A Multi-task Learning Approach for Predicting Spatio-temporal Patient Variables

Kaniz Fatema Madhobi
kanizfatema.madhobi@wsu.edu
School of Electrical Engineering and Computer Science, Washington State University
Pullman, WA, USA

Eric Lofgren
eric.lofgren@wsu.edu
Paul G. Allen School for Global Health, Washington State University
Pullman, WA, USA

Ananth Kalyanaraman
ananth@wsu.edu
School of Electrical Engineering and Computer Science, Washington State University
Pullman, WA, USA

Abstract

Predicting a patient's length of stay (LOS) or the units they are likely to visit during the course of the stay can be a vital source of information for healthcare administrators towards effective resource planning. However, predicting these parameters can be challenging due to the lack of sufficient information at admission time, and its potential dependence on inherent practices within the hospital. Prior efforts have focused predominantly on predicting LOS, statically at admission and in isolation. In this paper, we propose an adaptive multi-task learning approach to predict a patient's next unit and the expected length (in days) of the remaining stay. Our approach is capable of capturing any latent relationship that may exist between these two variables. Experimental results on a large real-world in-patient database show that our multi-task model outperforms its single-task counterpart and other classical machine learning models. Our study also demonstrates that: a) it is possible to achieve high prediction scores (e.g., mean absolute error of 2.0 days for remaining LOS, and over 80% accuracy for next unit); and b) such high prediction accuracy can be realized early on—in most cases within the first *two* days of a patient's stay.

CCS Concepts

• Applied computing → Health informatics; **• Computing methodologies → Multi-task learning**.

Keywords

multi-task learning, patient length-of-stay, patient unit prediction, machine learning, electronic health records

ACM Reference Format:
Kaniz Fatema Madhobi, Eric Lofgren, and Ananth Kalyanaraman. 2024. A Multi-task Learning Approach for Predicting Spatio-temporal Patient Variables. In *15th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics (BCB '24), November 22–25, 2024, Shenzhen, China.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3698587.3701340

1 Introduction

Predicting *how long* a patient will stay in the hospital (i.e., time) and *where* within the hospital a patient will spend that time (i.e., space) are two important decision variables that could help a hospital administrator make effective resource planning and allocation decisions. In particular, these predictions can help in hospital bed management and staffing decisions [6, 27], real-time discharge prioritization [8], and estimation of the overall patient flow toward better patient outcomes [31].

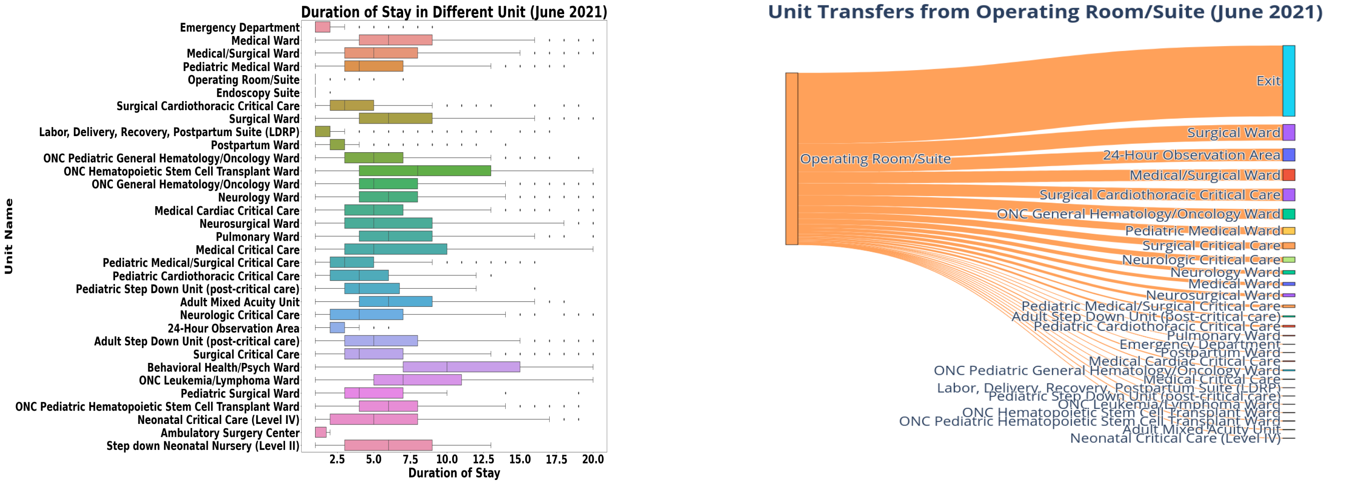
Prior works in prediction have largely focused on predicting the Length of Stay (LOS). Earlier efforts used arithmetic models such as mean or median value as proxy measure for LOS[18, 27, 31]. But since LOS can vary greatly depending on a patient's condition, simpler approaches that rely on mean/median measures are inadequate. Therefore alternative approaches using statistical methods (e.g., linear regression) have been used in several studies [19]. Recent advancement in machine learning and more specifically, deep learning methods have provided a new class of prediction approaches. These approaches have the capability of providing better prediction performance because of their inherent ability to capture non-linear and complex relationships within the data [24, 31]. However these approaches also need large amounts of data for training. Fortunately, with the pervasive adoption of Electronic Health Records (EHRs) in healthcare systems, deep learning approaches are being increasingly used for personalized clinical predictions [30–32].

Despite increasing data availability, LOS prediction still remains challenging for several reasons: First, for these predictions to be valuable, they need to be both accurate enough *and* made available early enough during a patient's stay in a hospital. The desirable thresholds for accuracy or for the timing of availability could vary relative to the overall duration of a patient's stay or the complexity of treatments (and thereby its implications on resources). The higher the prediction accuracy and the earlier such accurate prediction is made available, the more valuable it is for the hospital administrators. However, accurately predicting LOS during the early phases of a patient' admission is challenging due to potential inadequate information available at that phase [13].

Secondly, a patient's LOS (*time*) may not be just a function of the patient's health condition (diseased or pre-existing). There is evidence to suggest that LOS may also be impacted by the *space*, i.e., the hospital's environment in which the patient is treated during the stay. For instance, in a preliminary study with a large hospital in-patient database—the Duke Antimicrobial Stewardship Outreach

BCB '24, November 22–25, 2024, Shenzhen, China

ACM ISBN 979-8-4007-1302-6/24/11
https://doi.org/10.1145/3698587.3701340



(a) LOS vs. units

(b) Unit transfers from Operating room/suite

Figure 1: Part (a) shows the variability in Length of Stay (LOS) for admissions in different units. Secondly, as a patient can visit multiple units during the stay, part (b) shows the variability in unit-to-unit transfer rates. Transfer probabilities are shown from Operating Room/Suite as the source unit and all other units as destination units. The thickness of an arc indicates the probability of that transfer. All results were generated for patients admitted in the month of June 2021 in one of the hospitals (unidentified) in the Duke Antimicrobial Stewardship Outreach Network (DASON) clinical database.

