Length of Stay prediction for Hospital Management using Domain Adaptation

Lyse Naomi Wamba Momo *[a], Nyalleng Moorosi[b], Elaine O. Nsoesie[c], Frank Rademakers[d], and Bart De Moor[a]

[a]KU Leuven, Department of Electrical Engineering (ESAT), Stadius Center for Dynamical Systems, Signal Processing and Data Analytics, Leuven, Belgium

[b]Google Research

[c]Department of Global Health, Boston University School of Public Health

[d]Division of Cardiovascular Imaging and Dynamics, Department of Cardiovascular Sciences, KU Leuven, Leuven, Belgium

February 28, 2023

*Corresponding author: lysenaomi.wambamomo@kuleuven.be

Address: ESAT-STADIUS, Stadius Centre for Dynamical Systems, Signal Processing and Data Analytics, Kasteelpark Arenberg 10, 3001 Leuven, Belgium.

Abstract

Inpatient length of stay (LoS) is an important managerial metric which if known in advance can be used to efficiently plan admissions, allocate resources and improve care. Using historical patient data and machine learning techniques, LoS prediction models can be developed. Ethically, these models can not be used for patient discharge in lieu of unit heads but are of utmost necessity for hospital management systems in charge of effective hospital planning. Therefore, the design of the prediction system should be adapted to work in a true hospital setting. In this study, we predict early hospital LoS at the granular level of admission units by applying domain adaptation to leverage information learned from a potential source domain. Time-varying data from 110,079 and 60,492 patient stays to 8 and 9 intensive care units were respectively extracted from eICU-CRD and MIMIC-IV. These were fed into a Long-Short Term Memory and a Fully connected network to train a source domain model, the weights of which were transferred either partially or fully to initiate training in target domains. Shapley Additive exPlanations (SHAP) algorithms were used to study the effect of weight transfer on model explanability. Compared to the benchmark, the proposed weight transfer model showed statistically significant gains in prediction accuracy (between 1% and 5%) as well as computation time (up to 2hrs) for some target domains. The proposed method thus provides an adapted clinical decision support system for hospital management that can ease processes of data access via ethical committee, computation infrastructures and time.

Keywords— Length of Stay; Time-Series Prediction; Domain Adaptation; Long-Short Term Memory; SHAP feature explanability

1 Introduction

Monitoring patients' health condition or recovery trajectory as soon as they are admitted to the hospital or to a critical life-saving unit such as the intensive care unit (ICU) is important to determine and anticipate their needs throughout their entire stay. Patients could transit from a stable to an acutely ill state while under treatment, requiring immediate assistance leading to a prolonged stay. By continuously feeding AI-powered data-driven models, including deep learning methods, with continuously charted patient data (labs, vital signs, medications, prescriptions, etc.), researchers have been able to predict patients' risk of mortality [1–5], risk of extended length of hospital stay (LoS) [6, 7], remaining time in ICU [2, 5, 8–10], hours or days ahead of effective discharge or death. Predictions of LoS or remaining time in ICU would then enable hospital management systems to efficiently allocate both human and logistic resources [2, 11, 12], reassure family members and more importantly improve patient care and satisfaction [1, 13]. LoS, defined as the duration of inpatient stay from admission to discharge in a single care episode [14], is a tool used to assess the efficiency and effectiveness of hospital management systems. LoS predictions for each individual patient within the hospital at time t would enable timely capacity planning and management. With the large volumes of data being generated by patients during their hospital stay, accurate, robust and generalizable LoS prediction models can be developed. Robustness being accounted for by the heterogeneous nature of the patient population within an entire hospital.

However, the heterogeneity in the patient population that is reflected in the data, for e.g., the difference in the length and density of data points between ICU and non-ICU patients as mentioned in [15], or in varying frequencies of recording of the same parameters across different units [16], different age or disease groups, poses a number of challenges. One of which is the risk of affecting the overall model accuracy especially when the underlying distribution in each sub-population is different. In order to handle this diversity in patient populations, several authors have restricted their work on specific patient sub-populations delineated by age groups [17, 18], diagnostics [19] or medical units [20, 21]. In the work presented in [16], the authors highlighted this difference in patient behaviors by identifying that PCO2 was more frequently measured in cardiac ICU while Troponin T more frequently in coronary ICU. More specifically, regarding patient

heterogeneity and clinical outcome prediction, [22] first clustered ICU patients to obtain more homogeneous groups (clusters) onto which a multitask model was learned for mortality prediction. Multitask learning, much related to transfer learning (TL) [23] was used to simultaneously train and share learned parameters for mortality risk predictions for all clusters. Similarly, by considering four ICU sub-populations as different domains (either source or target), domain adaptation was applied in [16] for fine-tuning pre-trained weights from a CNN-LSTM-FCN network for mortality risk predictions.

Domain adaptation is a subclass of TL in which a learned model from a source domain is transferred to a target domain for the prediction of the same task, while accommodating to the difference in data distributions across domains [23]. In the present work, domain adaptation is also applied, for LoS prediction. In both [16] and [22], the authors acknowledged the specificities in patient populations for which individual models were constructed using a mechanism (domain adaptation and multitask learning respectively) that also allowed to simultaneously learn the existing similarities in these different populations with a net result in prediction accuracy gains as well as computation time. In our work, the specificity of each patient population is emphasized by not restricting the input spaces across all populations to be identical. More specifically, by setting a threshold on the frequency of recording of patient parameters, most recorded features in each population were kept for modelling. Domain adaptation was then applied to transfer pre-trained weights from a source to a target domain, where these two domains could have non-identical input spaces. By so doing, the transfer of pre-trained weights only happened for coinciding features between the two domains and for non-coinciding features, random weights were used. As a means to understand this simultaneous training of both pre-trained and random weights, an analysis was carried out by employing discriminative learning [24] where different learning rates were assigned to the two sets of features (coinciding and non-coinciding).

In the present work, we employ domain adaptation by transferring knowledge learned from one medical unit or population (source domain) unto others (target domains). Benefits of this approach when transposed to a real hospital environment are multifold because it can help overcome technical challenges related to data modelling and accuracy as well as managerial challenges related to data access and ethical approvals from a hospital. On the technical side, the storage, manipulation, curation, pre-processing and modelling of all patients' data of an entire hospital might stand as a big computing resource challenge in terms of memory capacity and computation time. The latter which might even be more problematic for a real time LoS prediction setup. On the managerial side, obtaining ethical approval for accessing a portion or a unit-level hospital data with addendum for a single unit at a time can be easier and more importantly, faster than requesting one for all patients in the entire hospital. Moreover, given that most hospitals (e.g. in Europe) have been incentivized to work in a very decentralized manner [25], i.e. allowing decision making at the local hospital unit levels, launching projects like LoS predictions at a unit (smaller) level by the unit head can be more effective and faster than from the hospital top management.

The key contributions of this work are as follows;

- We apply domain adaptation from a source to target domains for LoS predictions for ICU patients using first 24h of data and obtain significant gains in both prediction accuracy and computation time.
- By using a second dataset with four potential source domains, we investigate and gain insights into the choice of a suitable source domain for weights initialization on the target domains.
- We perform further analysis to understand the relevance and gains of domain adaptation including; discriminative learning to understand the effect of simultaneously training pre-trained and random weights on model accuracy and computation time, the effect of weight transfer on feature importance on the target domains, the transfer of the full model and the use of source domain hyperparameters on all target domains.

The remainder of this work is organized as follows; Section 2 describes the datasets, the model architecture including domain adaptation, expected gradients method and evaluation metrics. Section 3.2 covers all results including additional analyses carried out followed by a discussion on these results in Section 4. Finally, in Section 5, the work is concluded, limitations are discussed and the direction for future work is provided.

2 Materials and Methods

This section discusses the datasets used with descriptive statistics of each ICU unit, the prediction task and the proposed modelling strategies for domain adaptation via weight transfer with and without applying discriminative learning.

2.1 Datasets

The multi-centre eICU collaborative research database (eICU-CRD) [26] and uni-centre MIMIC-IV [27, 28] database were used. By modifying the pipeline by [2], we selected and pre-processed the first 24 hours of data into ICU. In eICU-CRD, data from 91,277 unique patients corresponding to 110,079 stays admitted to 8 ICU units; Medical Surgical (Med-Surg ICU), Coronary care - cardio-thoracic ICU (CCU-CTICU), Medical ICU (MICU), Neurological ICU (NICU), Cardiac ICU (CICU), Surgical ICU (SICU), Cardio-thoracic ICU (CTICU) and Cardio-surgical ICU (CSICU) was extracted. Similarly, data from 44,245 unique patients and 60,492 ICU stays admitted to 9 ICU units; Medical ICU (MICU), Medical Surgical (Med-Surg ICU), Cardiac Vascular ICU (CVICU), Surgical ICU (SICU), Trauma ICU (TICU), Coronary care ICU (CCU), Neuro surgical ICU (Neuro SICU), Neuro intermediate (NI), Neuro stepdown (NS) was extracted from MIMIC-IV. After hourly sampling extracted data using the mean, features were only kept for modelling if at least 2 unique recordings over 24h were present for at least 30% of the patients.[1]. Missing values were then imputed by first forward filling and then backward filling the most recent value for each patient, as in [29, 30].

Table 1: Baseline Characteristics per ICU unit in eICU-CRD. Total No. features = (No.Inputs $\times$ 2)+1 [2]

ICU unit	No. ICU stays	No.Inputs	Mean(LoS)	Std(LoS)	Median(Age)	Gender (% Male)
Med-Surg ICU	58,335	25	7.55	6.88	66	52.9
MICU	10,128	24	9.24	8.65	65	52.1
CCU-CTICU	9,950	27	7.24	6.78	67	59.0
NICU	8,777	25	8.26	8.04	63	51.9
CICU	7,744	25	7.87	7.61	65	50.4
SICU	7,684	26	9.45	8.47	65	57.5
CTICU	4,286	33	8.79	7.58	66	62.5
CSICU	3,175	35	7.32	6.39	69	59.8
All stays	110,079	26	7.93	7.36	65	54.3

[1] see Appendix A for details on feature extraction

[2] The input space was augmented with binary indicators of the same size as the original input space to indicate imputed and newly recorded parameter values following [31]. A variable *hour* indicating the time of the day at which a record was done was also added.

Table 2: Baseline Characteristics per ICU unit in MIMIC-IV. Total No. features = (No.Inputs × 2)+1

ICU unit	No. ICU stays	No.Inputs	Mean(LoS)	Std(LoS)	Median(Age)	Gender (% Male)
MICU	12,378	29	9.88	8.99	65	54.4
CVICU	11,070	38	8.05	7.03	69	67.4
Med-Surg ICU	10,152	23	9.93	9.41	66	51.3
SICU	9,262	28	10.82	9.94	64	53.5
TSICU	6,992	35	7.87	7.61	64	57.6
CCU	6,881	24	8.36	7.72	72	57.1
Neuro SICU	1,646	28	11.71	10.75	67	50.6
NI	1,491	20	7.15	7.67	68	50.6
NS	620	20	7.53	7.46	68	52.6
All stays	60,492	33	9.69	9.02	67	56.5

2.2 Prediction task

In this study, we predict the time lapse between ICU admission and hospital discharge using the first 24h of data into ICU. Selected admissions in both datasets are such that they have spent at least 24h in ICU to prevent data leakage. Datasets were split into train, validation and test sets following the 70:15:15 ratio.

