DYNAMIC PREDICTION OF ICU MORTALITY RISK USING DOMAIN ADAPTATION

TIAGO H. C. ALVES

DYNAMIC PREDICTION OF ICU MORTALITY RISK USING DOMAIN ADAPTATION

Dissertation presented to the Graduate Program in Computer Science of the Universidade Federal de Minas Gerais in partial fulfillment of the requirements for the degree of Master in Computer Science.

Advisor: Adriano Veloso
Co-advisor: Alberto H. F. Laender

Belo Horizonte
February 2018

Alves, Tiago H. C.
A474d Dynamic Prediction of ICU Mortality Risk Using Domain Adaptation / Tiago H. C. Alves. — Belo Horizonte, 2018
xxii, 41 f. : il. ; 29cm

Dissertação (mestrado) — Universidade Federal de Minas Gerais
Orientador: Adriano Veloso
Coorientador: Alberto H. F. Laender

1. Computação - Teses. 2. Aprendizado do computador. 3. Mortalidade. 4. Análise de domínio temporal. I. Orientador. II. Coorientador. III. Título

CDU 519.6*82(043)

UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

Predição dinâmica de risco de mortalidade em UTI usando adaptação de domínio

TIAGO HENRIQUE COSTA ALVES

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

PROF. ADRIANO ALONSO VELOSO - Orientador
Departamento de Ciência da Computação - UFMG

PROF. ALBERTO HENRIQUE FRADE LAENDER - Coorientador
Departamento de Ciência da Computação - UFMG

PROF. CAETANO TRAINA JÚNIOR
Instituto de Ciências Matemáticas e de Computação - USP

DR. JOSÉ MAURO VIEIRA JÚNIOR
Unidade de Terapia Intensiva Adulto - Hospital Sírio-Libanês

PROF. NIVIO ZIVIANI
Departamento de Ciência da Computação - UFMG

Belo Horizonte, 28 de Fevereiro de 2018.

I dedicate this work to all who helped me in this journey, specially to my fiancée Ingrid. Without her, nothing of this would have been possible.

Acknowledgments

I'd like to thank those who gave me strength to move forward, my fiancée Ingrid, my friend Lucas, all of my friends at Kunumi and the company itself. I'd like to thank Nivio, for the opportunities he has given me, and my advisors, for showing me the way, and most of all, UFMG.

"The science of today is the technology of tomorrow"
(Edward Teller)

Resumo

O reconhecimento antecipado de trajetórias de alto risco durante a estadia em uma Unidade de Tratamento Intensivo (UTI) é uma das chaves para aumentar a sobrevivência de pacientes. Aprender tais trajetórias a partir de parâmetros fisiológicos e epidemiológicos medidos continuamente durante uma estadia em UTI requer o aprendizado de *features* temporais que são robustas e discriminativas através de diversas populações de pacientes. Pacientes em populações de UTIs diferentes (ou domínios) podem variar de acordo com a idade, condições e intervenções, e modelos construídos usando dados de pacientes de um domínio de uma UTI particular performam mal em outros domínios, pois as *features* utilizadas para treinar tais modelos possuem distribuições diferentes entre os grupos. Neste trabalho, nós propomos um modelo profundo capaz de capturar e transferir as *features* locais e temporais de dados de UTIs compostos por séries temporais multivariadas. Tais *features* são capturadas de uma forma que o estado do paciente em um determinado tempo dependa do tempo anterior. Isto permite predições dinâmicas e cria um espaço de risco de mortalidade, permitindo uma fácil descrição do risco do paciente em qualquer momento. Um extenso experimento entre UTIs com diversos domínios revelou que nosso modelo supera todos os *baselines* considerados. Os ganhos de AUC vão de 4% a 8% para predições antecipadas, quando comparados com um representativo do estado-da-arte recente para predição de mortalidade em UTI. Nossos experimentos também mostram a importância de aprender modelos que são específicos para cada domínio de UTI. Em particular, modelos para o domínio Cardíaco alcançam valores de AUC tão altos quanto 0.87, mostrando utilidade clínica excelente para predição antecipada de mortalidade.