Network (DASON) clinical database [12]—we observed significant LOS divergence among hospital units (Figure 1a).

Furthermore, a single patient may transfer in and out of units and could therefore spend the duration at potentially *multiple* units during the stay. And such transfer rates are not necessarily uniform. For instance, as shown in the plot of Figure 1b, if a patient is currently in the Operating Room/Suite unit, then chances are higher that the patient would be either discharged (exit) or transferred to surgical ward, with transfers to other units also probable. This makes the predictability of the next unit (and thereby its implication on remaining days of LOS) more challenging to predict. But if the patient is at a different ward like one of the maternity wards (not shown in the plot), subsequent transfers become more predictable.

The above observations emphasize the need to not treat LOS prediction in isolation. Instead it becomes important to predict both LOS as well as the next likely unit *in tandem* during the course of a patient's stay. The limitation of all prior works in this research space is that they view LOS prediction in isolation. Furthermore, there exist *no* work in predicting the next unit for a patient to the best of our knowledge. In fact, a majority of prior works study patients within a specific facility or patients having a particular disease condition[3, 6]—thereby limiting the scope of prediction. While this potentially simplifies the model settings toward easier prediction for specific scenarios, it also means facility- or disease-specific models (where data are available) with little chance of success when used in more generalized settings.

1.1 Contributions

The main contribution of this paper is a new multi-task learning model to dynamically predict a patient's unit for the next time step (henceforth, we refer to this as the "next unit") and the remaining length of stay (henceforth we refer to this as "remaining LOS"). Multi-task learning represent a class of deep learning approaches that learn to predict multiple (two or more) variables (or tasks) simultaneously. Studies have shown improved model performance when multiple related tasks are learned together as well as less sensitivity to data noise[28, 29], including in clinical settings [14, 17, 33]. To the best of our knowledge, our work represents the first in predicting these two related outcomes together. Note that these two output variables (next unit and remaining LOS) correspond to two different types of prediction tasks—i.e., next unit is a label and is therefore a classification task, whereas remaining LOS (in days) is a regression task.

From a methodological standpoint, our model combines an artificial neural network (ANN) in addition to a recurrent neural network (RNN). The ANN takes input from the discrete features of a patient available at the time of prediction, whereas the RNN handles the unit transfer sequence during the patient's stay at the hospital. Our model is generic in that it is a hospital-wide trained model that can be used for predicting the two tasks for all types of patients admitted in any unit of the hospital. This generality also means that the prediction for units or patient classes with less amount of data can benefit from other units or patient classes where more data are available.

We evaluated our model on a large real-world in-patient data set containing 255,389 admission records that span over six years (January 2016 through December 2021), from the DASON database [12]. Among all the past related works reported, this is one of the largest in-patient data that has been used in training for LOS. Experimental results show that our new multi-tasking model outperforms its single task counterpart as well as classic machine learning models such as Random Forest, XGBoost and K-nearest neighbor. Our model achieves high prediction scores (e.g., mean absolute error around 2.1 in days and accuracy of 80% or above) on predicting a patient's LOS and next unit respectively; and b) with sufficient data this high prediction accuracy can be realized early on—in most cases within the first *two* days of a patient's stay.

Our model is publicly available for broader community use at https://github.com/madhobi/multitask_unit_and_days. The model is intended to be used in real-time (i.e., on a daily basis) by a hospital administrator, for any current patient admitted at the hospital. With each passing day, the latest information on the unit sequence seen so far as well as all patient attributes (demographics, disease codes, medications) are input to the model, and the output is the prediction of the two variables (next unit and remaining LOS). This approach makes the model adaptive in nature.

The rest of the paper is organized as follows. Section 2 provides a brief overview of the relevant prior works. Section 3 presents the model design and approach. Section 4 presents the detailed experimental evaluation and results of our approach. Finally, Section 5 concludes the paper with a discussion of future research directions.

2 Related Work

There has been an extensive body of works to predict Length of Stay (LOS), as reviewed in [31]. These include both statistical and machine learning approaches. However, a majority of these works are for patients with a specific disease condition or admitted into a specific hospital unit [6].

On predicting LOS as a classification task, ensemble methods have shown significant promise [6, 19, 31]. Some of these efforts predict LOS as either as a short stay (< 7 days) or a long stay (>= 7 days) [4, 10]. Several past efforts also focus on specific diseases or conditions. For instance, the work by Alsinglawi *et al.* is for patients with lung cancer [4]; whereas Chrusciel *et al.* use the unstructured written clinical notes from the Emergency Department admissions [10]. Both studies note best results with Random Forests (RF).

Several studies also exist that predict LOS as a continuous variable. Baek *et al.* [7] use an exploratory data analysis approach that internally uses linear regression to find statistically significant variables that are associated with LOS. Ricciardi *et al.* [26] explored multiple machine learning models (RF, multiple linear regression, radial bias framework, Support Vector Machine (SVM)) for a data set of patients with femur fracture. Bacchi *et al.* [5] use natural language processing with neural networks (artificial and Convolutional neural nets) on clinical notes in general medical unit and acute medical unit. Their approach combines free text and structured data to improve LOS prediction results. Kadri *et al.* [16] developed a Generative Adversarial Network (GAN) based model for patients in pediatric Emergency Department.

Approaches that explored the use of multi-task learning for predicting healthcare related outcomes are relatively recent. Rasmy *et al.* [25] present a deep learning model to predict three outcomes including in-hospital mortality, need for mechanical ventilation and prolonged hospital stay (more than a week). A deep attention based model was proposed by Harerimana *et al.* [13] to predict LOS and in-hospital mortality at admission time. But in this work, it is noted that predictions made in early phase of admission can result in distorted prediction due to lack of necessary information. A Long Short-Term Memory (LSTM) based multi-task learning model was proposed by Ali *et al.* [3] using physical activity sensory data. This work uses sensor information as time series data and predicts two outcomes—LOS continuous outcome, and patients readmission as binary outcome. But their sample set consists of only 47 patients which is not enough to draw a conclusion for broader group of patients.