2.3 Model architecture

The model architecture used consists of an LSTM layer followed by a fully connected layer (FCN) to handle the temporal dimension in the data and output the LoS prediction (in fractional days) respectively.

2.3.1 Long-Short Term Memory neural network (LSTM)

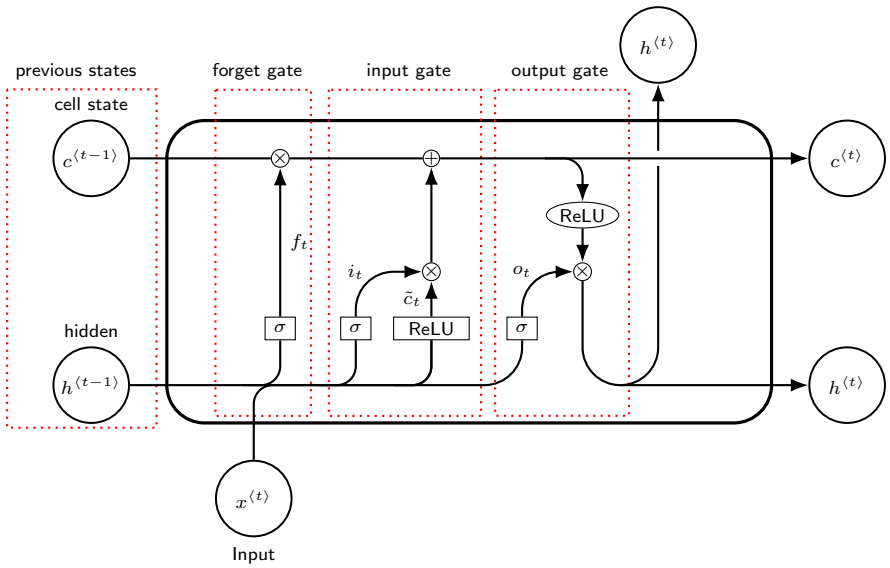


Figure 1: LSTM architecture with forget gate [32]

LSTM networks [33] belong to the family of RNNs [34–37] used to model and learn long-term dependencies from sequential data. The structure used in this work is given in Figure 1. At the core of RNNs like LSTM

is the recurrent cell with its memory cell state c_t that holds previous states and current input information. The recurrent cell of the LSTM in Figure 1 is given by;

$$\begin{aligned} \tilde{c}_t &= g\left(W_{\tilde{c}h} h_{t-1} + W_{\tilde{c}x} x_t + b_{\tilde{c}}\right) \\ c_t &= f_t \cdot c_{t-1} + i_t \cdot \tilde{c}_t \\ h_t &= o_t \cdot g(c_t), \end{aligned} \tag{1}$$

where h_t is the hidden state at a discrete time t (here, $t = 1$ hour), W_{ab} are weight parameters from layers a to b, $\cdot$ is the element-wise product and $g()$, a non-linear activation function applied to the results of matrix operations. Here, $g()$ was taken as the rectified linear Unit (ReLU) [38];

$$ReLU(x) = max(0, x). \tag{2}$$

The gate functions f_t, o_t and i_t control the amount of information flow via a sigmoid (logistic) activation function. These gates are respectively the forget, the output and the input and are defined as;

$$\begin{aligned} i_t &= \sigma\left(W_{ih} h_{t-1} + W_{ix} x_t + b_i\right) \\ f_t &= \sigma\left(W_{fh} h_{t-1} + W_{fx} x_t + b_f\right) \\ o_t &= \sigma\left(W_{oh} h_{t-1} + W_{ox} x_t + b_o\right). \end{aligned} \tag{3}$$

For implementation, we used the open source deep learning (DL) library, Keras [39] with Tensorflow [40] as back-end. In Keras, the weights W_{ab} are distributed in the kernel, the recurrent kernel and the bias term of the LSTM layer. The kernel stores all weights multiplied by the inputs x_t, i.e., $\boldsymbol{W_x} = \{W_{\tilde{c}x}, W_{ix}, W_{fx}, W_{ox}\}$, the recurrent kernel, those multiplied by the hidden state h_{t-1}, i.e., $\boldsymbol{W_h} = \{W_{\tilde{c}h}, W_{ih}, W_{fh}, W_{oh}\}$ and the bias stores all bias terms, i.e., $\boldsymbol{b} = \{b_{\tilde{c}}, b_i, b_f, b_o\}$. Therefore, the kernel weight matrix is of shape $(n_D \times 4H)$, where n_D is the input dimension of the domain D (source or target) and H, the number of hidden units of the LSTM layer. The recurrent weight matrix is of shape $(H \times 4H)$ and the bias is a vector of shape $(4H \times 1)$. From equations (1) and (3), it follows that the model generates a prediction at each time-step t given by h_t. This structure sometimes referred to as a many-to-many LSTM architecture was not used in this work, rather the many-to-one structure, in order to get only the prediction at the last time-step ($t = 24h$).

2.3.2 Fully connected network (FCN)

Given the last hidden state h_t from the LSTM layer, the final LoS prediction was computed as;

$$\hat{y}_t = g\left(W_{y_t} h_t + b_{y_t}\right), \tag{4}$$

with g again taken as the ReLU to output positive predictions. This fully connected layer contains only one unit to perform LoS regression and output a unique value for LoS.

2.4 Domain adaptation

2.4.1 Domain adaptation via weight transfer

The literature has shown that the use of pre-trained weights can be more beneficial than using random initial weights, sometimes, regardless of the difference in prediction tasks [41] and even from one related dataset to another [42, 43]. The net benefit of fine-tuning these pre-trained weights on the target domain is often computation time for hyperparameters optimization and model training [44].

Given the input space $\bar{\boldsymbol{X}}$ extracted from the database, the model inputs $\boldsymbol{X}$ are obtained by augmenting this input matrix with a matrix of binary indicators $\tilde{\boldsymbol{X}}$ such that, $\boldsymbol{X} = \text{concat}(\bar{\boldsymbol{X}}; \tilde{\boldsymbol{X}})$ where,

$$\tilde{\boldsymbol{X}} = \begin{cases} 0, if\ \bar{\boldsymbol{X}} \text{is recorded} \\ 1, if\ \bar{\boldsymbol{X}} \text{is imputed,} \end{cases} \tag{5}$$

following [31].

Following feature extraction in section 2.1, the input matrix $\boldsymbol{X}$ can differ both in the number of features and/or the features themselves (see, Tables 1 and 2 and Appendix A) between units. Thus given a source (S) and a target (T), we have;

$$\boldsymbol{X}_S \subset \boldsymbol{X}_T \tag{6a}$$

$$\boldsymbol{X}_S \not\subset \boldsymbol{X}_T \ \text{ or } \ \boldsymbol{X}_T \not\subset \boldsymbol{X}_S \tag{6b}$$

$$\boldsymbol{X}_S = \boldsymbol{X}_T \ \text{ or } \ \boldsymbol{X}_T \subset \boldsymbol{X}_S \tag{6c}$$

In the case of (6a) and (6b), only partial weight transfer from the source to the target can occur for coinciding features between the two domains. In case (6c), where all inputs in T are found in S, total weight transfer occurs. The proposed architecture is given in Figure 2.

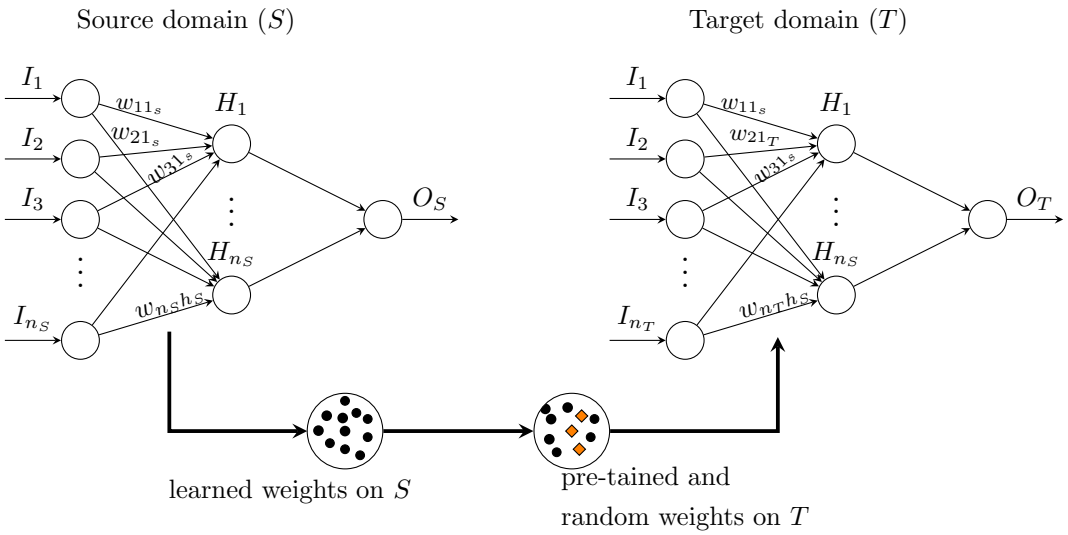


Figure 2: Proposed architecture: Domain adaptation via weight transfer from the source domain to the target domains. $I_{n_S}, I_{n_T}, H_{n_S}, O_S, O_T$ are respectively, the source inputs, target inputs, source hidden unit, source output and target output and $n_S \neq n_T$. Black dots are weights of coinciding features and orange diamonds are random weights for non-coinciding features.

The pseudo algorithm used for assigning pre-trained weights (either partially or totally) from the source to the target is given in Algorithm 1. After initializing the LSTM model for each T, model weights are set using the fully trained model from S and/or random kernel weights following Algorithm 1, recurrent kernel and bias weights from S, i.e., $[\boldsymbol{W}_{xT}, \boldsymbol{W}_{hS}, \boldsymbol{b}_S]$.

