IP, OS, ED enc., (μ & σ)	151.8 ±247.4 *p<0.001*	218.8 ±390.2 *p<0.001*	218.8 ±390.2 *p<0.001*	328.5 ±518.9 *p<0.001*	333.7 ±526.3

5.5.2 Upper Bound Results

This section presents results obtained from baseline methods training and testing the models internally at each site. Table 5.3 shows that F-1 scores range from 0.80 to 0.91 in experiment 1 and from 0.80 to 0.92 in experiment 2. The performance differs across sites since data quality of EHR data varied. Certain sites contained fewer missing data, denser features, and less erroneous data than others.

Table 5.3. Performance of the baseline LSTM method based on EHR data collected from five academic hospital systems evaluated using average F-1 score and accuracy metrics. Two experiments were conducted to assess the capabilities of the model on each site and obtain upper bound results. Experiment 1 was conducted by learning from EHR data of 15,966 patients randomly selected from *n* patients and tested on the remaining patients from the same site. Experiment 2 was conducted by adopting a data split of each site to 70% for training, 10% for validation, and 20% for testing.

		Experiment 1		**Experiment 2**	
Site	**N**	**F1-score**	**Accuracy**	**F1-score**	**Accuracy**
Site A	54,316	0.80	0.81	0.80	0.81
Site B	37,091	0.80	0.81	0.81	0.82
Site C	90,323	0.86	0.88	0.88	0.89
Site D	19,958	0.91	0.92	0.91	0.92
Site E	67,066	0.85	0.86	0.87	0.88

5.5.3 Distributional Difference and Selection of Unseen Target Domain

This section presents results obtained from analyzing distributional differences to validate the applicability of applying domain generalization techniques on EHR data and identify a particularly challenging unseen target domain.

To highlight the applicability of applying domain generalization techniques to EHR data sourced from various academic hospital systems, we provide evidence that EHR datasets collected from multiple hospital systems violate conventional machine learning modeling assumptions. This is evident as either the marginal distributions of source and target data diverge, indicating a covariate shift, or the conditional distributions vary due to contextual modifications, suggesting a concept shift. Obtained mean p-values ranges from 0.008 to 0.07, indicating that some distributions are statistically different (p-value < 0.05), and others are marginal. Thus, P_{XY}^{i} for $i \in \{1, \cdots, M\} \neq P_{XY}^{j}$ for $j \in \{1, \cdots, M\}, where\ i\ \neq j$, we can safely reject the null hypothesis, indicating distributions of data collected from five different hospital systems are different.

Fig. 5.2 presents the mean p-values quantifying the distance between pairs of data distributions. As can be seen, distributions are not identical, validating the applicability of leveraging domain generalization on EHR data collected from multiple hospital systems. Table 5.2 presents descriptive statistics of selected variables from varying hospital sites to exemplify main factors that entail distributional difference by testing if there are significant marginal differences between source sites and the unseen target site.

The experiment was leveraged to identify a particularly challenging unseen target site to evaluate the proposed ERR-DGN method by analyzing the data distributions from the five sites, in order to select the site that exhibited the most divergence from the others. Fig.

5.2 shows that site E is the least similar site to others with minimum p-value of 0.008 and maximum p-value of 0.02. Hence, site E was selected as the unseen target domain.

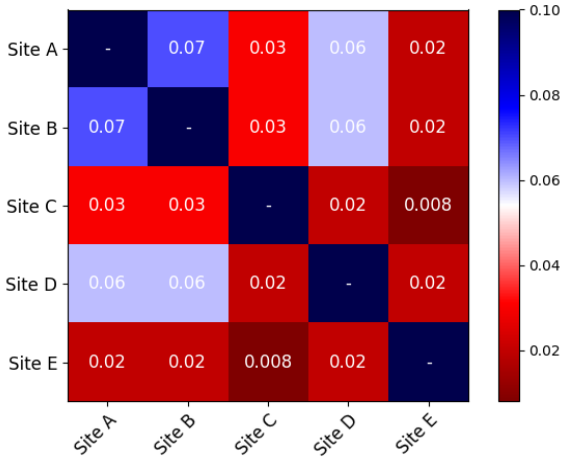

Figure 5.2. Presents the mean p-values quantifying the distance between two data distributions of EHR data collected from five different hospital systems. Results show that Site E exhibits the most divergence from others (p-value range: 0.008 – 0.02). Furthermore, due to small mean p-values, we safely reject the null hypothesis of the KS test indicating that distributions are statistically different.

Distributional difference was further confirmed by applying the baseline LSTM model trained and tested on site C and E respectively representing the sites that exhibit the greatest divergence; F1 score was 0.73, yielding a 15% deterioration in model performance when compared to the baseline model trained and tested on data collected from the same site. Additionally, we applied the baseline LSTM model trained and tested on site A and site B respectively, exhibiting the least divergence in F1 score (0.71), a 9% deterioration relative to the baseline model. Performances were significantly affected since the model struggled to generalize effectively and converge due to training data drawn from different

distributions. To overcome this limitation and the absence of a comprehensive model capable of generalizing on multiple source domains for hospital readmission on EHR data in literature, we adapted the concept of DAN, specifically tailored for domain generalization on several source domains, aiming to enhance prediction on an unseen target hospital.