Abstract

Early recognition of risky trajectories during an Intensive Care Unit (ICU) stay is one of the key steps towards improving patient survival. Learning such trajectories from epidemiological and physiological parameters that are continuously measured during an ICU stay requires learning time-series features that are robust and discriminative across diverse patient populations. Patients within different ICU populations (or domains) may vary by age, conditions and interventions, and models built using patient data from a particular ICU domain perform poorly in other domains because the features used to train such models have different distributions across the groups. In this work, we propose a deep model to capture and transfer complex spatial and temporal features from multivariate time-series ICU data. Features are captured in a way that the state of the patient in a certain time depends on the previous state. This enables dynamically predictions and creates a mortality risk space, allowing to easily describe the risk of the patient at a particular time. A comprehensive cross-ICU experiment with diverse domains reveals that our model outperforms all considered baselines. Gains in terms of AUC range from 4% to 8% for early predictions, when compared with a recent state-of-the-art representative for ICU mortality prediction. Our experiments also show the importance of learning models that are specific for each ICU domain. In particular, models for the Cardiac domain achieve AUC numbers as high as 0.87, showing excellent clinical utility for early mortality prediction.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

The Intensive Care Unit (ICU) is a department of a hospital in which patients who are dangerously ill are kept under constant observation. Usually, those units have a single specialization, such as cardiac surgery or pediatric diseases, and deal with patients who have high mortality risk and, therefore, need to be constantly monitored by means of equipments that control their status on real time (e.g., a heart beat monitor) or by exam results requested by ICU doctors, also called intensivists.

1.1 Motivation

According to Gruenberg et al. [2006], the estimated ICU length of stay in the United States is of 3.8 days and the leading causes of death in the ICU are multi-organ failure, cardiovascular failure and sepsis [Wunsch et al., 2010]. Multi-organ failure has a mortality rate of up to 15-28% and severe respiratory failure has a mortality rate ranging from 20% to 50%, while sepsis has a mortality rate of up to 45%. Overall, mortality rates in patients admitted to adult ICU average from 10% to 29%, depending on age and illness severity.

Data from patients in an ICU are extensive, complex and often produced at a rate far greater than intensivists can absorb. As a consequence, monitoring ICU patients is becoming increasingly complicated and systems that learn from ICU data in order to alert clinicians to the current and future risks of a patient are playing a significant role in the decision making process [McNeill and Bryden, 2013]. However, one of the main barriers in the deployment of these learning systems is the lack of generalization of results, i.e., the learning performance achieved in controlled environments often drops when the models are tested with different patient populations and conditions [Alemayehu and Warner, 2004; Seshamani and Gray, 2004].

This behavior could be explained by the difference between patients' data, since each specific environment has a set of conditions and means to treat the patient, creating thus sub-populations. For instance, a patient admitted on a Cardiac ICU probably has a different status than a patient on a Medical ICU, and both should be observed differently. Also, this could be explained by the reasons that eventually leads to death, observed not only in different hospital domains, but also inside the same hospital, from an ICU to another, and even inside a single ICU. Each patient is different and although there might be some similarity between them, other factors contribute to the outcome variance, such as the designated professional staff, applied treatments and the ICU environment.

1.2 Contributions

In this work, we explore domain adaptation to improve the performance of systems evaluated with mismatched training and testing conditions. We propose deep models that extract the domain-shared and the domain-specific latent features. This enables us to learn multiple models that are specific to each ICU domain, improving prediction accuracy over diverse patient populations. For this, we discuss several domain adaptation approaches that differ in terms of the choice of which layers to freeze or tune.

The proposed models are composed of convolutional and recurrent components. They capture local physiological interactions (e.g., heart rate, creatinine, systolic blood pressure) at the lower level using a Convolutional Neural Network (CNN) [LeCun et al., 1998] and extracts the long range dependencies based on convoluted physiological signals at the higher level using a Long Short-Term Memory network (LSTM) [Hochreiter and Schmidhuber, 1997]. Thus, our models exploit spatial and temporal information within vital signals and laboratorial findings to dynamically predict patient outcomes, i.e., the CNN component extracts spatial features of varying abstract levels and the LSTM component ingests a sequence of spatial features to generate temporally dynamic predictions for patient mortality. As a result, our models perform predictions that are based on information continuously collected over time and that can be dynamically updated as soon as new information becomes available.