2.4.2 Domain Adaptation via weight transfer and discriminative learning

During TL, all hyperparameters from S except the batch size are transferred, i.e., the learning rate, the dropout rate and the number of hidden units from S are all transferred to T. Our intuition here is that

Algorithm 1 Weight Transfer from Source domain S to Target domain T

Inputs: Given:
Target inputs of size $m_T \times 24 \times n_T$: $\boldsymbol{X}_T = \{x_{1t}, ..., x_{n_{T_t}}\}, \quad t = 1, ..., 24$
Source inputs of size $m_S \times 24 \times n_S$: $\boldsymbol{X}_S = \{x_{1t}, ..., x_{n_{S_t}}\}, \quad t = 1, ..., 24$
W_{xS}= load source model kernel weights of size $N_S \times 4H$
Output: Target kernel weight matrix W_{xT} of size $N_T \times 4H$
Initialization of weight matrix W_{xT} using Glorot uniform distribution [45]
for $x \in \boldsymbol{X}_T$ **do**
  target_index = $\boldsymbol{X}_T[x]$
  if $x \in \boldsymbol{X}_S$ **then**
    W_{xT}[target_index] = W_{xS}[source_index]
  else
    $W_{xT}[x] = W_{xT}[x]$
  end if
end for

during training, coinciding features between T and S that receive fully trained weights are overfitting whilst the random weights of non-coinciding features are still learning. To study this, discriminative fine-tuning was applied, (see Figure 3), s.t., different learning rates (lr) are assigned to the two groups of inputs (layers) and optimized using a multi-optimizer [24, 46].

$$lr_T = \begin{cases} lr_S & if \quad X_T \notin \boldsymbol{X}_S \\ \alpha lr_S & if \quad \boldsymbol{X}_S \equiv \boldsymbol{X}_T \end{cases} \tag{7}$$

The learning rate from S is transferred to the non-coinciding features (features in T not in S) while it is reduced by a power of α for coinciding features.

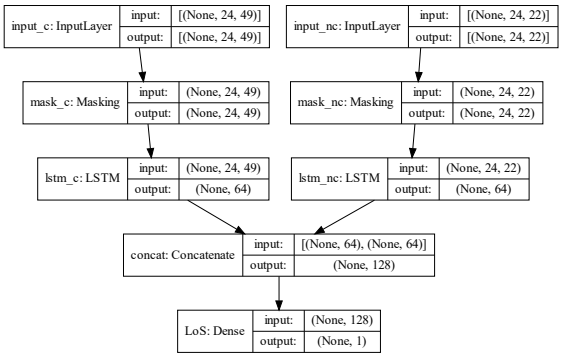


Figure 3: Transfer learning model with different learning rates assigned to layers *lstm_c* (layer with coinciding features) and *lstm_nc* (layer with non-coinciding features) for CSICU in eICU dataset

2.5 Interpretability using Expected gradients:

Expected gradients (EG) method [47] were used in order to unlock the black-box nature of our LSTM models and appreciate the effect of weights transfer on overall feature importance. The gradient explainer of the SHAP software package [3] was used. Ample details on the method can be found in [47].

[3] https://github.com/slundberg/shap

2.6 Evaluation metrics:

Model training was carried out using the mean squared log error (MSLE) to take into account the skewness in the outcome and model performance reported using the following metrics,

$$MAE = \frac{1}{n_D}\sum_{i=1}^{n_D}|y_i - \hat{y}_i|, MAPE = \frac{1}{n_D}\sum_{i=1}^{n_D}\left|\frac{y_i - \hat{y}_i}{y_i}\right|, MSE = \frac{1}{n_D}\sum_{i=1}^{n_D}(y_i - \hat{y}_i)^2,$$

where n_D is the number of stays in each domain.

3 Results

This section discusses all numerical results without and upon applying weights transfer, including some further analyses such as no hyperparameter optimization on target domains, full model transfer and discriminative learning.

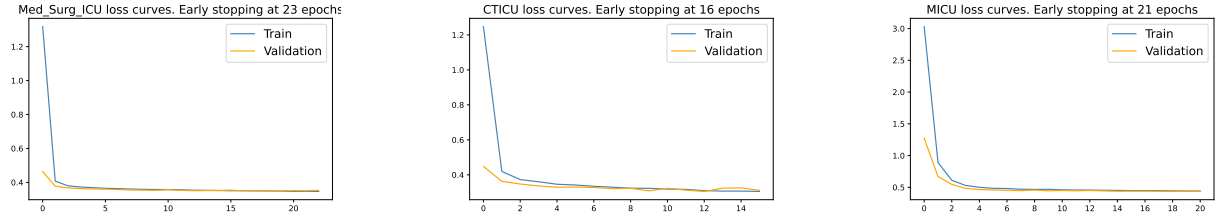
3.1 Hyperparameter Optimization

Training and model optimization were done following a systematic procedure (see Appendix B.1) by carefully monitoring both the generalization error (error on test set) and the loss curves in Figures 4 and 5. The batch size, the dropout, the number of hidden layers (No.layers), the number of hidden units (No.units) and the learning rate (lr) were all model hyperparameters obtained via Bayesian optimization [48, 49] and are shown in Tables 3 and 4. The Adam optimizer was used.

3.1.1 eICU-CRD data

Table 3: Hyperparamters results on eICU. S: source domain and T: target domain.

ICU data	batch size	No.layers	No.units	lr	dropout	domain type
Med-Surg ICU	512	1	64	1e-3	0.1	S
MICU	128	1	32	8.99e-4	0.1	T
CCU-CTICU	128	1	16	1.129e-3	0.2	T
NICU	128	1	16	1e-3	0.2	T
CICU	32	1	64	1e-3	0.2	T
SICU	64	1	32	8.99e-4	0.2	T
CTICU	64	1	16	1e-3	0.2	T
CSICU	64	1	64	1e-4	0	T
All stays	512	1	64	0	1.129e-3	



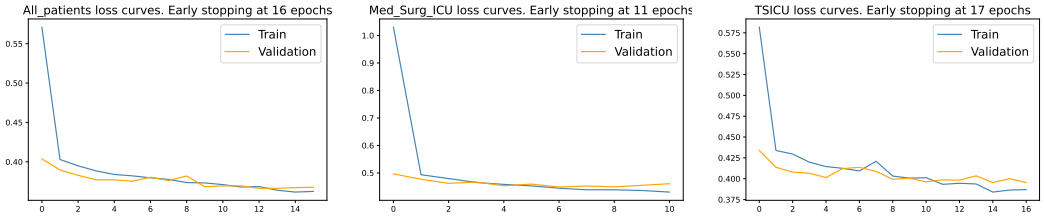
(a) Med-Surg ICU (b) CTICU (c) MICU

Figure 4: Training and Validation curves obtained after model optimization on some eICU-CRD units.

3.1.2 MIMIC-IV data

Table 4: Hyperparamters results on MIMIC-IV. S: source domain and T: target domain.

ICU data	batch size	No.layers	No.units	lr	dropout	domain type
MICU	128	1	32	7.142e-4	0.3	S or T
CVICU	128	1	64	5.46e-3	0.3	S or T
Med-Surg ICU	128	1	64	8.99e-4	0.2	S or T
SICU	128	1	64	4.833e-4	0.2	S or T
TSICU	64	1	64	1e-2	0.3	T
CCU	64	1	16	1e-2	0.1	T
Neuro SICU	16	1	8	6.952e-4	0.2	T
NI	16	1	16	4.833e-4	0.2	T
NS	8	1	16	1.624e-4	0.1	T
All stays	512	1	64	0.1	1.438e-3	



(a) All patients data (b) Med-Surg ICU (c) TSICU

Figure 5: Training and Validation curves obtained after model optimization on some MIMIC-IV units.

3.2 Baseline and weight transfer models

The performance of the weight transfer model was compared to that of the baseline model, where individual models were constructed for each unit using the hyperparameters obtained from optimization and listed in Tables 3 and 4. Also, for comparisons with the common practice in the literature, a model was ran where all units are considered as a single homogeneous entity and their data learned together in a single "all stay" model. For the weight transfer model, all hyperparameters except the batch size from the source are transferred to the targets. All models were ran 100 times by shuffling the data to obtain a distribution of the prediction error on the test set. 95% confidence intervals were estimated by calculating the 2.5th and 97.5th percentiles of the prediction error on the test set. Pairwise t-test in population means between "all stay" model and each population model in Tables 5 and 6 are indicated with stars (* $p < 0.05$, **$p < 0.001$). The red stars on boxplot figures indicate instances where the model stagnates, i.e., doesn't learn and stops after a few epochs due to the early stopping condition (that prevents overfitting).

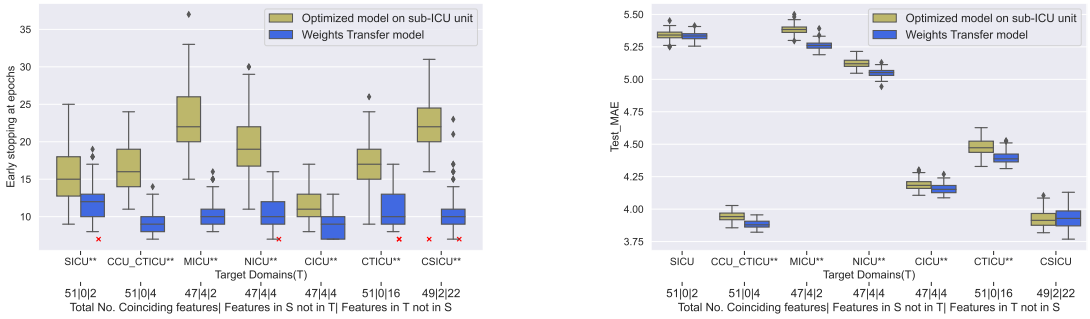
3.2.1 eICU-CRD data

Table 5 shows that feeding the model with all ICU stays data for LoS predictions irrespective of their admission units can sometimes overestimate the error in a unit. Looking at the most interpretable error measure, that is, MAE (reported in fractional days), we see that the error of the "all stay" model significantly over estimates that of Med-Surg ICU, CCU-CTICU, CICU and CSICU by 3.9, 6.35, 1 and 8.69 hours respectively.

Table 5: eICU-CRD: Baseline models for each unit and "all stay model": error estimates on test set with 95% confidence intervals: All stays data fitted in one model and one model per ICU domain (individuals models are learned and trained on each ICU domain).

ICU unit	MAE	MAPE	MSE
Med-Surg ICU	4.0311(4.0183-4.0559)**	0.6909(0.6614-0.7111)**	45.2873(44.8557-46.2811)**
MICU	5.3410(5.3158-5.4591)**	0.7722(0.7692-0.8431)**	80.4968(77.6054-82.2602)**
CCU-CTICU	3.9303(3.8816-4.0206)**	0.6497(0.6532-0.7227)**	49.6607(47.1492-50.0401)**
NICU	5.1263(5.0596-5.2009)**	0.8481(0.7985-0.8689)**	68.9331(67.0461-70.8095)**
CICU	4.1531(4.1224-4.2743)**	0.7305(0.6959-0.8269)**	48.8806(46.7863-50.0562)**
SICU	5.2859(5.2696-5.2966)**	0.6854(0.6463-0.7260)**	78.4585(77.5238-83.1262)**
CTICU	4.5133(4.3767-4.5937)**	0.5555(0.5179-0.5918)**	59.7552(55.8821-60.9689)**
CSICU	3.8328(3.8235-4.0838)**	0.6930(0.6703-0.7971)**	41.4605(41.4058-47.2725)**
All stays	4.1948(4.1851-4.2203)	0.6786(0.6568-0.7030)	50.3304(49.9702-51.4429)

Figure 6 shows the effect of using pre-trained weights from S as initial weights for coinciding features and assigning all hyperparameters (except the batch size) from S to T. In Figure 6(a), we see that model convergence always occurs significantly faster even when a good number of the features in T are not in S, as it is the case for CSICU and CTICU. Figure 6(b) shows that for most of the units, specifically, CCU-CTICU, MICU, NICU, CICU and CTICU, weight transfer significantly improves prediction accuracy.