5.5.4 Generalization on Unseen Target Domain

This section presents results obtained to assess the generalization capabilities of the proposed framework, ERR-DGN. Results obtained aided in validating the assumption that training on EHR data from one source domain and testing it on target data from a different hospital system violates conventional machine learning modeling assumptions.

Table 5.4 illustrates the baseline LSTM model when trained on combined data from Sites A-D and tested on the unseen Site E. This baseline model yielded 0.73 in F1-score, demonstrating no further improvement to a baseline model trained on a single source domain (Site A). The F1 score varied between 0.61 (indicating suboptimal performance when trained on Site B) and 0.73 (indicating satisfactory performance when trained on Site A). The proposed framework, ERR-DGN, yielded a 6% increase in the F1-score compared to the baseline model trained on Sites A-D, and on average, a 10% increase when compared to the baseline LSTM model trained on a single source. The confidence intervals were very small (<0.01), indicating a high degree of precision around the means.

Table 5.4. Performance of the proposed method, ERR-DGN and baselines. The average F1-Scores and their corresponding two-sided 95% confidence interval on 10 experiments.

Model	Source	Target (unseen)	Precision	Recall	F1	Specificity	Accuracy
Baseline LSTM	Site B	Site E	0.75	0.52	0.61 ±0.009	0.53	0.52
	Site C	Site E	0.75	0.67	0.70 ±0.005	0.74	0.67
	Site D	Site E	0.76	0.70	0.72 ±0.006	0.77	0.70
	Site A	Site E	0.76	0.70	0.73 ±0.002	0.78	0.70
	Sites A-D	Site E	0.75	0.71	0.73 ±0.007	0.78	0.70
ERR-DGN	Sites A-D	Site E	**0.79**	**0.80**	**0.79** ±0.006	**0.94**	**0.80**

5.5.5 Exploring Number of Source Sites

This section presents results obtained from extensive experiments conducted using a greedy approach to determine the optimal number of source sites needed to enhance predictions on an unseen target domain, Site E. Table 5 illustrates this greedy approach taken to find the number of sites that are most effective on Site E. Results from ERR-DGN for 1 site are not displayed, since ERR-DGN is designed to generalize over data from multiple sources. In contrast, the baseline LSTM model is constructed to handle data exclusively from a single source.

Table 5.5 shows that performance plateaued at three source sites, and adding more source sites was not found to increase performance. By testing and comparing different numbers of sites in combination, we observed a 3% boost in F1-score compared to when using two sites and a 6% improvement in comparison to using just one site with the baseline approach.

Table 5.5. Presents a comparative analysis using a greedy approach to find the optimal number of source sites needed to enhance predictions on unseen target domain. Presented results are evaluated using F1-Score.

# Sites	Source	Target (unseen)	ERR-DGN	Baseline LSTM
1	Site B	Site E	-	0.61
	Site C		-	0.70
	Site D		-	0.72
	Site A		**-**	0.73
2	Site A + Site B		0.72	0.69
	Site A + Site C		0.75	0.72
	Site A + Site D		0.76	0.73
3	Site A + Site D + Site B		0.76	0.72
	Site A + Site D + Site C		0.79	0.73
4	**Sites A-D**		0.79	0.73

5.5.6 Model Performance Over Time

This section illustrates the longevity of the proposed model to avoid a decline in model performance over time. Fig. 5.3 illustrates a gradual decline in F1-score over time. The model's performance remained relatively consistent when tested for predictions 1 and 2 years into the future. However, a noticeable dip in the F1-score was observed with a 4-year gap between training and testing data. On average, there was a 1% decline in F1-score when predictions were made 3 years post-training and a 4% drop after 4 years.

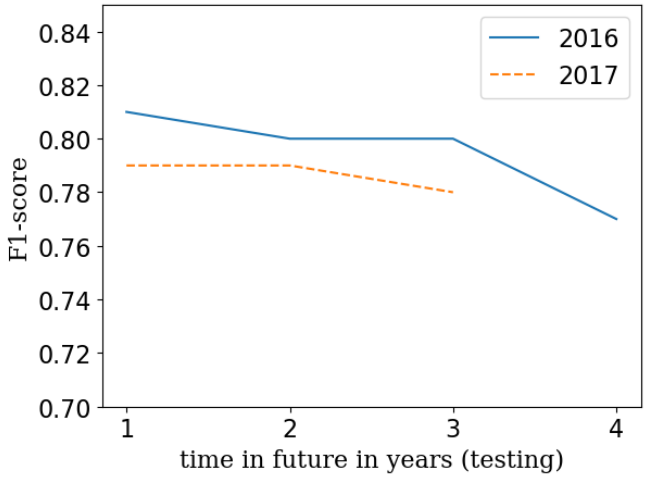

Figure 5.3. The model's longevity and its ability on sustain optimal performance are showcased here. We developed two distinct models using data acquired in 2016 and 2017, aiming to forecast outcomes with a time gap ranging from 1 to 4 years. The findings suggest that for preserving peak efficiency, it might be advantageous to retrain the proposed framework approximately every three years.