Common limitations of the existing works are lack of generalizability and constrained sample set. Since most of these models are specific to particular patient groups or hospital ward settings, the models are not necessarily transferable. Furthermore, none of the previous works are for predicting a patient's unit transfer sequence. As noted in Figure 1a, the unit information where a patient is staying is likely to also influence the length of stay. The model proposed in this paper (described in Section 3), overcomes these challenges. We propose a multi-task learning model that internally uses a recurrent neural network to predict a patient's next unit label (for day $i + 1$) as well as the patient's remaining LOS (measured in days) at the hospital.

3 Methodology

In this section, we discuss about the problem modeling and our solution approach. We start with our analysis and observations while creating the inputs and outputs from EHR data, and subsequently present our model design.

3.1 Data Preparation

Our EHR data source for this work is the DASON database [12], which comprises of a large collection of patient admission records for several years, across the Southeastern US regional hospital network. While tracking inpatient records, we encountered two types of data. There are some patient-specific attributes that remain *static* throughout a patient's stay (e.g., age, gender, service requested at admission time). Meanwhile, some attributes are *temporal*, i.e., changing over the time of a given patient's stay that encompasses daily data on the patient's treatment including the units they visit, medications taken, procedures underwent, etc.

On preparation of the input features, we experimented with a number of approaches. We experimented by creating input data sets where every variable is a discrete feature. Classic machine learning models and also artificial neural network models need the data to be in this format and this gave us a baseline prediction measure. To bring the time aspect, we later shifted our approach to sequence modeling. We experimented with a number variants on building sequence features to capture the effect of temporality in our prediction tasks. We first generated sequences by putting all the information of a patient in a sequence. Basically, we took the patient's unit transfer history, the list of medications along with their route categories (i.e. oral or intravenous or rectal etc.), the disease codes and other static attributes of the patient and append them all-together. But this resulted in very long sequences and processing long sequences using RNN can often lead to declined performance [23] as we also noticed in our evaluation. Furthermore, we observed that the unit transfer sequence helped in gaining improved performance on both of the prediction tasks. On the other hand, for the medications history, only keeping the records of current day medications suffices to generate similar performance as keeping the whole sequence of medications from the initial day. This led us to prepare a multi-modal input sets where one set contains the discrete features for each day and another set contains the sequence of units visited.

Since a patient can change multiple units in a day, this made our time step to be of variable length. It starts from 0 when a patient gets admitted, and in default settings, it is increased each day of patients stay (a "patient day" starts from 12.00am and ends at 11.59pm of that same day). Time step also increases if a patient moves from one unit to another within a day.

On the event of a time step increase, all the information of the patient gets updated and the model generates revised predictions based on the current information. The model continues to dynamically adjust the predictions until the patient gets discharged.

Problem formulation: Specifically, our problem has the following input-output requirements.

The input has the following features:

- *Discrete features:* These are the discrete patient information on each time step that includes the demographics (age and gender), comorbidity index, admission service name, disease diagnostic code, diagnosis related group (DRG) weight, daily medications and the medication routes.
- *Sequence feature:* This include the sequence of units (each identified by a unit label) visited by a patient on each day of their stay.

Section 3.2 describes the input data in more detail and Section 3.3 describes feature engineering.

Using the above input features, our prediction problem is defined as follows. For each patient, we accumulate the information available till time step i and predict two variables for the next time step $(i + 1)$:

- (*Next unit*) the most likely next unit label that could also include the current unit at i, which is a classification task; and
- (*Remaining LOS)* the remaining length of stay, which is a regression task.

3.2 Data Statistics

Our data set contains de-identified electronic health records from one academic medical center in the Southeastern United States. We extracted the admission records between January 2016 to December 2021 from Duke Antimicrobial Stewardship Outreach Network (DASON) database and Duke Health System [12, 22]. There are 34 hospital units and 255,389 admission records that span six years (January 2016 through December 2021). We partitioned the dataset into pre-COVID (2016-2019) and post-COVID (2020-2021) phases to account for the impact of the pandemic. Table 1 shows some basic statistics of our data set. We also analyzed the distribution of LOS values based on units (shown in Figure 1a) or patient types. Figure 2 shows the distribution of LOS values based on whether it is an adult or pediatric patient. The plots show a higher variability by the unit, whereas the distribution pattern is similar across adult and pediatric patients with the LOS values occupying to wider range (longer tail) with adults.

Table 1: Input statistics for the DASON data set. (For more details, please see Supplementary Table 1.)

Attributes		Adult (ages 19-90)	Pediatric (ages 0-18)
No. admissions		209,694	45,695
Gender	Female	113,152	22,228
	Male	96,542	23,467
LOS (in days)	Min	1	1
	Max	20	20
	Mean	6	4
	Median	5	3
No. units visited per admission	Min	1	1
	Max	7	6
	Mean	2	2
	Median	2	2