(a) Number of epochs to reach convergence.

(b) Mean absolute error in days on test set

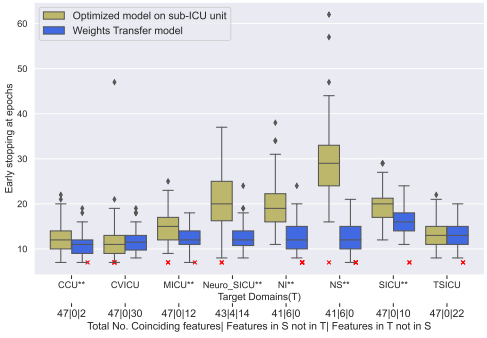
Figure 6: Distribution of number of epochs and error measures on Test set with (blue) and without (lemon green) transfer learning on eICU. Statistically significance per unit performed using a t-test (* p< 0.05, **p< 0.001).

3.2.2 MIMIC-IV data

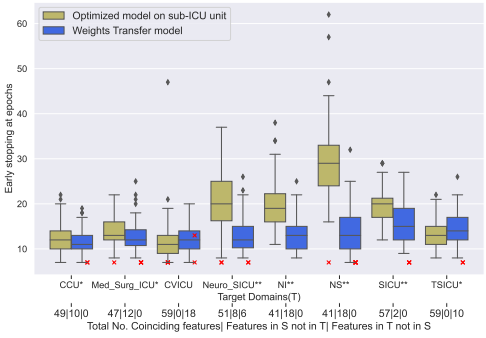
The observation drawn from Table 5 holds for MIMIC-IV data in Table 6, where the "all stay" model does not always perform significantly better than unit models as seen for CVICU, CCU and NI. For these units, the "all stay" model MAE error over estimates the MAE error by 1.79, 0.57 and 1.15 days respectively.

Table 6: MIMIC-IV: Baseline models for each unit and "all stay" model: error estimates on test set with 95% confidence intervals: All stays data fitted in one model and one model per ICU domain.

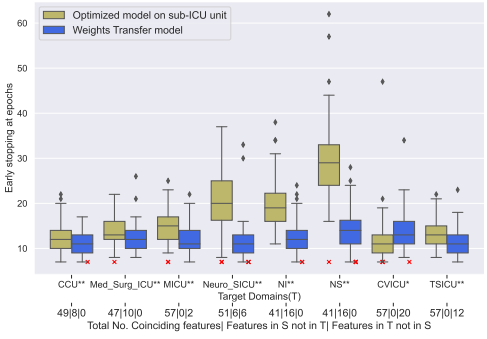
ICU unit	MAE	MAPE	MSE
Med-Surg ICU	5.6835(5.5664-5.7218)**	0.8407(0.7480-0.8633)**	91.4511(88.7403-94.8977)**
MICU	5.1596(5.0864-5.2703)	0.7046(0.6942-0.7968)**	68.3557(65.0615-69.3762)**
CVICU	3.3207(3.2265-5.9338)**	0.4041(0.3746-0.7682)**	42.7216(40.8025-78.0702)**
SICU	5.8401(5.8017-5.9516)**	0.6782(0.6808-0.7909)**	92.8705(87.6492-93.3682)**
TSICU	6.4758(6.3811-6.5416)	0.7936(0.6962-0.8180)	102.7655(98.9423-108.4076)
CCU	4.5474(4.4818-4.6177)**	0.7174(0.6637-0.7590)**	62.6330(61.2112-64.7852)**
Neuro SICU	7.2245(7.1893-8.1933)**	0.9108(0.8435-1.0569)**	141.0862(132.4134-220.3568)**
NI	3.9619(3.9564-4.2511)**	0.6184(0.6421-0.7700)**	50.7301(48.9124-55.6811)**
NS	5.5758(5.0034-5.7128)**	0.8699(0.6902-0.9102)**	94.2374(83.9943-102.3467)**
All stays	5.1140(5.0508-5.1424)	0.6687(0.6125-0.6781)	73.8215(73.0283-76.9459)



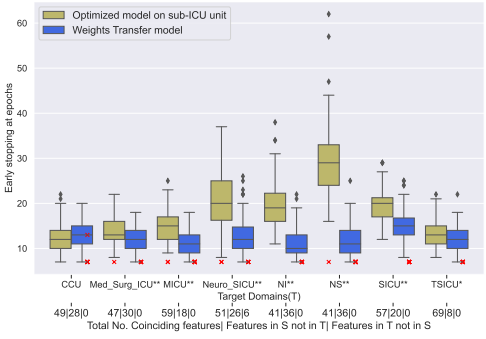
(a) Source domain (S): Med-Surg ICU



(b) Source domain (S): MICU



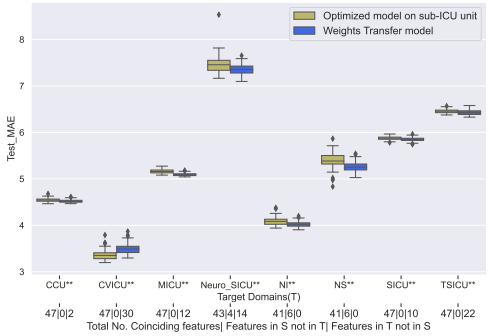
(c) Source domain (S): SICU



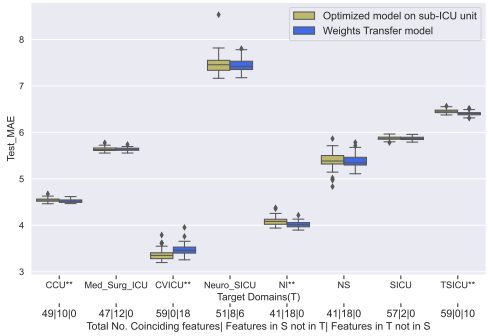
(d) Source domain (S): CVICU

Figure 7: Distribution of number of epochs to reach convergence with (blue) and without (lemon green) transfer learning on MIMIC-IV with Med-Surg ICU, MICU, SICU and CVICU as potential source domains. Statistically significance per unit performed using a t-test (* p< 0.05, **p< 0.001).

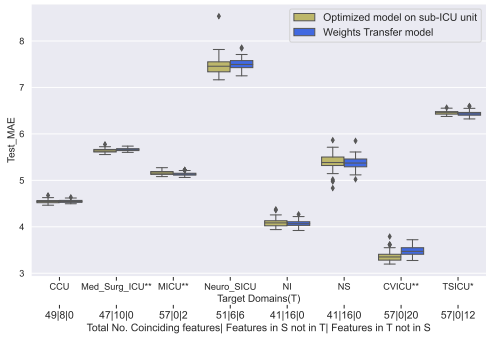
Looking at the four potential source domains on Figures 7 and 8, low populated ICUs like Neuro-SICU, NI and NS experience the highest gains with all source domains. As target domain, Med-Surg ICU gives the overall best improvement either in computation time, prediction accuracy or even both (see Figures 7(a) and 8(a)). As target domain, CVICU is negatively impacted by weight transfer (Figures 7(d) and 8(d)).



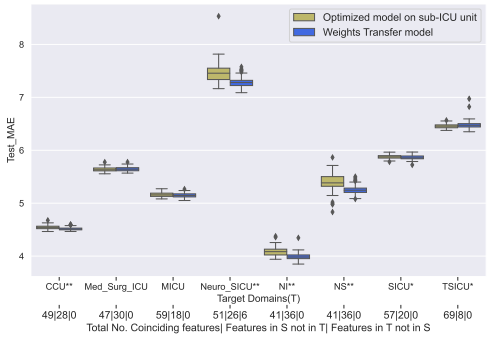
(a) Source domain (S): Med-Surg ICU



(b) Source domain (S): MICU



(c) Source domain (S): SICU



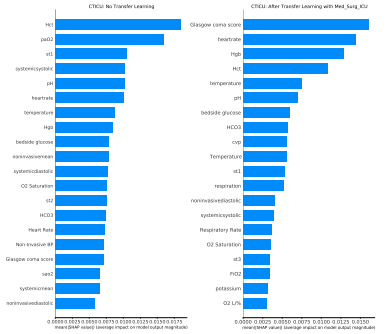
(d) Source domain (S): CVICU

Figure 8: Distribution of MAE error on test set with (blue) and without (lemon green) transfer learning on MIMIC-IV with Med-Surg ICU, MICU, SICU and CVICU as potential source domains. Statistically significance per unit performed using a t-test (* p< 0.05, **p< 0.001).

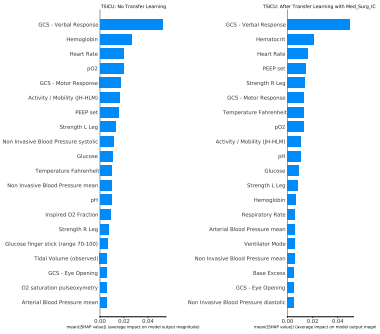
3.3 Model Interpretability

Given a trained model and the 3D training data set (number of patients × number of time-steps × number of inputs), 3D feature contributions for model prediction on the test set are obtained for each patient, at each time-step and for each input. Overall feature importance is then plotted by averaging over the time dimension and subsequently the patient dimension.

Patient features which can globally be split under charted parameters and labs are observed to be present before and after applying weight transfer as seen in Figure 9, though the order is not conserved.



(a) CTICU patients



(b) TSICU patients

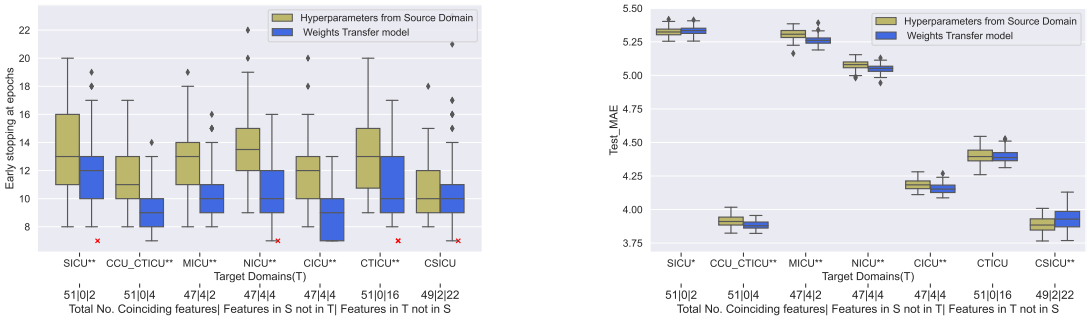
Figure 9: Top 25 most important features on CTICU (from eICU) and TSICU (from MIMIC-IV) targets before and after applying weight transfer

3.4 Further Analyses

To further understand the benefits of domain adaptation, we conducted three further analyses using eICU-CRD dataset. The first in which we do not optimize hyperparameters on each domain T and use those found on S, as seen in Figure 10. This was performed in order to understand whether weight transfer gains were due to poor hyperparameter optimization on each target unit. In the second analysis, non-coinciding features between S and T were removed and complete model transfer (including pre-trained weights and model optimizer state) was carried out. As final analysis, different learning rates were imposed on input feature groups as explained in section 2.4.2 and Eq(7) with $\alpha = 10^{-1}$.