5.6 Discussion

We propose a framework, ERR-DGN, that is capable of generalizing on EHR data from distinct hospital systems with different distributions. ERR-DGN is designed to enhance hospital readmission prediction when applied on an unseen target data relative to conventional approaches. Its strength lies in its ability to capture and internalize patterns and characteristics consistent across multiple source domains. ERR-DGN yielded a 6% increase in F1-score compared to the baseline. Baseline models struggled to generalize effectively when tested on an unseen target data, likely because concatenating data from varied different sources introduced bias. Several factors could lead to a distributional difference, including but not limited to sociodemographic, diagnosis and procedure codes, and capture of such codes. In addition, we studied the number of source sites needed to

enhance predictions on an unseen target domain, finding that three source hospital systems from varied area types were found to be sufficient to predict readmission on an unseen domain. Lastly, we examined model performance over time, showing that performance was stable for 3 years then declined in year 4. This suggests that periodic retraining at least every 3 years may mitigate model degradation over time.

This study is a substantial extension of our paper published at AIME-2023 [6], as it includes the following contributions:

1) We enhanced the framework proposed in AIME-2023 [3] to learn the model on EHR data observed at multiple hospital systems and subsequently apply the rehospitalization prediction model to a novel site, rather than leveraging historic data from a single source and a target domain to enhance readmission risk predictions on the target domain. In other words, the proposed method ERR-DGN has no access to the data of the target hospital in training, unlike the framework in [6] that requires both source and target data in training.
2) We broadened the scope by incorporating three additional hospital systems, bringing the total to five sites encompassing 268,754 patients.
3) We validated the applicability of utilizing domain generalization techniques on EHR data collected from five different hospital systems by comparing the underlying distributions to check whether source and target distributions are identical.
4) In a forward search-based optimization we analyzed benefits of learning from multiple hospital sites for deploying a readmission risk prediction model to a previously unencountered site.

5) We determined the duration of the optimal performance of the model.

In present study, we explored an often-overlooked feature of datasets collected from multiple sites: differences among sites. Indeed, we found that distributions of the variables and many characteristics of the samples drawn from each site were statistically significantly different. Furthermore, we confirmed that a conventional LSTM model trained on 4 sites and tested on the unseen fifth site performed rather poorly relative to models trained and tested internally on each site. It is impractical for most hospitals to train existing models on their own data, much less to develop new models. Existing methods lack the ability to generalize across varied hospital system data distributions without access to target training data. Hence, there is a need for modeling approaches that maximize performance on unseen domains like ERR-DGN described here. ERR-DGN is the first end-to-end domain generalization framework for hospital readmission.

The ERR-DGN approach performs well relative to most previously published 30-day readmission risk models, which reported F1-scores of 0.386 to 0.58 [97-99] and 0.812 [25]. Of note, these previously reported models were all trained and tested internally on samples drawn from the same source, unlike the ERR-DGN which was tested on data from an unseen source. In addition, most papers on readmission risk models do not report F1-score, so opportunities for comparison are limited [101]. Transfer learning has been explored for hospital readmission to address data scarcity using relevant source datasets.

The key strengths of this study include the use of a relatively large dataset, the incorporation of multiple sites with varied characteristics collected over 10 years, and the use of data curated according to the PCORnet CDM, which standardized data across sites. Some limitations should be acknowledged as well. Data were drawn from sites in 2 states

(PA and MD) and may not be nationally representative. Although data curated by PaTH in the CDM format is a strength in terms of data quality, our results might not generalize to institutions outside of PCORnet. To mitigate this risk, only variables that are commonly and routinely collected in EHRs were used for modeling. In addition, a deployment challenge for ERR-DGN is that it requires training data from at least two source domains. In a subsequent study, we aim to assess the feasibility of applying the proposed method prospectively as well as to further explore how Site E is different and what characteristics need to be considered to optimize performance across sites.

We proposed a framework to be applied on sites where historical EHR data is not available or the development of a new readmission model on a certain site might be infeasible. We conducted extensive experiments to evaluate our hypothesis and compared to baseline methods. The proposed method outperformed baseline methods and results show that ERR-DGN can be an effective tool to enhance hospital readmission on an unseen target hospital. We further supplemented our findings by exploring the number of hospital sites needed and longevity of the model. This approach might be propagated to other scenarios where training occurs on relevant EHR data from different distributions.

5.7 Acknowledgments

This research was supported by the National Health Institute (NIH) under grant number R01DK122073.

CHAPTER 6

CONCLUSION

In closing, the studies discussed introduced several transfer learning approaches to mitigate challenges faced with real-world, field-recorded data in different high-impact domain applications, including healthcare and power systems. Data drift, labels scarcity, and data quality are one of the primary reasons that impact the performance of traditional machine learning and deep learning methods. Hence, this dissertation introduces methods that can mitigate the labor-intensive manual labeling to address labels scarcity and data drift to aid stabilize electrical grids. Additionally, we proposed frameworks designed for hospital readmission that address data quality issues, learning from different distributions to enhance predictions in the target domain, and generalize to unseen target domains, aiming to improve patient care and reduce the significant costs associated with 30-day readmissions.