While the combination of convolutional and recurrent structures has been investigated in a prior scenario other than that of mortality prediction [Wang and Nyberg, 2015], this architecture is a proper choice here because it offers a complementary spatial-temporal perspective of the patient condition. As a result, predictions based on infor-

mation that are continuously collected over time can be dynamically updated as soon as new information becomes available.

We also propose a novel neural network layer, which we called Switch. This layer is able to create internal dense representations of the patient's features and then use these representations to modify the features themselves. With this modifications, our layer is able to find different distributions along the dataset, identify which distribution the patient belongs to and use that information to improve the prediction.

As a consequence, the learned representations along with the predictions for a specific patient during the ICU stay form the corresponding patient trajectory and, thus, a mortality risk space can be obtained from a set of past patient trajectories. The fundamental benefit of analyzing future patient trajectories in the mortality risk space is the focus on dynamics, emphasizing the proximity to risky regions of the space and the speed in which the patient condition changes. Therefore, the mortality risk space enables clinicians to track risky trends and to gain more insight into their treatment decisions or interventions.

The data used to validate our hypothesis was drawn from the PhysioNet 2012 dataset [Silva et al., 2012], an open competition that aimed to create new methods for patient-specific prediction of in-hospital mortality. The dataset includes the records of 4000 patients who have stayed at least 48 hours in one of the following four ICUs: Coronary Care Unit, Cardiac Surgery Recovery Unit, Medical ICU and Surgical ICU.

In this work, we elucidate the extent to which ICU mortality prediction may benefit from domain adaptation. In summary, our main contributions are:

- We propose deep models trained and applied for dynamic ICU mortality prediction. Our models are composed of convolutional and recurrent layers, thus offering a complementary spatial-temporal perspective of the patient condition. As a result, our models perform predictions that are based on information continuously collected over time and that can be dynamically updated as soon as new information becomes available.

- We propose a novel type of neural layer that not only improved the results of mortality prediction on ICU, but can also be used in many other domains, since it fits on any current neural network architecture.

- We show that patients within different ICU domains form sub-populations with different marginal distributions over their feature spaces. Therefore, we propose to learn specific models for different ICU domains that are trained using different feature transference approaches, instead of learning a single model for different

ICU domains. We show that the effectiveness of different feature transference approaches varies greatly depending on the factors that define the target domain.

- We conducted rigorous experiments using the PhysioNet 2012 dataset, which comprises data from four different ICU domains, that shows that multi-domain ICU data used for adaptation can significantly improve the effectiveness of the final model. Gains in terms of the Area Under the ROC Curve (AUC) range from 4% to 8% for early predictions, i.e., predictions based on data acquired during the first $5 - 20$ hours after admission, and from 2% to 4% for predictions within the first 48 hours after admission.

- We show that the patient representations along with the predictions provided by our models are meaningful in the sense that they form trajectories in a mortality risk space. Dynamics within this space can be very discriminative, enabling clinicians to track risky trends and to gain more insight into their treatment decisions or interventions.

1.3 Organization

The rest of this dissertation is structured as follows. First, Chapter 2 discusses related work and Chapter 3 describes our methodology, including the proposed mortality prediction model, the application addressed and the experiments designed to evaluate our multi-domain model for mortality preview. Then, Chapter 4 describes our experimental results and Chapter 5 concludes the dissertation.

Chapter 2

Related Work

In this chapter, we bring some of the most relevant research results that guided our work, exposing the methodologies used by the authors and how they correlate to ours. Research on predicting ICU mortality is of great academic interest in medicine [Cai et al., 2016; Tabak et al., 2014; Wu et al., 2017] and in clinical machine learning [Ghassemi et al., 2014; Johnson et al., 2016a; Luo et al., 2016; Nori et al., 2017], since a good model can help doctors to save lives. A number of researchers have investigated how to correlate ICU data with patient outcomes. In one of the first studies [Patel et al., 2009], a group of computer scientists, chemists, geneticists and philosophers of science was brought together to develop a model that could identify parameters in patient data that correlate with its outcome.

Next, we present an overview of works that address some of the most critical problems we faced in this dissertation, such as different domains and sub-populations and imbalanced data. We also explain how our work is different from the previous ones, since we explore local and temporal dependencies, which are able to create dynamic predictions.