3.4.1 No hyperparameter optimization on target units

Here, hyperparameters on all lemon green boxplots are obtain from the trained source domain model.



(a) Epochs

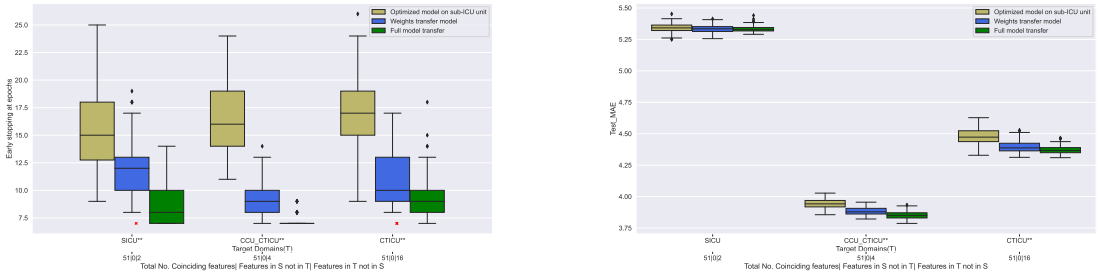
(b) Error on test set

Figure 10: Number of epochs and error measures with (blue) and without (lemon green) transfer learning on eICU-CRD where hyperparameters are fixed from source domain.

Figures 10(a) and 10(b) show that pre-trained weights do speed model convergence and improve prediction accuracy even when both models (with and without weight transfer) have the same hyperparameters.

3.4.2 Full Model Transfer

Here non-coinciding features are removed from the target domains such that there is complete correspondence of the feature space between the source and the targets. In this way, not only the weights are transferred but also the model optimizer state. Only SICU, CCU-CTICU and CTICU qualify for this exercise as their input space is a subset of the target, Med-Surg ICU.



(a) Epochs

(b) Error on test set

Figure 11: Number of epochs and error measures when performing full model transfer. Tukey's HSD test is used for multiple comparisons[50]. Individual model (lemon green), weight transfer (blue) and full model transfer - including weights and optimizer state (dark green).

Figure 11 shows that if we had restricted the input space across all domains to be identical and continued training the source domain model on the target domains input data, thereby transferring not only the weights but also the optimizer state, model convergence would have occurred earlier than when transferring only the weights with an improvement in prediction accuracy. However, this would not be possible for all units because not all most recorded features in S are found in all targets T.

3.4.3 Weight Transfer with different learning rates

Here, non-coinciding and coinciding features are trained simultaneously using different learning rates as explained in Section 2.4.2 and Eq (7).

(a) Epochs

(b) Error on test set

Figure 12: Number of epochs and error measures on CSICU from eICU-CRD dataset after assigning different learning rates to different features for transfer learning. Tukey's HSD test is used for multiple comparisons [50]. Individual model (lemon green), weight transfer (blue) and different learning rates (red).

Here it can be observed that indeed imposing discriminative learning produces a finer confidence interval (Figure 12(b)) for the prediction error while requiring more training time compared to weight transfer but still less than optimizing a full model (12(a)).

4 Discussion

The experiments in section 3.2 show that even partial weight transfer significantly improves prediction accuracy in a shorter time for most of the target domains. Furthermore, overall model importance is not affected.

4.1 Prediction error

Section 3.2 showed that considering the ICU populations as a single homogeneous entity greatly reduces model utility by hospital management systems because each patient is assigned to a bed belonging to a specific unit. Thus the information of the LoS prediction at a more granular level of the unit is important. Moreover, tables 5 and 6 showed that an "all stay" prediction model is not always an optimal estimation model for individuals groups. For e.g., CVICU patients in MIMIC-IV have a prediction error of 3.32 days, which is 1.79 days less than the former model. In eICU-CRD (Figures 6(a) and 6(b)), benefits are visible on populations such as CCU-CTICU (51,4), MICU (47,2) and CTICU (51,16) pairs of coinciding and non-coinciding features. However we notice that with (49,22), the prediction error increases for CSICU patients. Though the difference in the mean prediction error for SICU population is not statistically significant, we

observe that weight transfer model produces more confident results with a finer boxplot.
In MIMIC-IV, the main gain as observed in Figure 7 is that finer boxplots are obtained after weight transfer. For some targets, e.g., CVICU, a significant drop in prediction accuracy is observed in Figures 7(a), 7(b), 7(c) and 7(d). This could be associated to the fact that this unit has the highest number of input features. Thus, as target, a considerable number of non-coinciding features receive random weights (30 in Med-Surg ICU, 18 in MICU and 20 in SICU) that train alongside pre-trained weights.

The three further analyses carried out show that weight transfer has a significant positive impact on the model convergence and in several cases on the prediction error with at least, more confident error estimates.

4.2 Computation time

On average, the CPU time measured using python timer for the systematic Bayesian optimization procedure as explained in Appendix B.1 was 1 hour and 45 minutes (depending on the data size). Table 7 shows the execution times for some ICU units. OP_m: Optimized model on each ICU unit, WT_m: Weight transfer model, FT_m: Full model transfer.

Table 7: CPU time in hours for re-training all models 100 times in eICU-CRD data.

ICU units	$100 \times OP_m$	$100 \times WT_m$	$100 \times FT_m$
NICU	2.24	1.40	
MICU	2.99	1.95	
CSICU	1.75	1.07	
SICU	2.87	2.39	1.86

Given our resources (see headline of Appendix B) and the size of these units, a minimum of 1 hour was saved on units like MICU, CSICU taking into account the time needed for hyperparameter optimization. Optimization not done during TL since hyperparameters come from the source. Regarding MIMIC-IV data on Table 8, over 2 hours are saved for target domain NS. As discussed before, weight transfer has a negative impact on CVICU as a target domain requiring more computation time.

Table 8: CPU time in hours for re-training models 100 times in MIMIC-IV data

$100 \times OP_m$ on T_k	$100 \times WT_m$ with S_i			
NS	Med-Surg ICU	MICU	SICU	CVICU
3.63	1.66	1.78	1.85	1.59
Neuro-SICU	Med-Surg ICU	MICU	SICU	CVICU
3.44	2.32	2.26	2.21	2.60
NI	Med-Surg ICU	MICU	SICU	CVICU
2.75	2.11	2.23	2.28	2.29
CVICU	Med-Surg ICU	MICU	SICU	CVICU
1.75	1.99	2.18	2.10	/

4.3 Insights into domain adaptation

This section intends to discuss first, the reasons why domain adaptation works even when only partial information is transferred and secondly, the choice of the source domain.

Regarding eICU-CRD data (Figure 6), where the Med-Surg ICU unit is the only source domain, statistical significant improvements both in terms of early convergence and prediction error on all target populations except CSICU are observed. For CSICU, the error distribution appears to increase after partial weight transfer which could be due to the highest number of non-coinciding features (22) that receive random weights. As later shown in Figure 12, by assigning different learning rates, the error distribution is narrower (Figure 12(b)).

In an attempt to investigate the effect of the choice of the source domain, MIMIC-IV data was used with four potential source domains. Figures 7 and 8 show that contrary to eICU-CRD where the source domain Med-Surg ICU occupies over 50% of the data, this same domain returns the overall best performance when its weights are used to initiate training in the rest of the domains. Med-Surg ICU that contains both medical and surgical patients, that is a diverse patient population, has the greater impact on the other populations when used as source domain. In terms of similarity between populations, Figure 7(c) shows that the source domain SICU has the greatest impact on the target TSICU. However, instances when TL works or not could not all be explained. For e.g., in Figure 7(d), CVICU has a slightly negative though non-significant impact on CCU. Though CVICU is a surgical unit type, and CCU a medical one, SICU which is also surgical appears to have a significant positive impact on CCU (Figure 7(c)) with exactly the same 49 coinciding features. Thus pre-trained weights of CVICU used for total weight transfer do not improve learning as much as those of SICU.

4.4 Effects of domain adaptation on model interpretability

As shown in Figures 9(a) and 9(b), overall, the most important features which can be grouped under vital signs and lab parameters for early prediction of LoS are not affected by weight transfer though with slight changes in the order. Since the same groups of features appear as important before and after performing weight transfer.

4.5 True hospital setting

Though our work was not tested in a true hospital setting due to inability to access hospital data, we believe that transposing this to a true hospital setting where measures like the LILRANK [51] can be used to group or cluster departments based on their mode of functioning and other factors, our method can be applied at low cost within these clusters. Here the source can be chosen as a diverse population within the cluster and its weights used to initiate training to targets of the same cluster. This method can be of great benefit also for units with very small data sizes where their data will serve for fine-tuning pre-trained models rather than training a model from scratch.

5 Conclusions and Limitations

In this work, domain adaptation is exploited to reuse knowledge learned from a source unto target domains by transferring learned weights from the trained source model. By not restricting the input space such that it is identical across all units, we allow both shared and unit-specific information to be disseminated in the targets by fine-tuning both pre-trained (from the source) and random (unit-specific) weights. This resulted in statistically significant improvements in computation time as well as prediction accuracy for most of the

targets. However, we noticed that weight transfer was not always beneficial, especially when the target had a high number of non-coinciding features that receive random weights. By implementing discriminative learning and assigning different learning rates to the two feature groups (coinciding and non-coinciding), an improvement in the prediction error was noticed. In terms of feature importance, it appeared that the proposed approach maintains the overall importance by keeping the majority of important features though it displaces the order. Insights into this work showed that significant improvements in both prediction accuracy and computation time are observed when the source domain consists of a diverse population.

This work has a number of limitations. First, we couldn't fully understand all instances where weight transfer do not work. Secondly, when performing discriminative learning, a fixed factor of $\alpha = 10^{-1}$ was used to reduce the learning rate for coinciding features. Thirdly, only time-varying features were used to predict LoS. Finally, the method was not evaluated in a true hospital setting. As future work, optimization of α, the use of an adaptive learning rate and implementing mechanisms of data privacy, such as, differential privacy during weight transfer to prevent data leakage are envisaged.