We examined the hypothesis that it is feasible to enhance hospital readmission risk predictions on EHR data using data collected from a related source domain. The proposed model yielded a 5% increase in F-1 score when compared to baselines. Furthermore, we conducted studies to address concerns and research questions related to this domain application to avoid performance degradation due to data drift over time. We proposed and validated a framework that establishes spatial knowledge transfer based on a temporal deep adaptation network tailored for hospital readmission predictions of the target task using higher quality data from a related source domain under different distributions by matching

the mean embeddings to reduce domain discrepancy. This approach is the first end-to-end transfer learning framework based on domain adaptation for hospital readmission. Conducted experiments show that the proposed early readmission risk temporal domain adaptation network method can enhance hospital readmission and improve patients' healthcare.

We further proposed a framework, ERR-DGN, that is capable of generalizing on EHR data from distinct hospital systems with different distributions to enhance hospital readmission prediction when applied on an unseen target data relative to conventional approaches. Its strength lies in its ability to capture and internalize patterns and characteristics consistent across multiple source domains. ERR-DGN yielded a 6% increase in F1-score compared to the baseline. Baseline models struggled to generalize effectively when tested on unseen target data, likely because concatenating data from varied different sources introduced bias. In addition, we studied the number of source sites needed to enhance predictions on an unseen target domain, finding that three source hospital systems from varied area types were found to be sufficient to predict readmission on an unseen domain. Lastly, we examined model performance over time, showing that future performance was stable for 3 years then declined in year 4. This suggests that periodic retraining at least every 3 years may mitigate model degradation over time.

Proposed transfer learning in conjunction with a semi-supervised detectors that detect events based on minimal, well-defined labeled data instances from a related source task to mitigate the labor-intensive manual labeling efforts. This study shows that the proposed transfer learning method yields a substantial increase in AUROC compared with alternative state-of-the-art baselines (fully supervised, semi-supervised, and unsupervised).

Experiments conducted show that this method is more feasible than alternative baselines when conventional machine learning modeling assumptions are violated and outperforms the baselines when reusing labeled data instances from one power system to detect events from another. Furthermore, this method can detect events based on a small amount of transferred relevant labeled data from another power system.

BIBLIOGRAPHY

1. Abdel Hai, A., Weiner, M.G., Paranjape, A.P., Livshits, A., Brown, J.R., Obradovic, Z., & Rubin, D.J. (2022). "Deep Learning vs Traditional Models for Predicting Hospital Readmission among Patients with Diabetes." In *Proceedings of the AMIA 2022 Annual Symposium,* Washington, DC.

2. Abdel Hai, A., et al. (2021). "Transfer Learning for Event Detection from PMU Measurements with Scarce Labels." In *IEEE Access*, 9, 127420-127432.

3. Abdel Hai, A., Mohamed, T., Pavlovski, M., Kezunovic, M., & Obradovic, Z. (2022). "Transfer Learning on Phasor Measurement Data from a Power System to Detect Events in Another System." In *Proceedings of the 21st IEEE International Conference on Machine Learning and Applications (ICMLA)*, Nassau, Bahamas, 1567-1572.

4. Dokic, T., et al. (2022). "Machine Learning Using a Simple Feature for Detecting Multiple Types of Events from PMU Data." In *2022 International Conference on Smart Grid Synchronized Measurements and Analytics (SGSMA)*, Split, Croatia, 1-6.

5. Otudi, H., Mohamed, T., Hu, Y., Kezunovic, M., & Obradovic, Z. (2023). "Training Machine Learning Models with Simulated Data for Improved Line Fault Events Classification from 3-Phase PMU Field Recordings." *In Proceedings of the 56th IEEE Hawaii International Conference on System Science (HICSS)*, Maui, 2641-2650. https://hdl.handle.net/10125/102957

6. Abdel Hai, A., Weiner, M.G., Livshits, A., Brown, J.R., Paranjape, A.P., Obradovic, Z., & Rubin, D.J. (2023). "Spatial Knowledge Transfer with Deep Adaptation Network for Predicting Hospital Readmission." In *Proceedings of the 21st International Conference on Artificial Intelligence in Medicine*, Portoroz, Slovenia.

7. NASPI Data Network Management Task Team. (2020). "NASPI 2020 Survey of Industry Best Practices for Archiving Synchronized Measurements." *North American Synchrophasor Initiative Technical Report*. NASPI-2020-TR-024.

8. North American SynchroPhasor Initiative. (2019). "Data Mining Techniques and Tools for Synchrophasor Data." *NASPI, Technical Report.* NASPI-2018-TT-007.

9. Chandola, V., Banerjee, A., & Kumar, V. (2009). "Anomaly Detection: A survey." *ACM Computing Surveys*, 41(3), 1-72.

10. Ramaswamy, S., Rastogi, R., & Shim, K. (2000). "Efficient Algorithms for Mining Outliers from Large Data Sets." *ACM SIGMOD Record,* 29(2), 427-438.

11. Vercruyssen, V., Meert, W., & Davis, J. (2020). "Transfer Learning for Anomaly Detection through Localized and Unsupervised Instance Selection." In *Proceedings of the AAAI Conference on Artificial Intelligence*, 6054-6061.

12. Xie, L., Chen, Y., & Kumar, P. R. (2014). "Dimensionality Reduction of Synchrophasor Data for Early Event Detection: Linearized Analysis. *IEEE Transactions on Power Systems*, 29(6), 2784-2794.