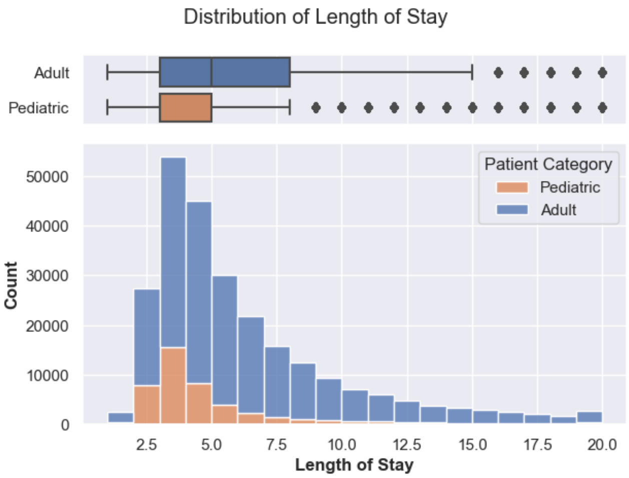
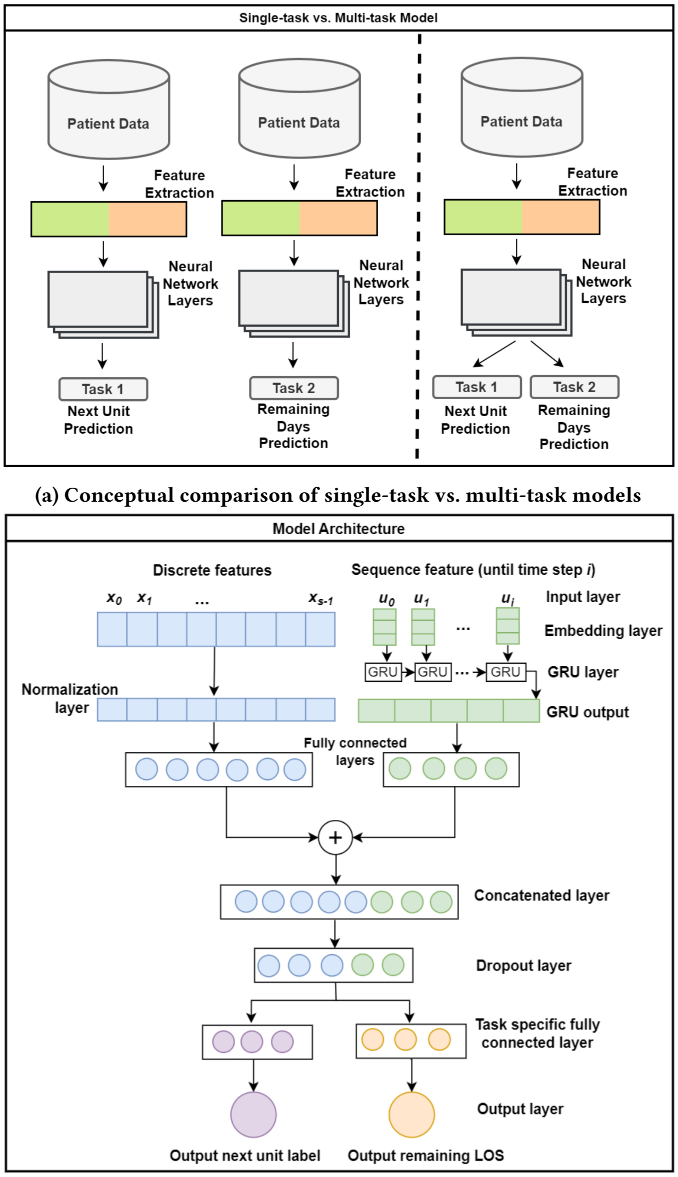


Figure 2: Distributions of Length of Stay for adult and pediatric patients

3.3 Feature Engineering

We tried to keep data filtering at minimum for generalizability. In this study, we consider only "in-patient" admission records, i.e. patients who stayed in the hospital for at least one day. We set the cutoff LOS value to 20 days as there are less than 5% records of patients who stayed longer than that.

Most of the features in our data are categorical values e.g., patient demographics (gender and race), medicine id, medicine route, admission service, unit labels and dignostic codes. The numerical features include patients age, Elixhauser comorbidity score, DRG weight [1] and number of diagnoses. We binned patient's age into categories. The age range for pediatric patients is 0 to 18, which was binned as: $[0, 1)$ for newborn, $[1, 12]$ for children, and $(12, 18]$ for teens. The adult patient ages ranged from 18 to 90, which were binned into seven categories identified by age intervals: $(18, 30]$,$(30, 40]$,$(40, 50]$,$(50, 60]$,$(60, 70]$,$(70, 80]$,and $(80, 90]$. The diagnostic codes are in the form of tenth international classification of diseases (ICD-10). There are 27,738 distinct diagnostic codes in the dataset. One-hot-encoding for this variable is not feasible as it results in extremely sparse data. We labeled the codes into 24 super categories following the standard ICD-10 derivation [2].



(a) Conceptual comparison of single-task vs. multi-task models

(b) Proposed model architecture

Figure 3: A high-level conceptual comparison of two single-task models vs one multi-task model is illustrated in 3a. The overall architecture of our proposed model is shown in 3b.

3.4 Our Multi-task Model for Predicting the Next Unit and Remaining LOS

We used gated recurrent neural network (GRU) based deep learning model that has been proven to be effective in recent research literature for learning from temporal relationship [11]. Our input layer consists of two parts: (i) the discrete features, and (ii) the sequence feature with the unit labels.

For each time step i, we denote the set of s discrete features associated with each patient record as a vector $\mathbf{x^i} : [x_0, x_2, \ldots x_{s-1}]$. As for the temporal feature, the unit labels until that time step is cumulatively appended and we denote this as $\mathbf{u^i} : [u_0, u_1, u_2, \ldots u_i]$. For the purpose of training, this implies that any admission record that has ℓ time steps in total, will contribute ℓ instances of input as $\{x^i, u^i\}$. We use n to denote the total number of admission records in the training data, and the corresponding discrete and sequence feature set as X_n and U_n respectively.

Model design and architecture: Given the training set of $\{X_n, U_n\}$ for n admissions, the goal is to train a neural network so that it learns how to predict the two desired output variables, namely the next unit and remaining LOS for a patient. One approach is to view this as two separate prediction tasks, which we refer to as the single task setting. However, as observed earlier, the unit prediction problem is also expected to be related to LOS prediction. Therefore, in our approach, we treat this as a multi-task learning problem. Figure 3a illustrates the conceptual difference between these two approaches.

In Figure 3b, we show the detailed view of our multi-task learning model.

The discrete features ($\mathbf{x}$) are passed to a normalization layer and then they are fed to a fully connected layer.

The temporal features ($\mathbf{u^i}$), on the other hand, get accumulated at each time step and go through an embedding layer. The embedding layer is a deep learning alternative to label encoding and have been used in many clinical studies involving health records[9, 20, 21]. The outputs from the embedding layer are fed to a sequence of GRU layers, which in turn passes its output to another fully connected layer. Another alternative architecture for sequence modeling is Long Short-Term Memory (LSTM) which is computationally more expensive than GRU. It has been shown in study that the performance of GRU vs LSTM depends on particular dataset and use cases [11, 15]. However, we experimented with both of them and decided to use GRU because of the better prediction performance and less training time in our dataset. The outputs from the dense layers are concatenated and further passed to task specific dense layers. In our experiments (Section 4), we evaluated and compared both the single-task and multi-task models. We will see that the results show improved performance when these two tasks are learned together under the multi-task learning framework.

3.5 Software Availability

The source code of our model implementation is publicly available as open source at the GitHub site: https://github.com/madhobi/multitask_unit_and_days for the broader research community to evaluate and use our implementations. Note that we cannot make the data sets from the DASON public due to privacy restrictions. To ensure code use, we have provided toy example training inputs.

4 Results and Discussions

In this section, we present the experimental results of our proposed RNN-based multi-task model ("RNN-mtl") and compare it against its single-task component ("RNN-stl") and other classical machine learning/deep learning models. More specifically, we compare against Random Forest (RF), K-nearest neighbor (KNN) and XGBoost.