CRediT authorship contribution statement

L.N.W.M: Conceptualization, Methodology, Data curation, Model implementation, Writing - original draft & editing. **N.M**: Methodology supervision. **E.N**: Methodology supervision & draft review. **F.R**: Supervision, draft review & editing. **B.DM**: Review.

Declaration of competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by KU Leuven: Research Fund (projects C16/15/059, C3/19/053, C24/18/022, C3/20/117, C3I-21-00316), Industrial Research Fund (Fellowships 13-0260, IOFm/16/004, IOFm/20/002) and several Leuven Research and Development bilateral industrial projects; Flemish Government Agencies: FWO: EOS Project no G0F6718N (SeLMA), SBO project S005319N, Infrastructure project I013218N, TBM Project T001919N; PhD Grants (SB/1SA1319N, SB/1S93918, SB/1S1319N), EWI: the Flanders AI Research Program VLAIO: CSBO (HBC.2021.0076) Baekeland PhD (HBC.20192204) and Innovation mandate (HBC.2019.2209) European Commission: European Research Council under the European Union's Horizon 2020 research and innovation programme (ERC Adv. Grant grant agreement No 885682); Other funding: Foundation 'Kom op tegen Kanker', CM (Christelijke Mutualiteit)

Appendix

A Patient features extracted per ICU unit

1. By using the pipeline by [2], SQL queries were used to extract data tables (*vitalperiodic*, *vitalaperiodic*, *respiratorycharting*, *lab*, *nursecharting* from eICU-CRD and *labevents*, *chartevents* from MIMIC-IV) by imposing thresholds of presence of each feature in at least 25% 13% or 12.5% of all adult patients.
2. Next, using the *unit type* column from *patient* table in eICU-CRD and *first_careunit* column from *icustays* table in MIMIC-IV, stay ids were collected for each ICU unit type.

3. Extracted tables in 1. were then filtered on stays ids from 2. of each ICU unit for both datasets.
4. Again by modifying the pipeline by [2], data curation and pre-processing was done per ICU unit type for both datasets one at a time.
5. In the pre-processing, all stays data tables for each unit, were merged, re-sampled hourly using the mean and scaled. From this, only features with at least 2 unique values for at least 30% of the patients were retained for modelling. For each dataset and for each ICU unit type, final extracted features are grouped under the table they appear in the original database.
6. Steps 2., 3., 4. and 5. were also performed on the initial tables extracted in 1. without filtering per *unit type* which constitutes the *all stays* data.

A.1 eICU data

A.2 MIMIC-IV data

Patients	Source Tables					
	vitalperiodic	***vitalaperiodic***	***respiratorycharting***	***lab***	***nursecharting***	***N/A***
All Stays	cvp, heartrate, st1, st2, st3, respiration, temperature, sao2	noninvasivediastolic, noninvasivemean, noninvasivesystolic, systemicdiastolic, systemicmean, systemicsystolic		FiO2, bedside glucose	Glasgow coma score, Heart Rate, O2 L/%, O2 Saturation, Pain Score/Goal, Invasive BP, Non-Invasive BP, Respiratory Rate, Sedation Scale/Score/Goal, Temperature	Time in ICU
Med Surg ICU	cvp, heartrate, st1, st2, st3, respiration, temperature, sao2	noninvasivediastolic, noninvasivemean, noninvasivesystolic, systemicdiastolic, systemicmean, systemicsystolic		FiO2, bedside glucose	Glasgow coma score, Heart Rate, O2 L/%, O2 Saturation, Pain Score/Goal, Invasive BP, Non-Invasive BP, Respiratory Rate, Temperature	Time in ICU
SICU	cvp, heartrate, st1, st2, st3, respiration, temperature, sao2	noninvasivediastolic, noninvasivemean, noninvasivesystolic, systemicdiastolic, systemicmean, systemicsystolic		FiO2, bedside glucose	Glasgow coma score, Heart Rate, O2 L/%, O2 Saturation, Pain Score/Goal, Invasive BP, Non-Invasive BP, Respiratory Rate, Sedation Scale/Score/Goal, Temperature	Time in ICU
NICU	heartrate, st1, st2, st3, respiration, sao2	noninvasivediastolic, noninvasivemean, noninvasivesystolic, systemicdiastolic, systemicmean, systemicsystolic	Total RR	FiO2, bedside glucose	Glasgow coma score, Heart Rate, O2 L/%, O2 Saturation, Pain Score/Goal, Invasive BP, Non-Invasive BP, Respiratory Rate, Sedation Scale/Score/Goal, Temperature	Time in ICU
MICU	heartrate, st1, st2, st3, respiration, sao2	noninvasivediastolic, noninvasivemean, noninvasivesystolic, systemicdiastolic, systemicmean, systemicsystolic		FiO2, bedside glucose	Glasgow coma score, Heart Rate, O2 L/%, O2 Saturation, Pain Score/Goal, Invasive BP, Non-Invasive BP, Respiratory Rate, Sedation Scale/Score/Goal, Temperature	Time in ICU
CTICU	cvp, heartrate, st1, st2, st3, respiration, temperature, sao2	noninvasivediastolic, noninvasivemean, noninvasivesystolic, systemicdiastolic, systemicmean, systemicsystolic		FiO2, bedside glucose, HCO3, Hct, Hgb, pH, paCO2, paO2, potassium	Glasgow coma score, Heart Rate, O2 L/%, O2 Saturation, Pain Score/Goal, Invasive BP, Non-Invasive BP, Respiratory Rate, Temperature, Bedside Glucose	Time in ICU
CICU	cvp, heartrate, st1, st2, st3, respiration, sao2	noninvasivediastolic, noninvasivemean, noninvasivesystolic, systemicdiastolic, systemicmean, systemicsystolic	PEEP	FiO2, bedside glucose	Glasgow coma score, Heart Rate, O2 L/%, O2 Saturation, Pain Score/Goal, Non-Invasive BP, Respiratory Rate, Sedation Scale/Score/Goal, Temperature	Time in ICU

CCU-CTICU	cvp, heartrate, st1, st2, st3, respiration, temperature, sao2	noninvasivediastolic, noninvasivemean, noninvasivesystolic, systemicdiastolic, systemicmean, systemicsystolic		FiO2, bedside glucose, potassium	Glasgow coma score, Heart Rate, O2 L/%, O2 Saturation, Pain Score/Goal, Invasive BP, Non-Invasive BP, Respiratory Rate, Sedation Scale/Score/Goal, Temperature	Time in ICU
CSICU	cvp, heartrate, st1, st2, st3, respiration, sao2	noninvasivediastolic, noninvasivemean, noninvasivesystolic, systemicdiastolic, systemicmean, systemicsystolic	Total RR, PEEP, Peak Insp. Pressure, TV/kg IBW, Mean Airway Pressure, Tidal Volume (set), Vent Rate, Exhaled MV, LPM O2	FiO2, bedside glucose, potassium	Glasgow coma score, Heart Rate, O2 L/%, O2 Saturation, Pain Score/Goal, Invasive BP, Non-Invasive BP, Respiratory Rate, Temperature, Bedside Glucose	Time in ICU

Table 9: List of Features extracted per ICU unit in eICU-CRD.

Patients	Source Tables		
	labevents	***chartevents***	***N/A***
All Stays	Base Excess, Calculated Total CO2, Glucose, Hematocrit, Hemoglobin, pCO2, pH, pO2	Activity / Mobility (JH-HLM), Arterial Blood Pressure diastolic, Arterial Blood Pressure mean, Arterial Blood Pressure systolic, GCS - Eye Opening, GCS - Motor Response, GCS - Verbal Response, Glucose finger stick (range 70-100), Heart Rate, Inspired O2 Fraction, Mean Airway Pressure, Minute Volume, Non Invasive Blood Pressure diastolic, Non Invasive Blood Pressure mean, Non Invasive Blood Pressure systolic, O2 saturation pulseoxymetry, PEEP set, Respiratory Rate, Richmond-RAS Scale, Strength L Arm, Strength L Leg, Strength R Arm, Strength R Leg, Temperature Fahrenheit, Tidal Volume (observed)	hour
MICU	Glucose, Hematocrit, Hemoglobin, pH	Activity / Mobility (JH-HLM), Arterial Blood Pressure diastolic, Arterial Blood Pressure mean, Arterial Blood Pressure systolic, GCS - Eye Opening, GCS - Motor Response, GCS - Verbal Response, Glucose finger stick (range 70-100), Heart Rate, Inspired O2 Fraction, Mean Airway Pressure, Minute Volume, Non Invasive Blood Pressure diastolic, Non Invasive Blood Pressure mean, Non Invasive Blood Pressure systolic, O2 saturation pulseoxymetry, PEEP set, Respiratory Rate, Richmond-RAS Scale, Strength L Arm, Strength L Leg, Strength R Arm, Strength R Leg, Temperature Fahrenheit, Tidal Volume (observed)	hour
CVICU	Base Excess, Calculated Total CO2, Free Calcium, Glucose, Hematocrit, Hematocrit, Calculated, Hemoglobin, Lactate, Potassium, Whole Blood, pCO2, pH, pO2	Activity / Mobility (JH-HLM), Arterial Blood Pressure diastolic, Arterial Blood Pressure mean, Arterial Blood Pressure systolic, GCS - Eye Opening, GCS - Motor Response, GCS - Verbal Response, Glucose finger stick (range 70-100), Heart Rate, Inspired O2 Fraction, Mean Airway Pressure, Minute Volume, Non Invasive Blood Pressure diastolic, Non Invasive Blood Pressure mean, Non Invasive Blood Pressure systolic, O2 saturation pulseoxymetry, PEEP set, Respiratory Rate, Richmond-RAS Scale, Strength L Arm, Strength L Leg, Strength R Arm, Strength R Leg, Temperature Fahrenheit, Tidal Volume (observed), Ventilator Mode	hour
Med-Surg ICU	Glucose, Hematocrit,	Activity / Mobility (JH-HLM), Arterial Blood Pressure diastolic, Arterial Blood Pressure mean, Arterial Blood Pressure systolic, GCS - Eye Opening, GCS - Motor Response, GCS - Verbal Response, Glucose finger stick (range 70-100), Heart Rate, Inspired O2 Fraction, Non Invasive Blood Pressure diastolic, Non Invasive Blood Pressure mean, Non Invasive Blood Pressure systolic, O2 saturation pulseoxymetry, Respiratory Rate, Richmond-RAS Scale, Strength L Arm, Strength L Leg, Strength R Arm, Strength R Leg, Temperature Fahrenheit	hour