13. Rafferty, M., Liu, X., Laverty, D. M., & McLoone, S. (2016). "Real-Time Multiple Event Detection and Classification Using Moving Window PCA." *IEEE Transactions on Smart Grid,* 7(5), 2537-2548.

14. Dahal, O. P., Brahma, S. M., & Cao, H. (2014). "Comprehensive Clustering of Disturbance Events Recorded by Phasor Measurement Units." *IEEE Transactions on Power Delivery*, 29(3), 1390-1397.

15. Biswal, M., Brahma, S. M., & Cao, H. (2016). "Supervisory Protection and Automated Event Diagnosis using PMU Data." *IEEE Transactions on Power Delivery*, 31(4), 1855-1863.

16. Biswal, M., Hao, Y., Chen, P., Brahma, S., Cao, H., & De Leon, P. (2016). "Signal Features for Classification of Power System Disturbances using PMU Data." In *Proceedings of the Power Systems Computation Conference (PSCC)*, 1-7.

17. Brahma, S., Kavasseri, R., Cao, H., Chaudhuri, N. R., Alexopoulos, T., & Cui, Y. (2017). "Real-Time Identification of Dynamic Events in Power Systems using PMU Data, and Potential Applications - Models, Promises, and Challenges." *IEEE Transactions on Power Delivery*, 32(1), 294-301.

18. Kim, D. -I., Chun, T. Y., Yoon, S. H., Lee, G., & Shin, Y. J. (2017). "Wavelet-Based Event Detection Method using PMU Data." *IEEE Transactions on Smart Grid,* 8(3), 1154-1162.

19. Wang, S., Dehghanian, P., & Li, L. (2020). "Power Grid Online Surveillance through PMU-Embedded Convolutional Neural Networks. *IEEE Transactions on Industry Applications*, 56(2), 1146-1155.

20. Khan, M., Ashton, P. M., Li, M., Taylor, G. A., Pisica, I., & Liu, J. (2015). "Parallel Detrended Fluctuation Analysis for Fast Event Detection on Massive PMU Data." *IEEE Transactions on Smart Grid*, 6(1), 360-368.

21. Cui, M., Wang, J., Tan, J., Florita, A. R., & Zhang, Y. (2019). "A Novel Event Detection Method using PMU Data with High Precision." *IEEE Transactions on Power Systems*, 34(1), 454-466.

22. Yadav, R., Pradhan, A. K., & Kamwa, I. (2019). "Real-Time Multiple Event Detection and Classification in Power System using Signal Energy Transformations*." IEEE Transactions on Industrial Informatics*, 15(3), 1521-1531.

23. Shahsavari, A., Farajollahi, M., Stewart, E. M., Cortez, E., & Mohsenian-Rad, H. (2019). "Situational Awareness in Distribution Grid using Micro-PMU Data: A Machine Learning Approach." *IEEE Transactions on Smart Grid*, 10(6), 6167-6177.

24. Jafarzadeh, S., Moarref, N., Yaslan, Y., & Genc, V. M. I. (2019). "A CNN-Based Post-Contingency Transient Stability Prediction Using Transfer Learning." In *Proceedings of the 11th International Conference on Electrical and Electronics Engineering (ELECO)*, Bursa, Turkey, 156-160.

25. Mrabet, Z. E., Selvaraj, D. F., & Ranganathan, P. (2019). "Adaptive Hoeffding Tree with Transfer Learning for Streaming Synchrophasor Data Sets." *In Proceedings of the IEEE International Conference on Big Data*, Los Angeles, CA, USA, 5697-5704.

26. Zhang, Y., Wang, X., Luo, Y., Xu, Y., He, J., & Wu, G. (2020). "A CNN Based Transfer Learning Method for High Impedance Fault Detection." In *Proceedings of the IEEE Power and Energy Society General Meeting (PESGM)*, Montreal, QC, Canada, 1-5.

27. Van Haaren, J., Kolobov, A., & Davis, J. (2015). "TODTLER: Two-Order-Deep Transfer Learning." In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, 3007-3015.

28. Kouw, W. M., & Loog, M. (2018). "An Introduction to Domain Adaptation and Transfer Learning." *arXiv preprint arXiv:1812.11806.* http://arxiv.org/abs/1812.11806

29. Sathya, R., & Abraham, A. (2013). "Comparison of Supervised and Unsupervised Learning Algorithms for Pattern Classification." *International Journal of Advanced Research in Artificial Intelligence*, 2(2), 34-38.

30. Bandaragoda, T. R., Ting, K. M., Albrecht, D., Liu, F. T., Zhu, Y., & Wells, J. R. (2018). "Isolation-Based Anomaly Detection using Nearest-Neighbor Ensembles." Computational Intelligence, 34(4), 968-998.

31. Scikit-Learn. Machine Learning in Python. https://scikit-learn.org/stable/

32. Zhu, X., & Goldberg, A. (2009). "A Concise Introduction to Multiagent Systems and Distributed Artificial Intelligence." *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 3(1), 1-130.

33. Vercruyssen, V., Meert, W., Verbruggen, G., Maes, K., Baumer, R., & Davis, J. (2018). "Semi-Supervised Anomaly Detection with an Application to Water Analytics." In *Proceedings of the IEEE International Conference on Data Mining* (ICDM), Singapore, 527-536.