We also compare the results of the multitask model with its equivalent single task models having the same configurations for hidden layers and drop out set up. First, we will discuss the experimental setup and then we will move our discussion to the evaluation of the prediction tasks i.e. predicting days remaining (regression) and predicting next unit (classification).

We have six years of hospital data containing 255,389 inpatient admission records in total. The timeline of the records is from January 2016 to December 2021. In light of the COVID-19 pandemic and the distinct patterns of unit visits observed for adult and pediatric patients, we partitioned our dataset into four subsets: {PreCOVID, PostCOVID} * {Adult, Pediatric}. Each of the following four data sets were split into training (70%) and testing (30%) sets during the model development:

- Dataset-1 (*Pre-COVID, Adult*): Contains the hospital records between January 2016 to December 2019 for adult patients (with age $>$ 18 years) .
- Dataset-2 (*Post-COVID, Adult*): Contains the hospital records between January 2020 and December 2021 for adult patients (with age $>$ 18 years).
- Dataset-3 (*Pre-COVID, Pediatric*): Contains the hospital records between January 2016 to December 2019 for pediatric patients (with age $\leq$ 18 years).
- Dataset-4 (*Post-COVID, Pediatric*): Contains the hospital records between January 2020 and December 2021 for pediatric patients (with age $\leq$ 18 years).

Test platform: In all our testing, we used a compute node which has an Intel(R) Xeon(R) Platinum 8175M CPU, running at 2.50GHz (8 cores, 16 threads), and with 64GB RAM. The software environment consisted of Ubuntu 20.04 operating system, Python 3.10, TensorFlow 2.11.0, and other libraries such as NumPy, Keras, and Pandas. Our total training time on this platform took approximately 5 hours.

4.1 Evaluation Metrics

The task of predicting days remaining is a regression task. For this task, we set the evaluation metrics as Mean Absolute Error (MAE), Mean Squared Error (MSE) and Root Mean Squared Error (RMSE). On the other hand, the task of predicting next unit is a classification task and for this task we recorded the exact accuracy score and top-k accuracy score setting k to 2–i.e., computing the number of times the correct unit label appears in the top two predictions of our output. We also recorded the precision, recall and F1-score to evaluate each model's performance. Since unit prediction is a multi-class classification problem and also, the target class has an uneven distribution of observations, we calculated the weighted average values for these metrics. To calculate the weighted average precision, we multiply the precision of each label and multiply them with their sample size and divide it by the total number of samples in the dataset. Similarly, we calculate the weighted average values for recall and F1-score. The formulas for calculating these values are presented in equations 1-6.

$$\text{Precision} = \frac{Tp}{Tp + Fp} \tag{1}$$

$$\text{WeightedAveragePrecision} = \frac{\sum_{i=1}^{n} |y_i| \frac{Tp_i}{Tp_i+Fp_i}}{\sum_{i=1}^{n} |y_i|} \tag{2}$$

$$\text{Recall} = \frac{Tp}{Tp + Fn} \tag{3}$$

$$\text{WeightedAverageRecall} = \frac{\sum_{i=1}^{n} |y_i| \frac{Tp_i}{Tp_i+Fn_i}}{\sum_{i=1}^{n} |y_i|} \tag{4}$$

$$\text{F1-score} = \frac{2 * \text{ Precision } * \text{ Recall}}{\text{Precision } + \text{ Recall}} \tag{5}$$

$$\text{WeightedAverageF1-score} = \frac{\sum_{i=1}^{n} |y_i| \, F1\text{-score}}{|y_i|} \tag{6}$$

In all our results, our main approach is the *RNN (Multi-task)* model. We compared it against classic machine learning models including RF, XGBoost, and KNN as well as the single-task version of the RNN model.

4.2 Results for Predicting Remaining LOS

Tables 2 and 3 show the results of prediction accuracy for predicting the remaining days of LOS, for adult and pediatric patients respectively. Figure 4 can give a glance at the results which is drawn from the post-COVID adult data set. The key observations are as follows.

- The error profiles of the models varied between adult and pediatric patients. In general, error values generated by models are slightly lower on the pediatric patients data, suggesting better predictability for pediatric patients.
- For the adult patients, our proposed multi-task learning model consistently outperforms all other models, over both Pre-COVID and Post-COVID data sets. Among the other models, XGBoost and our RNN-based Single-task models perform comparably while their errors were still larger than our default RNN Multi-task model.
- For the pediatric patients, the models performed better than the adult data sets and the MAE value generated is around $1.8 - 1.9$ days for almost all models. Although XGBoost performed slightly better than our model in the Pre-COVID data set, for Post-COVID data, RNN Multi-task model performed the best.
- We also observe that the RNN Multi-task results are consistently better than their respective RNN Single-task results, suggesting the value of a shared model for the two prediction tasks.
- Overall, our RNN Multi-task model yielded a Mean Absolute Error of around 2 days on all inputs. This implies regardless of the unit or the type of patient, we were able to predict the remaining LOS within approximately ±2 days of the actual. Later we breakdown this error by the number of days since admission for a more detailed analysis.

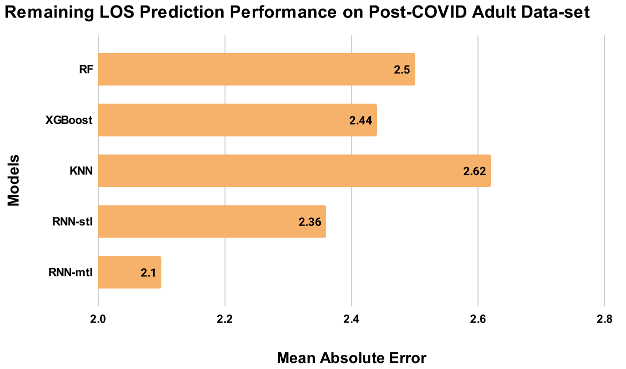


Figure 4: Performance comparison of different models on prediction of remaining LOS. The mean absolute has been generated from post-COVID test data set of adult patients.

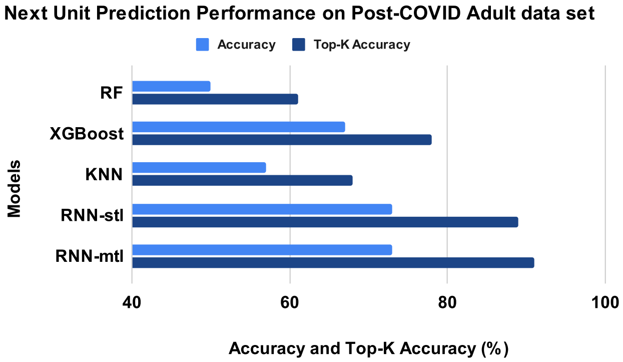


Figure 5: Performance comparison of different models on prediction of next unit label. The evaluation metrics are drawn from post-COVID test data set of adult patients.