SICU	Glucose, Hematocrit, Hemoglobin	Activity / Mobility (JH-HLM), Arterial Blood Pressure diastolic, Arterial Blood Pressure mean, Arterial Blood Pressure systolic, GCS - Eye Opening, GCS - Motor Response, GCS - Verbal Response, Glucose finger stick (range 70-100), Heart Rate, Inspired O2 Fraction, Mean Airway Pressure, Minute Volume, Non Invasive Blood Pressure diastolic, Non Invasive Blood Pressure mean, Non Invasive Blood Pressure systolic, O2 saturation pulseoxymetry, PEEP set, Respiratory Rate, Richmond-RAS Scale, Strength L Arm, Strength L Leg, Strength R Arm, Strength R Leg, Temperature Fahrenheit, Tidal Volume (observed)	hour
TSICU	Base Excess, Calculated Total CO2, Glucose, Hematocrit, Hemoglobin, pCO2, pH, pO2	Activity / Mobility (JH-HLM), Arterial Blood Pressure diastolic, Arterial Blood Pressure mean, Arterial Blood Pressure systolic, GCS - Eye Opening, GCS - Motor Response, GCS - Verbal Response, Glucose finger stick (range 70-100), Heart Rate, Inspired O2 Fraction, Mean Airway Pressure, Minute Volume, Non Invasive Blood Pressure diastolic, Non Invasive Blood Pressure mean, Non Invasive Blood Pressure systolic, O2 saturation pulseoxymetry, PEEP set, Respiratory Rate, Richmond-RAS Scale, Strength L Arm, Strength L Leg, Strength R Arm, Strength R Leg, Temperature Fahrenheit, Tidal Volume (observed), Ventilator Mode	hour
CCU	Glucose, Hematocrit, Hemoglobin	Activity / Mobility (JH-HLM), Arterial Blood Pressure diastolic, Arterial Blood Pressure mean, Arterial Blood Pressure systolic, GCS - Eye Opening, GCS - Motor Response, GCS - Verbal Response, Glucose finger stick (range 70-100), Heart Rate, Inspired O2 Fraction, Non Invasive Blood Pressure diastolic, Non Invasive Blood Pressure mean, Non Invasive Blood Pressure systolic, O2 saturation pulseoxymetry, Respiratory Rate, Richmond-RAS Scale, Strength L Arm, Strength L Leg, Strength R Arm, Strength R Leg, Temperature Fahrenheit	hour
Neuro-SICU		H, I, L, Activity / Mobility (JH-HLM), Arterial Blood Pressure diastolic, Arterial Blood Pressure mean, Arterial Blood Pressure systolic, GCS - Eye Opening, GCS - Motor Response, GCS - Verbal Response, Glucose finger stick (range 70-100), Heart Rate, Inspired O2 Fraction, Mean Airway Pressure, Minute Volume, Non Invasive Blood Pressure diastolic, Non Invasive Blood Pressure mean, Non Invasive Blood Pressure systolic, O2 saturation pulseoxymetry, PEEP set, Respiratory Rate, Richmond-RAS Scale, Strength L Arm, Strength L Leg, Strength R Arm, Strength R Leg, Temperature Fahrenheit, Tidal Volume (observed)	hour

NI		Activity / Mobility (JH-HLM), Arterial Blood Pressure diastolic, Arterial Blood Pressure mean, Arterial Blood Pressure systolic, GCS - Eye Opening, GCS - Motor Response, GCS - Verbal Response, Glucose finger stick (range 70-100), Heart Rate, Non Invasive Blood Pressure diastolic, Non Invasive Blood Pressure mean, Non Invasive Blood Pressure systolic, O2 saturation pulseoxymetry, Respiratory Rate, Richmond-RAS Scale, Strength L Arm, Strength L Leg, Strength R Arm, Strength R Leg, Temperature Fahrenheit	hour
NS		Activity / Mobility (JH-HLM), Arterial Blood Pressure diastolic, Arterial Blood Pressure mean, Arterial Blood Pressure systolic, GCS - Eye Opening, GCS - Motor Response, GCS - Verbal Response, Glucose finger stick (range 70-100), Heart Rate, Non Invasive Blood Pressure diastolic, Non Invasive Blood Pressure mean, Non Invasive Blood Pressure systolic, O2 saturation pulseoxymetry, Respiratory Rate, Richmond-RAS Scale, Strength L Arm, Strength L Leg, Strength R Arm, Strength R Leg, Temperature Fahrenheit	hour

Table 10: List of Features extracted per ICU unit in MIMIC-IV.

B Implementation details

All algorithms were written in Python language. All experiments were performed on a 128GB RAM computer with Intel(R) Core(TM) i9-10900X CPU processor and NVIDIA GeForce RTX 2080 Ti GPU.

B.1 Hyperparameter Search Methodology and Implementation Details

The hyperparameter search space for all ICU units is given in the table below. An asterisk * is indicated for those in which the space was enlarged for particular units.

Table 11: Hyperparameters search space and corresponding scale. No. Hidden units was enlarged to the interval [4 - 512] for Neuro-Stepdown ICU patients in MIMIC-IV and the learning rate to $[1e^{-5} - 1e^{-2}]$ for Medical-Surgical ICU patients in MIMIC-IV

Hyperparameter	Search Space Interval	Scale
No.Hidden Layers	[1 - 2]	Linear
No.Hidden units*	[8 - 512]	$\log_2$
Learning rate*	$[1e^{-4} - 1e^{-2}]$	$\log_{10}$
Dropout	[0.1 - 0.5]	Linear
Batch Size	[4 - 512]	$\log_2$

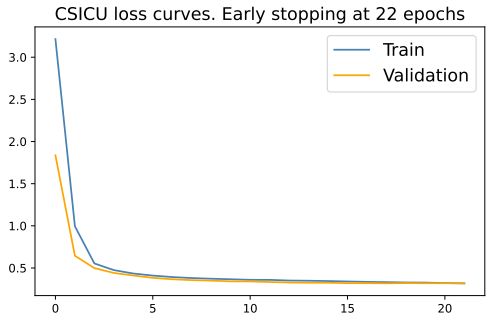
Hyperparameter search was performed in a systematic manner for each of the patient populations using the Bayesian Optimizer from KerasTuner [48, 49]. Essentially, Bayesian Optimization makes use of both a prior function and an acquisition function. The former is used as a surrogate to obtain estimates of the objective function and the latter, that measures the evaluation of the objective function at a new point and proposes next candidate points within the search space [52]. When using the Bayesian optimization, we minimize the validation loss, that is the, mean squared logarithmic error on the validation dataset. As explained before, the candidate hyperparameters values used at each trial are proposed depending on the performance of the previously chosen values. We noticed that orienting the search direction seriously affects the number of hidden units. Therefore, the hidden units space was split into two resulting in a three-step procedure as follows;

1. Firstly, the search space involving all hyperparameters except the batch size and No. hidden units in the interval [8 - 64] was used.
2. Secondly, the previous step was repeated with No. hidden units in the interval [64 - 512].
3. In the third step, the refined space following model performance was used and hyperparameter search was repeated.
4. The best hyperparameters from step three were then used to fit the model using early stopping to prevent overfitting. By monitoring the validation loss over six epochs, early stopping occurs if this doesn't further decrease by at least 0.5%.

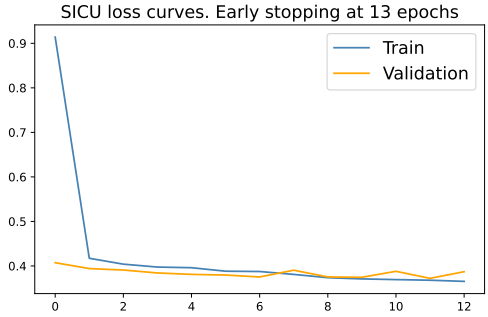
In steps 1 and 2, ten trials were done with two executions per trial. In step 3, ten trials were done with three executions per trial. By performing multiple executions per trial ensures that the reported hyperparameter values return the lowest and most stable average error over all trials.

C Loss Curves

Additional loss curves from Section 3.1

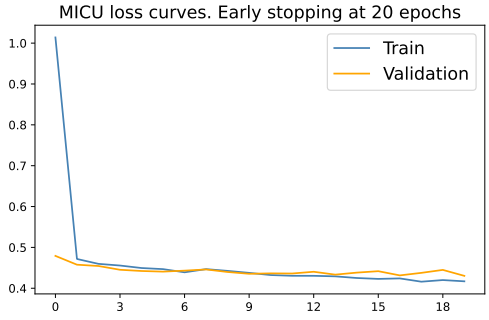


(a) CSICU patients

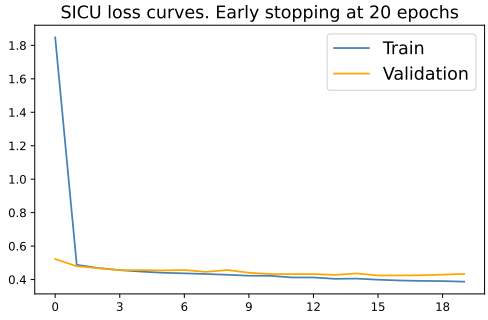


(b) SICU patients

Figure 13: Loss curves obtained after training CSICU and SICU in eICU-CRD data with optimized hyperparameters



(a) MICU patients



(b) SICU patients

Figure 14: Loss curves obtained after training MICU and SICU in MIMIC-IV data with optimized hyperparameters as in Section 3.1.

D Multiple comparisons of Means using Tukey test

Complementary results for Figure 11 and 12.

Table 12: SICU: Multiple comparison of Means for epochs and Test MAE using Tukey HSD [50], $\alpha = 5\%$. OM: Optimized Model on each sub-ICU unit, WTM: Weight Transfer Model, FTM: Full Transfer Model

	group 1	group 2	mean diff	p-adj	lower	upper	reject H_0
Epochs	OM	WTM	-3.6331	0.001	-4.5746	-2.6917	True
	FTM	OM	6.3301	0.001	5.3886	7.2716	True
	FTM	WTM	-2.697	0.001	1.7531	3.6408	True
Test MAE	OM	WTM	-0.0071	0.2555	-0.0177	0.0035	False
	FTM	OM	0.0095	0.0891	-0.0011	0.02	False
	FTM	WTM	0.0024	0.8418	-0.0082	0.013	False

Table 13: CCU-CTICU: Multiple comparison of Means for epochs and Test MAE using Tukey HSD, $\alpha = 5\%$. OM: Optimized Model on each sub-ICU unit, WTM: Weight Transfer Model, FTM: Full Transfer Model

	group 1	group 2	mean diff	p-adj	lower	upper	reject H_0
Epochs	OM	WTM	-7.07	0.001	-7.7585	-6.3815	True
	FTM	OM	9.04	0.001	8.3515	9.7285	True
	FTM	WTM	1.97	0.001	1.2815	2.6585	True
Test MAE	OM	WTM	-0.0591	0.001	-0.0702	-0.048	True
	FTM	OM	0.0922	0.001	0.0812	0.1033	True
	FTM	WTM	0.0331	0.001	0.022	0.0442	True

Table 14: CTICU: Multiple comparison of Means for epochs and Test MAE using Tukey HSD, $\alpha = 5\%$. OM: Optimized Model on each sub-ICU unit, WTM: Weight Transfer Model, FTM: Full Transfer Model

	group 1	group 2	mean diff	p-adj	lower	upper	reject H_0
Epochs	OM	WTM	-6.2298	0.001	-7.0824	-5.3772	True
	FTM	OM	7.8012	0.001	6.9487	8.6538	True
	FTM	WTM	1.5714	0.001	0.7146	2.42283	True
Test MAE	OM	WTM	-0.0829	0.001	-0.0992	-0.0666	True
	FTM	OM	0.1079	0.001	0.00916	0.1242	True
	FTM	WTM	0.0249	0.0011	0.0086	0.0413	True

Table 15: CSICU: Multiple comparison of Means for epochs and Test MAE using Tukey HSD, $\alpha = 5\%$. OM: Optimized Model on each sub-ICU unit, WTM: Weight Transfer Model, Diff_LR: Different Learning Rates Model

	group 1	group 2	mean diff	p-adj	lower	upper	reject H_0
Epochs	Diff_LR	OM	7.7071	0.001	6.6032	8.811	True
	Diff_LR	WTM	-4.0707	0.001	-5.1746	-2.9668	True
	OM	WTM	-11.7778	0.001	-12.8817	-10.6739	True
Test MAE	Diff_LR	OM	0.0173	0.2146	-0.0069	0.0415	False
	Diff_LR	WTM	0.0214	0.0961	-0.0029	0.0456	True
	OM	WTM	0.0041	0.9	-0.0201	0.0283	False

References

[1] Dongdong Zhang et al. "Combining structured and unstructured data for predictive models: a deep learning approach". In: *BMC medical informatics and decision making* 20.1 (2020), pp. 1–11.