34. Emmott, A. F., Das, S., Dietterich, T., Fern, A., & Wong, W.-K. (2013). "Systematic Construction of Anomaly Detection Benchmarks from Real Data." In *Proceedings of the ACM SIGKDD Workshop Outlier Detection Description*, 16-21.

35. Powers, D. M. W. (2008). "Evaluation: From Precision, Recall and F-measure to ROC, Informedness, Markedness and Correlation." *Journal of Machine Learning Technologies,* 2, 1-27.

36. Baldi, P., Brunak, S., Chauvin, Y., Andersen, C. A. F., & Nielsen, H. (2000). "Assessing the Accuracy of Prediction Algorithms for Classification: An Overview." *Bioinformatics,* 16(5), 412-424.

37. Pan, S. J., Tsang, I. W., Kwok, J. T., & Yang, Q. (2011). "Domain Adaptation Via Transfer Component Analysis." *IEEE Transactions on Neural Networks,* 22(2), 199-210.

38. Hodges, J. L. (1958). "The Significance Probability of the Smirnov Two-Sample Test." *Arkiv Matematik,* 3(5), 469-486.

39. Shi, J., Yamashita, K., Yu, N. (2022). "Power System Event Identification with Transfer Learning Using Large-scale Real-world Synchrophasor Data in the United States." *IEEE ISGT.*

40. Wang, W., Huang, Y., Wang, Y., & Wang, L. "IEEE Conference on Computer Vision and Pattern Recognition Workshops." 14, 490-497.

41. Cheng, Z., Hu, Y., Obradovic, Z., & Kezunovic, M. (2022). "Using Synchrophasor Status Word as Data Quality Indicator: What to Expect in the Field?" In *Proceedings of the IEEE International Conference on Smart Grid Synchronized Measurements and Analytics (SGSMA)*, Split, Croatia.

42. Benbassat, J., & Taragin, M. (2000). "Hospital Readmissions as a Measure of Quality of Health Care: Advantages and Limitations." *Archives of Internal Medicine*, 160, 1074-1081.

43. Rubin, D. J. (2015). "Hospital Readmission of Patients with Diabetes*." Current Diabetes Reports,* 15, 1-9.

44. Ostling, S., Wyckoff, J., Ciarkowski, S. L., Pai, C.-W., Choe, H. M., Bahl, V., et al. (2017). "The Relationship Between Diabetes Mellitus and 30-day Readmission Rates." *Clinical Diabetes and Endocrinology*, 3, 3.

45. Enomoto, L. M., Shrestha, D. P., Rosenthal, M. B., Hollenbeak, C. S., & Gabbay, R. A. (2017). "Risk Factors Associated with 30-day Readmission And Length Of Stay In Patients with Type 2 Diabetes." *Journal of Diabetes Complications,* 31, 122-127.

46. AHRQ, Healthcare Cost and Utilization Project (hcup) National Inpatient Sample (nis). (2018).

47. ADA. (2018). "Economic Costs of Diabetes in the U.S." In *2017. Diabetes Care*, 41, 917-928.

48. Rubin, D. J., & Shah, A. A. (2021). "Predicting and Preventing Acute Care Re-Utilization by Patients with Diabetes." *Current Diabetes Reports,* 21.

49. Rubin, D. J., Handorf, E. A., Golden, S. H., Nelson, D. B., McDonnell, M. E., & Zhao, H. (2016). "Development and Validation of a Novel Tool to Predict Hospital Readmission Risk Among Patients with Diabetes." *Endocrine Practice*, 22, 1204-1215.

50. Rubin, D. J., Recco, D., Turchin, A., Zhao, H., & Golden, S. H. (2018). "External Validation of the Diabetes Early Re-admission Risk Indicator (derri())." *Endocrine Practice*, 24, 527-541.

51. Alamer, A. A., Patanwala, A. E., Aldayyen, A. M., & Fazel, M. T. (2019). "Validation and Comparison of Two 30-day Readmission Prediction Models in Patients with Diabetes." *Endocrine Practice,* 25, 1151-1157.

52. Karunakaran, A., Zhao, H., & Rubin, D. J. (2018). "Predischarge and Postdischarge Risk Factors for Hospital Readmission Among Patients with Diabetes." *Medical Care*, 56, 634-642.

53. Alloghani, M., Aljaaf, A., Hussain, A., Baker, T., Mustafina, J., Al-Jumeily, D., et al. (2019). "Implementation of Machine Learning Algorithms to Create Diabetic Patient Re-Admission Profiles." *BMC Medical Informatics and Decision Making*, 19, 253.

54. Alturki, L., Aloraini, K., Aldughayshim, A., & Albahli, S. (2019). "Predictors of Readmissions and Length of Stay for Diabetes Related Patients." In *2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA).*

55. Bhatt, V., Chakravorty, T., & Chakraborty, S. (2022). "Re-admission Rate Prediction of Diabetes Patient: Health Analytics-Based Approach." *In Springer Singapore*, 743-754.