4.3 Results for Predicting Next Unit

Next, we predict the unit label for a patient for the next day. Since unit is a label out of a total 34 valid unit labels (and hence a classification task), we computed the accuracy score for an exact match, as well as the accuracy score for the top-2 predictions—i.e., whether the correct unit label appears in the top two predictions of our output. Tables 4 and 3 summarize the results of our prediction accuracies for predicting the next unit, for adult and pediatric patients respectively. Barplots in Figure 5 shows the comparison of different models performance on post-COVID adult data set, with additional ROC curves in Supplementary Figure S1. The key observations are as follows.

- First we observe that our RNN Multi-task model provide the best accuracy values across almost all metrics, for both adult and pediatric patients. The second most accurate model was XGBoost.
- Similar to the LOS prediction task, here too we see that the RNN Multi-task results are consistently better than their respective RNN Single-task results—again suggesting the value of a shared model for the two prediction tasks.
- Next, we observe that the prediction accuracy improves significantly (from around 73% to 91% for RNN Multi-task) when we look at the top-k ($k = 2$) predictions in units. A similar behavior is observed not only for our default RNN Multi-task model but also for all other models compared. However, only the RNN Multi-task model exceeds 90% accuracy. These results suggest the value of considering not just the top predicted unit label but also the second most probable unit predicted by the model. Intuitively, this is because there may be related units (e.g., labor and delivery ward, or postpartum ward) and even if the model mispredicts for its top prediction with a related unit, it may capture the correct unit in its second prediction.

Overall the above results on both next unit and remaining LOS predictions demonstrate that the proposed RNN Multi-task learning model is able to outperform other models for a majority of the input cases.

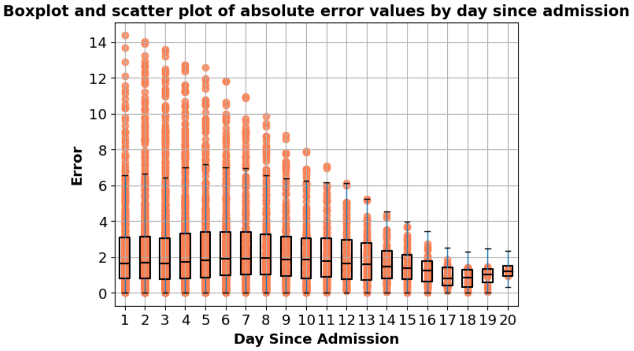


Figure 6: Absolute error values on predicting remaining days of LOS. The X-axis shows the number of days since admission, and Y-axis shows the range of the Mean Absolute Error values. The plot is shown for Post-COVID adult dataset.

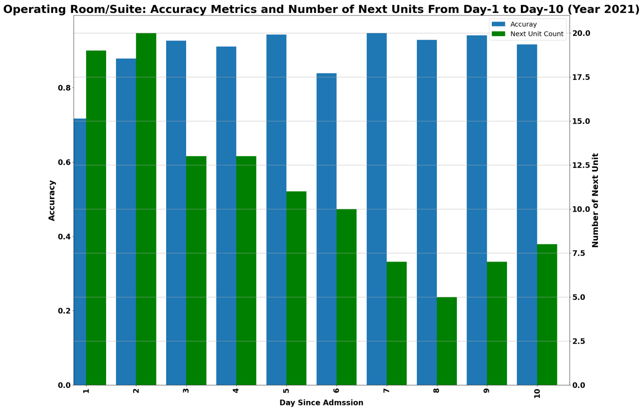


Figure 7: Accuracy on predicting next units from operating room/suite as a function of days since admission. These results shown are for Post-COVID adult dataset.

Table 2: Prediction accuracy, as measured by the different *error* metrics (lower the better), for predicting the remaining days of LOS (regression task) in Adult Patients. Table shows different approaches in each model column. RNN (Multi-task) is our main approach. Bold face entries are the best values reported under each metric.

		Machine Learning Models			Deep Learning Models	
	Error Metrics	Random Forest	XGBoost	KNN	RNN (Single-task)	RNN (Multi-task)
Dataset-1 (Pre-COVID, Adult)	Mean Absolute Error (MAE)	2.48	2.23	2.39	2.22	**2.01**
	Root Mean Squarred Error (RMSE)	3.38	3.06	3.31	3.02	**2.86**
	Mean Squared Error (MSE)	11.41	9.35	10.97	9.13	**8.22**
Dataset-2 (Post-COVID, Adult)	Mean Absolute Error (MAE)	2.5	2.44	2.62	2.36	**2.10**
	Root Mean Squarred Error (RMSE)	3.39	3.3	3.3	3.12	**3.05**
	Mean Squared Error (MSE)	11.49	10.87	10.87	9.76	**9.3**

Table 3: Prediction accuracy, as measured by the different *error* metrics (lower the better), for predicting the remaining days of LOS (regression task) in Pediatric Patients. Table shows different approaches in each model column. RNN (Multi-task) is our main approach. Bold face entries are the best values reported under each metric.

		Machine Learning Models			Deep Learning Models	
	Error Metrics	Random Forest	XGBoost	KNN	RNN (Single-task)	RNN (Multi-task)
Dataset-3 (Pre-COVID, Pediatric)	Mean Absolute Error (MAE)	1.86	**1.82**	1.93	1.95	1.92
	Root Mean Squarred Error (RMSE)	2.82	**2.76**	2.95	2.99	2.95
	Mean Squared Error (MSE)	7.98	**7.61**	8.72	9.13	8.76
Dataset-4 (Post-COVID, Pediatric)	Mean Absolute Error (MAE)	2.0	1.98	2.08	1.92	**1.8**
	Root Mean Squarred Error (RMSE)	3.04	3.01	3.16	3.0	**2.98**
	Mean Squared Error (MSE)	9.23	9.07	10.02	8.95	**8.88**

Table 4: Prediction accuracy, as measured by the different *accuracy* metrics (higher the better), for predicting the next unit (classification task) in Adult Patients. Table shows different approaches in each model column. RNN (Multi-task) is our main approach. Bold face entries are the best values reported under each metric.