2.1 Mortality Prediction

The PhysioNet ICU Mortality Challenge 2012 [Silva et al., 2012] provided benchmark data that incorporate evolving clinical data for ICU mortality prediction. As Johnson et al. [2014] reported, this benchmark data fostered the development of new approaches, leading to up to 170% improvement over traditional risk scoring systems that do not incorporate such clinical data currently used in ICUs [Gall et al., 1993]. In what follows, we discuss previous work in contrast with ours.

Most current work uses the PhysioNet ICU Mortality Challenge 2012 data. The most effective approaches are based on learning discriminative classifiers for specific sub-populations. One of the first works to use this data and also a top scorer on the challenge was that of Citi and Barbieri [2012], which proposes a robust Support Vector Machine (SVM) classifier. In their work, the authors train six different SVMs, each with a sixth of the negative examples and all positive examples, thus making each model capture specific patterns that lead to the patient outcome. Then, they used a linear model to combine the output of all SVMs into a single binary output, in order to predict the patient's survival. This work is focused on feature extraction and the authors used both general descriptors (features that do not change over time) and time-series features, which they represented by statistic descriptors (e.g., mean, minimum and maximum values). Other top scorers on the challenge, Bera and Nayak [2012] and Hamilton and Hamilton [2012], proposed similar approaches, but using a logistic regression classifier instead.

Vairavan et al. [2012] also employed logistic regression classifiers, but coupled them with Hidden Markov Models in order to model time-series data. Their Markov Chain was modeled to output the transaction probability between a patient being alive to being dead. They used this model to predict at each time step the patient's survival probability, and then used this score as an input to the logistic classifier, along with the patient's general descriptors and some selected features. One of their contributions is that their model does not need to receive the data from the whole 48 hours to output a prediction, allowing it to be used as a real time predictor.

Unlike the aforementioned works, Xia et al. [2012] used a shallow neural network approach. They trained one hundred small networks, composed of only two layers and fifteen neurons, and each of those models voted for the patient's output. The final decision was made by averaging all votes. Also, they did not used all available features, having selected the 26 most relevant features, and modeled the time-series with simple statistics, such as maximum, minimum and mean values. The work of Johnson et al. [2012] employed a tree-based Bayesian ensemble classifier. They also performed data pre-processing, polishing the input data based upon a domain knowledge, and feature extraction on the time-series signals. Their ensemble was composed of 500 weak learners, being each a decision tree with depth of two. Finally, they used a Markov chain Monte Carlo sampler to fit the ensemble parameters.

Krajnak et al. [2012] employed fuzzy rule-based systems for mortality prediction, aiming to combine clinicians expertise and machine learning techniques and used a genetic algorithm to generate the final solution. Like other works, they also represented the time series with statistic features. The usage of expert opinions in their work

was also a novel approach and they showed that it indeed improved the algorithms' performance, achieving competitive results.

McMillan et al. [2012] also proposed an unconventional approach. In their work, the authors created a model that identifies and integrates information in motifs that are statistically over- or under-represented in ICU time series of patients. They first discretized the time-series signals into sequences of symbols, which were then searched for short subsequences associated with the patient's true outcome, and finally used this information to train an SVM model that outputs the prediction.

More recently, Lee and Horvitz [2017] proposed a Markov model that accumulates mortality probabilities. They applied an exponential statistical model with parameter lambda to calculate at each time step the probability of a patient death, accumulating each result into the Markov model that outputs the final probability after the 48 hours of observation. Finally, the parameter lambda is estimated using statistical inference. Likewise, Barajas and Akella [2015] proposed an approach that models the mortality probability as a latent state that evolves over time. The latent state is created with the patient features, both general descriptors and time-series, and updated at each time step with the new patient observations. Unlike the previous works, they also used text features that provided context about the patient state.

Gong et al. [2015] proposed an approach to address the problem of small data using transfer learning in the context of developing risk models for cardiac surgeries. They explored ways to build surgery-specific and hospital-specific models using information from other kinds of surgeries and hospitals. Their approach is based on weighting examples according to their similarity to the target task training examples. The three aforementioned works are considered as baselines and compared with our approach.