56. Dinh Phu Cuong, L., & Wang, D. (2021). "A Comparison of Machine Learning Methods to Predict Hospital Readmission of Diabetic Patient." *Studies of Applied Economics,* 39.

57. Cui, S., Wang, D., Wang, Y., Yu, P. W., & Jin, Y. (2018). "An Improved Support Vector Machine-Based Diabetic Readmission Prediction." *Computer Methods and Programs in Biomedicine,* 166, 123-135.

58. Grampurohit, S. (2021). “Diabetes Patients Hospital Re-Admission Prediction Using Machine Learning Algorithms”. In *Springer Singapore*, 485-497.

59. Neto, C., Senra, F., Leite, J., Rei, N., Rodrigues, R., Ferreira, D., et al. (2021). “Different Scenarios for the Prediction of Hospital Readmission of Diabetic Patients.” *J Med Syst*, 45(11).

60. Ramírez, J. C., & Herrera, D. (2019). “Prediction of Diabetic Patient Readmission Using Machine Learning. In *Springer International Publishing*, 78-88.

61. Shang, Y., Jiang, K., Wang, L., Zhang, Z., Zhou, S., Liu, Y., et al. (2021). “The 30-Days Hospital Readmission Risk in Diabetic Patients: Predictive Modeling with Machine Learning Classifiers.” *BMC Medical Informatics and Decision Making*, 21.

62. Shibly, M. M. A., Tisha, T. A., & Mazumder, M. M. I. (2021). “Predicting Early Readmission of Diabetic Patients: Toward Interpretable Models. In *International Conference on Communication, Computing and Electronics Systems: Proceedings of ICCCES 2020.* Springer.

63. Shih, D.-H., Huang, F.-C., Weng, C.-L., Shih, P.-Y., & Yen, D. C. (2020). “Thirty-Day Re-Hospitalization Rate Prediction of Diabetic Patients.” *Journal of Internet Technology,* 21, 2065-2074.

64. Hammoudeh, A., Al-Naymat, G., Ghannam, I., & Obied, N. (2018). “Predicting Hospital Readmission among Diabetics Using Deep Learning. *Procedia Computer Science*, 141, 484-489.

65. Hu, P., Li, S., Huang, Y., & Hu, L. (2019). “Predicting Hospital Readmission of Diabetics Using Deep Forest.” In *2019 IEEE International Conference on Healthcare Informatics (ICHI)*.

66. Reddy, S. S., Sethi, N., & Rajender, R. (2020). “Evaluation of Deep Belief Network to Predict Hospital Readmission of Diabetic Patients.” In *2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA),* IEEE.

67. Sarthak, Shukla, S., & Prakash Tripathi, S. (2021). “Embpred30: Assessing 30-days Readmission for Diabetic Patients Using Categorical Embeddings.” In *Springer Singapore*, 81-90.

68. Welchowski, T., & Schmid, M. (2016). “A Framework for Parameter Estimation and Model Selection in Kernel Deep Stacking Networks.” *Artificial Intelligence in Medicine*, 70, 31-40.

69. Strack, B., DeShazo, J. P., Gennings, C., Olmo, J. L., Ventura, S., Cios, K. J., et al. (2014). “Impact of HbA1c Measurement on Hospital Readmission Rates: Analysis of 70,000 Clinical Database Patient Records.” *BioMed Research International,* 781670.

70. Elixhauser, A. (2014). "Clinical Classifications Software (CCS): Agency for Healthcare Research and Quality." http://www.hcup-us.ahrq.gov/toolssoftware/ccs/ccs.jsp Accessed 12/27/2021.

71. Elixhauser, A., Steiner, C., Harris, D. R., & Coffey, R. M. (1998). "Comorbidity Measures for Use with Administrative Data." *Medical Care*, 36, 8-27.

72. Centers for Medicare & Medicaid Services (CMS). (2016). All-Cause Hospital-Wide Measure Updates and Specifications Report. Hospital-Level 30-Day Risk-Standardized Readmission Measure – Version 5.0.

73. Williams, R. J., & Zipser, D. (1989). "A Learning Algorithm for Continually Running Fully Recurrent Neural Networks." *Neural Computation*, 1, 270-280.

74. Cho, K., Van Merrienboer, B., Gulcehre, A., Bahdanau, D., Bougares, F., Schwenk, H., et al. (2014). "Learning Phrase Representations Using RNN Encoder-Decoder for Statistical Machine Translation." In *EMNLP*.

75. Powers, D. M. W. (2011). "Evaluation: From Precision, Recall and f-measure to ROC., Informedness, Markedness & Correlation." *Journal of Machine Learning Technologies*, 2, 37-63.

76. Povalej Brzan, P., Obradovic, Z., & Stiglic, G. (2017). "Contribution of Temporal Data to Predictive Performance in 30-Day Readmission of Morbidly Obese Patients." *PeerJ,* 5, e3230.

77. Zhao, H., Tanner, S., Golden, S. H., Fisher, S. G., & Rubin, D. J. (2020). "Common Sampling and Modeling Approaches to Analyzing Readmission Risk that Ignore Clustering Produce Misleading Results." *BMC Medical Research Methodology*, 20, 281.

78. McIlvennan, C. K., Eapen, Z. J., & Allen, L. A. (2015). "Hospital Readmissions Reduction Program." *Circulation,* 131(20), 1796-1803.