	Accuracy Metrics	Machine Learning Models			Deep Learning Models	
		Random Forest	XGBoost	KNN	RNN (Single-task)	RNN (Multi-task)
Dataset-1 (Pre-COVID, Adult)	Accuracy	54%	67%	60%	73%	**74%**
	TopKAccuracy(k=2)	64%	80%	70%	90%	**91%**
	Weighted Avg Precision	74%	**78%**	68%	62%	73%
	Weighted Avg Recall	54%	68%	60%	64%	**75%**
	Weighted Avg F1-score	60%	70%	63%	60%	**71%**
Dataset-2 (Post-COVID, Adult)	Accuracy	50%	67%	57%	**73%**	**73%**
	TopKAccuracy(k=2)	61%	78%	68%	89%	**91%**
	Weighted Avg Precision	75%	**77%**	67%	61%	72%
	Weighted Avg Recall	51%	67%	58%	64%	**74%**
	Weighted Avg F1-score	58%	69%	61%	59%	**70%**

4.4 Further Analysis and Discussions

In what follows we provide a more in-depth look into the predictions by the different units and also as a function of time (i.e., days since admission).

First, we examine the prediction of remaining LOS as a function of the number of days since admission. Prediction of remaining LOS is harder during the initial day, and we expect the prediction to get better as the days progress. Figure 6 shows the results of our analysis. More specifically, the mean values for the absolute error fluctuates between roughly ±2 days initially until about 10 days and then reduces to ±1 subsequently. However, the range of error values are spread over a wider range in the early days of admission and narrow down as the number of days increases—eventually diminishing to a negligible range.

We also studied the dependence of the predictive power for both variables (remaining LOS, and next unit) on the units that the patients were at the time of the prediction. Our analysis in a nutshell showed significant variability of the errors and prediction accuracy over the different units—clearly suggesting that predictive ability depends on which unit a patient is in, at the time of prediction. Some units are more predictable than others. As an example, the model's performance on maternity ward patients (Postpartum and LDRP), is high—with $> 95\%$ for the next unit and < 1 day error for remaining LOS. The largest error values (3.5 days) were seen for the Behavioral Health Ward. In general, the MAE values for days remaining were higher for ICU patients compared to non-ICU patients.

Table 5: Prediction accuracy, as measured by the different *accuracy* metrics (higher the better), for predicting the next unit (classification task) in Pediatric Patients. Table shows different approaches in each model column. RNN (Multi-task) is our main approach. Bold face entries are the best values reported under each metric.

	Accuracy Metrics	Machine Learning Models			Deep Learning Models	
		Random Forest	XGBoost	KNN	RNN (Single-task)	RNN (Multi-task)
Dataset-3 (Pre-COVID, Pediatric)	Accuracy	60%	68%	62%	64%	**73%**
	TopKAccuracy(k=2)	71%	79%	72%	84%	**93%**
	Weighted Avg Precision	74%	**77%**	70%	62%	73%
	Weighted Avg Recall	60%	68%	62%	64%	**73%**
	Weighted Avg F1-score	65%	71%	65%	60%	**72%**
Dataset-4 (Post-COVID, Pediatric)	Accuracy	57%	67%	59%	63%	**73%**
	TopKAccuracy(k=2)	68%	77%	69%	83%	**91%**
	Weighted Avg Precision	74%	**76%**	68%	61%	72%
	Weighted Avg Recall	58%	67%	59%	64%	**74%**
	Weighted Avg F1-score	63%	70%	63%	59%	**70%**

The accuracy for next unit prediction is above 80% for all units apart from the Emergency Department (ED). Emergency Department is distinguishable for various reasons including high volume of patients with diverse conditions. The model's lower prediction score is acceptable for ED as the patients usually stay there temporarily and then move to a condition specific facility. Figure S2 in the Supplementary section shows the detailed spread of error values for remaining LOS (left panel), and the prediction accuracy for the top prediction of the next unit (right panel)—for adult and pediatric units.

To provide a more in-depth analysis of the classification task, we look into the prediction accuracies based on patient's current unit. As a case scenario, we are discussing our observations made on Operating room/suite as this is the unit from where patients get transferred to almost all other units in the hospital. We plotted the accuracy of predicting next unit from the operatin room (OT) in Figure 7. The plot also shows the number of distinct possible next units in the test data set on each day. If a patient visits OT in the first two days of hospital stay, there are around 20 different units in the data set where the patient could get transferred to afterwards. We see an increase in prediction accuracy from around 70% on the first day to above 80% on the second day. If a patient visits OT later on the admission, the model accumulates more information about the patient and can predicts better as we can see from Figure 7, the prediction accuracy going above 90% as the day since admission increases. We observed similar trends in general ward patients. In summary, the model can achieve 80% accuracy or higher as early as 2 days of patients stay, which is a promising result and can be of significant benefit for hospital planning and resource allocation.

5 Conclusions and Future Works

This paper presents a new multi-task learning model to predict patients next unit label along with the remaining length of stay in the hospital. Prediction of these two correlated information can help in devising a comprehensive picture of the hospital resource usage and in turn lead to a better and efficient clinical management system. Future research avenues involve integrating diagnostic codes with medication data and extending the output to generate sequence of most likely units visit coupled with duration of stay.

Acknowledgment

This work was in part supported by the U.S. Centers for Disease Control and Prevention (CDC) under grant number U01CK000673 and by the U.S. National Institutes of Health (NIH) under grant number R35GM147013, and U.S. National Science Foundation (NSF) CBET 2226680.

References

[1] MS-DRG Classifications and Software. cms.gov/medicare/medicare-fee-for-service-payment/acuteinpatientpps/ms-drg-classifications-and-software.

[2] International Classification of Diseases 10th revision (ICD-10). https://www.icd10data.com/ICD10CM/Codes.

[3] Sajid Ali, Shaker El-Sappagh, Farman Ali, Muhammad Imran, and Tamer Abuhmed. Multitask deep learning for cost-effective prediction of patient's length of stay and readmission state using multimodal physical activity sensory data. *IEEE Journal of Biomedical and Health Informatics*, 26(12):5793–5804, 2022.

[4] Belal Alsinglawi, Osama Alshari, Mohammed Alorjani, Omar Mubin, Fady Alnajjar, Mauricio Novoa, and Omar Darwish. An explainable machine learning framework for lung cancer hospital length of stay prediction. *Scientific reports*, 12(1):1–10, 2022.

[5] Stephen Bacchi, Samuel Gluck, Yiran Tan, Ivana Chim, Joy Cheng, Toby Gilbert, David K Menon, Jim Jannes, Timothy Kleinig, and Simon Koblar. Prediction of general medical admission length of stay with natural language processing and deep learning: a pilot study. *Internal and emergency medicine*, 15:989–995, 2020.