Following Gong et al. [2015], in this work we use feature transference, but in a quite different way, as follows: (i) instead of applying instance weighting, we employed a deep model that transfers domain-shared features; (ii) we studied a broader scenario that includes diverse ICU domains; and (iii) our models employ temporal feature extraction, being able to dynamically predict the patient's outcomes.

2.2 ICU Domains and Sub-Populations

Imbalanced data [Bhattacharya et al., 2017], sub-populations of patients with different marginal distributions over their feature spaces [Nori et al., 2017] and sparse data acquired from heterogeneous sources [Ghassemi et al., 2015; Huddar et al., 2016] are

issues that pose significant challenges for ICU mortality prediction.

Gong et al. [2017] discussed problems due to the lack of consistency in how semantically equivalent information is encoded in different ICU databases. They argue that information is recorded differently across institutions and even over time, which can render potentially useful data obsolescent. The authors then propose a mapping that allows models to be built across different databases, thus making it possible to use more data for training.

Bhattacharya et al. [2017] discussed the problem of imbalanced ICU data, which occurs when one of the possible patient outcomes is significantly under-represented in the data. Further, since features are often imbalanced, some ICU domains have a significantly larger number of observations than others (e.g., respiratory failure in adults vs. children). The authors' approach is to transform the feature space, making the new features easier to classify. They approximate the probability distribution function for the set of samples of each class and then skew those probabilities, minimizing the intersection.

In a recent work, Bonomi and Jiang [2017] carried out a mortality study based on the notion of burstiness. They study the patient as a time-series where high values of burstiness indicate presence of rapidly occurring events in short time periods. This, in ICU data, may relate to possible complications in the patient's medical condition and hence provide indications on the mortality. Through this method they ended up encapsulating the dynamics of the patient's condition, basing their predictions on the behavior of the time-series, instead of on the values themselves.

While most studies on mortality prediction for ICU patients have assumed that one common risk model could be developed and applied to all the patients, Nori et al. [2017] advocated that this might fail to capture the diversity of ICU patients. Their method consist in constructing a few latent basis tasks, each having its own parameter vector, and then creating a parameter vector for each patient as a linear combination of those. The latent representation of a patient is then learned based on the collection of diseases associated with her. This way, the authors were able to model the patient, in place of creating a classifier, and could use the patient's representation not only for predictions but also for uncovering patient-specificity from different viewpoints.

2.3 Our Work

The works mentioned in this chapter show that ICU mortality prediction is a very well studied problem, with a very broad range of solutions, from simple classifiers

to complex statistical modeling. Although, none of the aforementioned approaches attempted to perform ICU domain adaptation, which is the core focus of our work. As shown by Alemayehu and Warner [2004], as well as by Seshamani and Gray [2004], models built using patient data from particular age groups perform poorly on other age groups because the features used to train the models have different distributions across the groups. There is often a mismatch between different ICU domains or patient sub-populations, and domain adaptation seems to be a natural solution for learning more robust models, as different ICU domains share features that exhibit different distributions. While data in different ICU domains may vary, there are potentially shared or local invariant features that shape patients in different ICU domains.

Another focus of our work is to capture local and temporal features from time-series ICU data. Features are captured in a way that the state of the patient in a certain time depends on the previous state. This forms a mortality risk space, and trajectories in this space allow to easily describe the state of the patient at a particular time, helping intensivists to estimate the patient progress from the current patient state. Most of the work cited before did not consider the time dependency of the problem, usually describing the patient's time-series as a set of statistics, while others were able to model the series features, but did not considered the general descriptors or even the feature dependency on a single time step.

Chapter 3

Methodology

In this chapter we will discuss the methods and guidelines used to create our predictive models and their applications. One can define the task of predicting patient outcomes from ICU data over time as follows. Each ICU patient can be represented by their physiological observations at a given time, such as heart rate, temperature, blood pressure, and others. Since a patient is continuously observed, his representation is an ordered set of multiple discrete time observations.

We then have as input the *training set*, which consists of a sequence of observations of the form $< A_t, o >$, where A_t is a vector of values corresponding to physiological parameters associated with a patient at time t and o is the outcome for the patient (i.e., whether or not the patient survived the hospitalization). The training set is used to construct a model that relates features within the sequence of observations to the patient outcome. The *test set* consists of a sequence of observations $< A_t, ? >$ for which only the physiological parameters for the patient until time t are available, while the corresponding patient outcome is unknown. The model learned from the training set is used to produce predictions of the outcome for patients in the test set.