79. Long, M., Cao, Y., Wang, J., & Jordan, M. (2015). "Learning Transferable Features with Deep Adaptation Networks." In *Proceedings of the 32nd International Conference on Machine Learning*, Proceedings of Machine Learning Research, 37, 97-105.

80. Helm, J. E., Alaeddini, A., Stauffer, J. M., Bretthauer, K. M., & Skolarus, T. A. (2016). "Reducing Hospital Readmissions by Integrating Empirical Prediction with Resource Optimization. *Production and Operations Management,* 25(2), 233-257.

81. Desautels, T., Das, R., Calvert, J., Trivedi, M., Summers, C., Wales, D. J., & Ercole, A. (2017). "Prediction of Early Unplanned Intensive Care Unit Readmission in a UK Tertiary Care Hospital: A Cross-Sectional Machine Learning Approach." *BMJ Open*, 7(9), e017199.

82. Gupta, P., Malhotra, P., Narwariya, J., et al. (2020). “Transfer Learning for Clinical Time Series Analysis Using Deep Neural Networks.” *Journal of Healthcare Informatics Research,* 4(2) 112-137.

83. Wang, M., & Deng, W. (2018). “Deep Visual Domain Adaptation: A Survey.” *Neurocomputing*, 312, 135-153.

84. Deng, W., Zheng, L., Ye, Q., Kang, G., Yang, Y., & Jiao, J. (2018). “Image-Image Domain Adaptation with Preserved Self-Similarity and Domain-Dissimilarity for Person Re-identification.” In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),* 994-1003.

85. Chen, Y., Li, W., Sakaridis, C., Dai, D., & Van Gool, L. (2018). “Domain Adaptive Faster R-CNN for Object Detection in the Wild.” In *Proceedings 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),* 994-1003.

86. Hosseini-Asl, E., Keynton, R., & El-Baz, A. (2016). “Alzheimer’s Disease Diagnostics by Adaptation of 3D Convolutional Network.” In *Proceedings IEEE International Conference on Image Processing (ICIP),* 126-130.

87. Hochreiter, S., & Schmidhuber, J. (1997). “Long Short-Term Memory.” *Neural Computation*, 9(8), 1735-1780.

88. Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., & Smola, A. (2012). “A Kernel Two-Sample Test.” *Journal of Machine Learning Research*, 13, 723–773.

89. Wang, J., et al. (2023). “Generalizing to Unseen Domains: A Survey on Domain Generalization.” *IEEE Transactions on Knowledge and Data Engineering,* 35(8), 8052-8072.

90. Pan, S., & Yang, Q. (2010). “A Survey on Transfer Learning”. *IEEE Transactions on Knowledge and Data Engineering,* 22, 1345–1359.

91. Weiss, K., Khoshgoftaar, T. M., & Wang, D. (2016). “A Survey of Transfer Learning.” *Journal of Big Data*, 3(1), 1–40.

92. Caruana, R. (1997). “Multitask Learning.” *Machine Learning*, 28(1), 41–75.

93. Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., & Smola, A. (2012). “A Kernel Two-Sample Test.” *Journal of Machine Learning Research*, 13, 723–773.

94. Snijders, T. A. B., & Borgatti, S. P. (1999). “Non-Parametric Standard Errors and Tests for Network Statistics.” *Connections*, 22, 61–70.

95. Borgatti, S. P., Everett, M. G., & Freeman, L. C. (2002). "UCINET for Windows: Software for Social Network Analysis." *Harvard, MA: Analytic Technologies*.

96. Saito, T., & Rehmsmeier, M. (2015). "The Precision-Recall Plot Is More Informative Than the ROC Plot when Evaluating Binary Classifiers on Imbalanced Datasets." *PLoS ONE*, 10(3), e0118432.

97. Ashfaq, A., Sant'Anna, A., Lingman, M., & Nowaczyk, S. (2019). "Readmission Prediction Using Deep Learning on Electronic Health Records." *Journal of Biomedical Informatics*, 97, 103256.

98. Baig, M. M., et al. (2019). "Machine Learning-based Risk of Hospital Readmissions: Predicting Acute Readmissions within 30 Days of Discharge." In *Proceedings 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2178-2181.

99. Mahajan, S. M., & Ghani, R. (2019). "Using Ensemble Machine Learning Methods for Predicting Risk of Readmission for Heart Failure." *Studies in Health Technology and Informatics,* 264, 243-247.

100. Lay, J., Alfonso-Lizarazo, E., Augusto, V., Bongue, B., Masmoudi, M., Xie, X., & Célarier, T. (2022). "Prediction of Hospital Readmission of Multimorbid Patients Using Machine Learning Models." *PLoS ONE*, 17(12), e0279433.

101. Zhou, H., Della, P. R., Roberts, P., Goh, L., & Dhaliwal, S. S. (2016). "Utility of Models to Predict 28-Day Or 30-Day Unplanned Hospital Readmissions: An Updated Systematic Review." *BMJ Open,* 6(6), e011060.

102. Abdel Hai, A., et al. (In review). "Domain Generalization for Enhanced Predictions of Hospital Readmission on Unseen Domains." *Artificial Intelligence in Medicine.*