[6] Stephen Bacchi, Yiran Tan, Luke Oakden-Rayner, Jim Jannes, Timothy Kleinig, and Simon Koblar. Machine learning in the prediction of medical inpatient length of stay. *Internal medicine journal*, 52(2):176–185, 2022.

[7] Hyunyoung Baek, Minsu Cho, Seok Kim, Hee Hwang, Minseok Song, and Sooyoung Yoo. Analysis of length of hospital stay using electronic health records: A statistical and data mining approach. *PloS one*, 13(4):e0195901, 2018.

[8] Sean Barnes, Eric Hamrock, Matthew Toerper, Sauleh Siddiqui, and Scott Levin. Real-time prediction of inpatient length of stay for discharge prioritization. *Journal of the American Medical Informatics Association*, 23(e1):e2–e10, 2016.

[9] Edward Choi, Cao Xiao, Walter Stewart, and Jimeng Sun. Mime: Multilevel medical embedding of electronic health records for predictive healthcare. *Advances in neural information processing systems*, 31, 2018.

[10] Jan Chrusciel, François Girardon, Lucien Roquette, David Laplanche, Antoine Duclos, and Stéphane Sanchez. The prediction of hospital length of stay using unstructured data. *BMC Medical Informatics and Decision Making*, 21(1):351, 2021.

[11] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[12] Duke University School of Medicine. Duke Antimicrobial Stewardship Outreach Network (DASON), 2019. https://dason.medicine.duke.edu/.

[13] Gaspard Harerimana, Jong Wook Kim, and Beakcheol Jang. A deep attention model to forecast the length of stay and the in-hospital mortality right on admission from icd codes and demographic data. *Journal of Biomedical Informatics*, 118:103778, 2021.

[14] Hrayr Harutyunyan, Hrant Khachatrian, David C Kale, Greg Ver Steeg, and Aram Galstyan. Multitask learning and benchmarking with clinical time series data. *Scientific data*, 6(1):96, 2019.

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

[16] Farid Kadri, Abdelkader Dairi, Fouzi Harrou, and Ying Sun. Towards accurate prediction of patient length of stay at emergency department: A gan-driven deep learning framework. *Journal of Ambient Intelligence and Humanized Computing*, pages 1–15, 2022.

[17] Tae San Kim and So Young Sohn. Multitask learning for health condition identification and remaining useful life prediction: deep convolutional neural network approach. *Journal of Intelligent Manufacturing*, 32:2169–2179, 2021.

[18] Andy H Lee, Wing K Fung, and Bo Fu. Analyzing hospital length of stay: mean or median regression? *Medical care*, pages 681–686, 2003.

[19] Vincent Lequertier, Tao Wang, Julien Fondrevelle, Vincent Augusto, and Antoine Duclos. Hospital length of stay prediction methods: a systematic review. *Medical care*, 59(10):929–938, 2021.

[20] Irene Li, Jessica Pan, Jeremy Goldwasser, Neha Verma, Wai Pan Wong, Muhammed Yavuz Nuzumlalı, Benjamin Rosand, Yixin Li, Matthew Zhang, David Chang, et al. Neural natural language processing for unstructured data in electronic health records: A review. *Computer Science Review*, 46:100511, 2022.

[21] Yikuan Li, Shishir Rao, José Roberto Ayala Solares, Abdelaali Hassaine, Rema Ramakrishnan, Dexter Canoy, Yajie Zhu, Kazem Rahimi, and Gholamreza Salimi-Khorshidi. Behrt: transformer for electronic health records. *Scientific reports*, 10(1):7155, 2020.

[22] Rebekah W Moehring, Michael E Yarrington, Angelina E Davis, April P Dyer, Melissa D Johnson, Travis M Jones, S Shaefer Spires, Deverick J Anderson, Daniel J Sexton, and Elizabeth S Dodds Ashley. Effects of a collaborative, community hospital network for antimicrobial stewardship program implementation. *Clinical Infectious Diseases*, 73(9):1656–1663, 2021.

[23] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In *International conference on machine learning*, pages 1310–1318. Pmlr, 2013.

[24] Alvin Rajkomar, Eyal Oren, Kai Chen, Andrew M Dai, Nissan Hajaj, Michaela Hardt, Peter J Liu, Xiaobing Liu, Jake Marcus, Mimi Sun, et al. Scalable and accurate deep learning with electronic health records. *NPJ digital medicine*, 1(1):18, 2018.

[25] Laila Rasmy, Masayuki Nigo, Bijun Sai Kannadath, Ziqian Xie, Bingyu Mao, Khush Patel, Yujia Zhou, Wanheng Zhang, Angela Ross, Hua Xu, et al. Recurrent neural network models (covrnn) for predicting outcomes of patients with covid-19 on admission to hospital: model development and validation using electronic health record data. *The Lancet Digital Health*, 4(6):e415–e425, 2022.

[26] Carlo Ricciardi, Alfonso Maria Ponsiglione, Arianna Scala, Anna Borrelli, Mario Misasi, Gaetano Romano, Giuseppe Russo, Maria Triassi, and Giovanni Improta. Machine learning and regression analysis to model the length of hospital stay in patients with femur fracture. *Bioengineering*, 9(4):172, 2022.

[27] Gordon H Robinson, Louis E Davis, and Richard P Leifer. Prediction of hospital length of stay. *Health services research*, 1(3):287, 1966.

[28] Sebastian Ruder. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.

[29] Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. *Advances in neural information processing systems*, 31, 2018.

[30] Benjamin Shickel, Patrick James Tighe, Azra Bihorac, and Parisa Rashidi. Deep ehr: a survey of recent advances in deep learning techniques for electronic health record (ehr) analysis. *IEEE journal of biomedical and health informatics*, 22(5):1589–1604, 2017.

[31] Kieran Stone, Reyer Zwiggelaar, Phil Jones, and Neil Mac Parthaláin. A systematic review of the prediction of hospital length of stay: Towards a unified framework. *PLOS Digital Health*, 1(4):e0000017, 2022.

[32] Pranjul Yadav, Michael Steinbach, Vipin Kumar, and Gyorgy Simon. Mining electronic health records (ehrs) a survey. *ACM Computing Surveys (CSUR)*, 50(6):1–40, 2018.

[33] Yu Zhang and Qiang Yang. An overview of multi-task learning. *National Science Review*, 5(1):30–43, 2018.