The full set of data is split into five equally large stratified folds, used to perform a 5-fold cross validation. Each fold is divided in training and test set. Early stopping [Prechelt, 1998] was also applied, so the training set is divided itself in the actual training set, which is used to build the model, and a validation set, used to prevent the neural network from overfitting.

The task of predicting patients outcomes in the ICU has two important requirements:

- It is a domain-specific problem, i.e., a prediction model learned from a subpopulation (or ICU domain) is likely to fail when tested against data from other

population [Seshamani and Gray, 2004]. Feature transferability is thus an appealing way to provide robustness to prediction models.

- It is a time-sensitive problem, i.e., accurately predicting patient outcomes as early as possible may lead to earlier diagnosis and more effective therapy.

Our goal is to analyze patient data at each moment and evaluate the probability of a patient not surviving the treatment, simulating a real time medical expert with full attention to each patient. In order to do so, we need a well defined data structure that consists of fixed time steps and a invariable set of patient's signs at each time step.

3.1 Data and Domains

We use the publicly available dataset of multivariate clinical time-series of 4,000 patients from the PhysioNet 2012 challenge [Silva et al., 2012]. The data for each patient includes age, gender, height, weight and 37 time-stamped physiological parameters measured during the first 48 hours of ICU stay. All those parameters are listed in Table 3.1. Patient outcomes, including mortality, are available. Note that some of those features are measured a lot more frequently than others, as the difficulty to measure each feature differs. For instance, it is quite simple to measure someone's heart rate or temperature, but it is a lot harder and more costly to measure his Cholesterol.

In order to better understand the patients, we analyze some their outcomes that are not in-hospital death. Figure 3.1 shows a boxplot with the patient's length of stay grouped by the ICU. This type of plot allows us to understand the ICU population through the quantiles, indicating where is concentrated and how spread is the data. In this Figure, we can observe that patients's stay usually last around 10 to 15 days, going from a minimum of 2 days to almost 40. Also, the Cardiac ICU has the least variation, with most patients staying from 6 to 14 days, while the Surgical ICU variates the most, with patients' stay concentrated between 8 and 20 days.

We also show in Figure 3.2 how many days the patients survive after being hospitalized. We only included in this figure patients that have died some time after hospitalization. Patients without information of death were not included. Through this figure it is possible to observe that patients that go through the Cardiac ICU have the most survival time, reaching over 6 years, while patients from the Medical ICU are limited to less than 3 years. It is important to note from Table 3.1 that the average age from patients on the Cardiac ICU is 67.91 years old while from the patients in the Medical ICU is 62.83 years old.

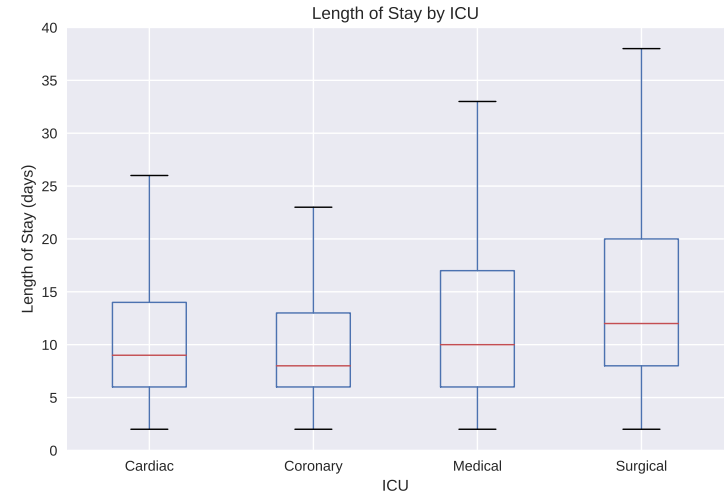

Figure 3.1. Relative frequency in which physiological parameters are measured in different ICU domains.