[2] Emma Rocheteau, Pietro Liò, and Stephanie Hyland. "Temporal pointwise convolutional networks for length of stay prediction in the intensive care unit". In: *Proceedings of the Conference on Health, Inference, and Learning*. 2021, pp. 58–68.

[3] Emma Rocheteau et al. "Predicting Patient Outcomes with Graph Representation Learning". In: *arXiv preprint arXiv:2101.03940* (2021).

[4] Seyedmostafa Sheikhalishahi, Vevake Balaraman, and Venet Osmani. "Benchmarking machine learning models on multi-centre eICU critical care dataset". In: *PloS one* 15.7 (2020), e0235424.

[5] Hrayr Harutyunyan et al. "Multitask learning and benchmarking with clinical time series data". In: *Scientific data* 6.1 (2019), pp. 1–18.

[6] Jaret M Karnuta et al. "The value of artificial neural networks for predicting length of stay, discharge disposition, and inpatient costs after anatomic and reverse shoulder arthroplasty". In: *Journal of Shoulder and Elbow Surgery* 29.11 (2020), pp. 2385–2394.

[7] Aditya V Karhade et al. "Development of predictive algorithms for length of stay greater than one day after one-or two-level anterior cervical discectomy and fusion". In: *Seminars in Spine Surgery*. Elsevier. 2021, p. 100874.

[8] Huan Song et al. "Attend and diagnose: Clinical time series analysis using attention models". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 32. 1. 2018.

[9] Swaraj Khadanga et al. "Using clinical notes with time series data for ICU management". In: *arXiv preprint arXiv:1909.09702* (2019).

[10] Yanbo Xu et al. "Raim: Recurrent attentive and intensive model of multimodal patient monitoring data". In: *Proceedings of the 24th ACM SIGKDD international conference on Knowledge Discovery & Data Mining*. 2018, pp. 2565–2573.

[11] Hamed M Zolbanin et al. "Data analytics for the sustainable use of resources in hospitals: Predicting the length of stay for patients with chronic diseases". In: *Information & Management* (2020), p. 103282.

[12] Jingyi Wu et al. "Development of a scoring tool for predicting prolonged length of hospital stay in peritoneal dialysis patients through data mining". In: *Annals of Translational Medicine* 8.21 (2020).

[13] Ayman Alahmar, Emad Mohammed, and Rachid Benlamri. "Application of data mining techniques to predict the length of stay of hospitalized patients with diabetes". In: *2018 4th International Conference on Big Data Innovations and Applications (Innovate-Data)*. IEEE. 2018, pp. 38–43.

[14] Segen's Medical Dictionary. *length of stay*. 2011. URL: https://medical-dictionary.thefreedictionary.com/length+of+stay (visited on 03/03/2021).

[15] Alvin Rajkomar et al. "Scalable and accurate deep learning with electronic health records". In: *NPJ Digital Medicine* 1.1 (2018), pp. 1–10.

[16] Tiago Alves et al. "Dynamic prediction of icu mortality risk using domain adaptation". In: *2018 IEEE International Conference on Big Data (Big Data)*. IEEE. 2018, pp. 1328–1336.

[17] CP Launay et al. "Predicting prolonged length of hospital stay in older emergency department users: use of a novel analysis method, the Artificial Neural Network". In: *European journal of internal medicine* 26.7 (2015), pp. 478–482.

[18] Enrique Casalino et al. "Predictive factors for longer length of stay in an emergency department: a prospective multicentre study evaluating the impact of age, patient's clinical acuity and complexity, and care pathways". In: *Emergency Medicine Journal* 31.5 (2014), pp. 361–368.

[19] Whitney E Muhlestein et al. "Predicting inpatient length of stay after brain tumor surgery: Developing machine learning ensembles to improve predictive performance". In: *Neurosurgery* 85.3 (2019), pp. 384–393.

[20] Ru Ding et al. "Predicting emergency department length of stay using quantile regression". In: *2009 International Conference on Management and Service Science*. IEEE. 2009, pp. 1–4.

[21] Pei-Fang Jennifer Tsai et al. "Length of hospital stay prediction at the admission stage for cardiology patients using artificial neural network". In: *Journal of healthcare engineering* 2016 (2016).

[22] Harini Suresh, Jen J Gong, and John V Guttag. "Learning tasks for multitask learning: Heterogenous patient populations in the icu". In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2018, pp. 802–810.

[23] Abolfazl Farahani et al. "A brief review of domain adaptation". In: *Advances in data science and information engineering* (2021), pp. 877–894.

[24] Xiaojie Jin et al. "Collaborative layer-wise discriminative learning in deep neural networks". In: *European Conference on Computer Vision*. Springer. 2016, pp. 733–749.

[25] Bernd Rechel, Antonio Duran, and Richard Saltman. "What is the experience of decentralized hospital governance in Europe". In: (2018).

[26] Tom J Pollard et al. "The eICU Collaborative Research Database, a freely available multi-center database for critical care research". In: *Scientific data* 5.1 (2018), pp. 1–13.

[27] Alistair Johnson et al. *MIMIC-IV-ED (version 1.0)*. 2021. DOI: 10.13026/77Z6-9W59. URL: https://physionet.org/content/mimic-iv-ed/1.0/.

[28] Ary L Goldberger et al. "PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals". In: *circulation* 101.23 (2000), e215–e220.

[29] Sanjay Purushotham et al. "Benchmarking deep learning models on large healthcare datasets". In: *Journal of biomedical informatics* 83 (2018), pp. 112–134.

[30] Zhengping Che et al. "Recurrent neural networks for multivariate time series with missing values". In: *Scientific reports* 8.1 (2018), pp. 1–12.

[31] Zachary C Lipton, David C Kale, Randall Wetzel, et al. "Modeling missing data in clinical time series with rnns". In: *Machine Learning for Healthcare* 56 (2016).

[32] Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. "Learning to forget: Continual prediction with LSTM". In: *Neural computation* 12.10 (2000), pp. 2451–2471.

[33] Sepp Hochreiter and Jürgen Schmidhuber. "Long short-term memory". In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[34] AJ Robinson and Frank Fallside. *The utility driven dynamic error propagation network*. University of Cambridge Department of Engineering Cambridge, 1987.

[35] Paul J Werbos. "Generalization of backpropagation with application to a recurrent gas market model". In: *Neural networks* 1.4 (1988), pp. 339–356.

[36] MarcAurelio Ranzato et al. "Video (language) modeling: a baseline for generative models of natural videos". In: *arXiv preprint arXiv:1412.6604* (2014).

[37] Yong Yu et al. "A review of recurrent neural networks: LSTM cells and network architectures". In: *Neural computation* 31.7 (2019), pp. 1235–1270.

[38] Rafal Jozefowicz, Wojciech Zaremba, and Ilya Sutskever. "An empirical exploration of recurrent network architectures". In: *International conference on machine learning*. PMLR. 2015, pp. 2342–2350.

[39] Francois Chollet et al. *Keras*. 2015. URL: https://github.com/fchollet/keras.

[40] Martıén Abadi et al. “Tensorflow: Large-scale machine learning on heterogeneous distributed systems”. In: *arXiv preprint arXiv:1603.04467* (2016).

[41] Jason Yosinski et al. “How transferable are features in deep neural networks?” In: *Advances in neural information processing systems* 27 (2014).

[42] Edward Choi et al. “Doctor AI: Predicting Clinical Events via Recurrent Neural Networks”. In: *Proceedings of the 1st Machine Learning for Healthcare Conference*. Ed. by Finale Doshi-Velez et al. Vol. 56. Proceedings of Machine Learning Research. Northeastern University, Boston, MA, USA: PMLR, 2016, pp. 301–318. URL: https://proceedings.mlr.press/v56/Choi16.html.

[43] Ji Young Lee, Franck Dernoncourt, and Peter Szolovits. “Transfer learning for named-entity recognition with neural networks”. In: *arXiv preprint arXiv:1705.06273* (2017).

[44] Priyanka Gupta et al. “Transfer learning for clinical time series analysis using deep neural networks”. In: *Journal of Healthcare Informatics Research* 4.2 (2020), pp. 112–137.

[45] Xavier Glorot and Yoshua Bengio. “Understanding the difficulty of training deep feedforward neural networks”. In: *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings. 2010, pp. 249–256.

[46] Jeremy Howard and Sebastian Ruder. “Universal language model fine-tuning for text classification”. In: *arXiv preprint arXiv:1801.06146* (2018).

[47] Gabriel Erion et al. “Improving performance of deep learning models with axiomatic attribution priors and expected gradients”. In: *Nature machine intelligence* 3.7 (2021), pp. 620–631.

[48] Tom O’Malley et al. “Keras Tuner”. In: *Github.[(accessed on 31 January 2021)]* (2019).

[49] Tom O’Malley et al. *KerasTuner*. https://github.com/keras-team/keras-tuner. 2019.

[50] Hervé Abdi and Lynne J Williams. “Tukey’s honestly significant difference (HSD) test”. In: *Encyclopedia of research design* 3.1 (2010), pp. 1–5.

[51] Paul Lillrank, P Johan Groop, and Tomi J Malmström. “Demand and supply–based operating modes—a framework for analyzing health care service production”. In: *The Milbank Quarterly* 88.4 (2010), pp. 595–615.

[52] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. “Practical bayesian optimization of machine learning algorithms”. In: *Advances in neural information processing systems* 25 (2012).