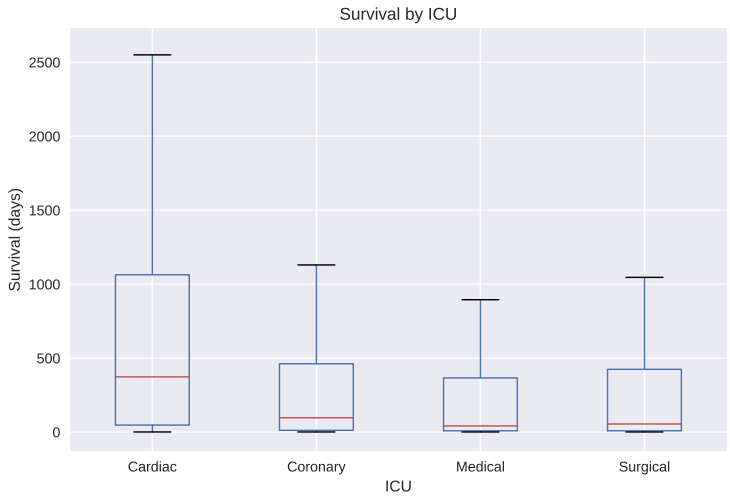

Figure 3.2. Relative frequency in which physiological parameters are measured in different ICU domains.

Figures 3.3 and 3.4 show calculated risk scores for each ICU. The SAPS (Simplified Acute Physiology Score) [Le et al., 1984] is a simple scoring system based on 14 easily measured biologic and clinical variables that aims to reflect the risk of death in ICU patients. This risk score ranges from 0 to 100. Figure 3.3 shows us that the SAPS score for all ICU has a similar distribution, varying from 1 to 30, with exception of

the Cardiac ICU that has a higher low boundary, median, first and third quartiles. In summary, the SAPS score for this ICU is usually higher.

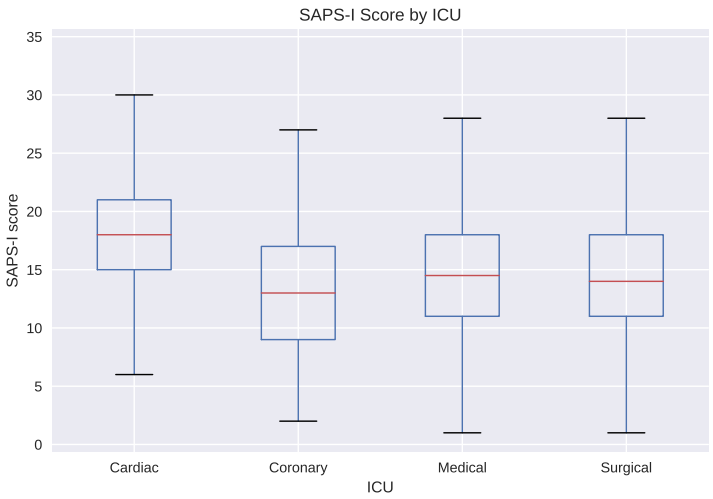

Figure 3.3. Relative frequency in which physiological parameters are measured in different ICU domains.

On the other hand, the SOFA (Sepsis-related Organ Failure Assessment) [Vincent et al., 1996] score describes quantitatively the degree of organ dysfunction or failure over time in a patient. It is composed of six scores, each designed to measure the risk of a specific body part, being those respiration, coagulation, live, cardiovascular, central nervous system and renal. In Figure 3.4 we see that the SOFA score ranges from 1 to 17, again not show much difference for all ICU but the Cardiac one, which as a similar range from the other but higher quartiles.

In order to make the data equally formated for each patient, we first propagate measurements forward (or backward) in time to fill gaps, so observations that are less frequent are considered constant until new measurement. We then resample the time series on an hourly basis, averaging the values observed on each hour for each patient feature, so that our patient can be represented by the mean value for each physiological observation on each hour during its ICU stay. Finally, we scale each variable to fall into the $[0, 1]$ interval. All patients are 16 years or older and had ICU stays of at least 48 hours. In contrast to Bhattacharya et al. [2017], we did not perform feature selection and thus used the entire feature-set in all experiments.

Table 3.1 shows the average physiological data for patients in each ICU domain. The dataset also specifies the ICU domain to which the patient has been admitted: Cardiac Surgery, Coronary Care Unit, Medical and Surgical. It is possible to conclude

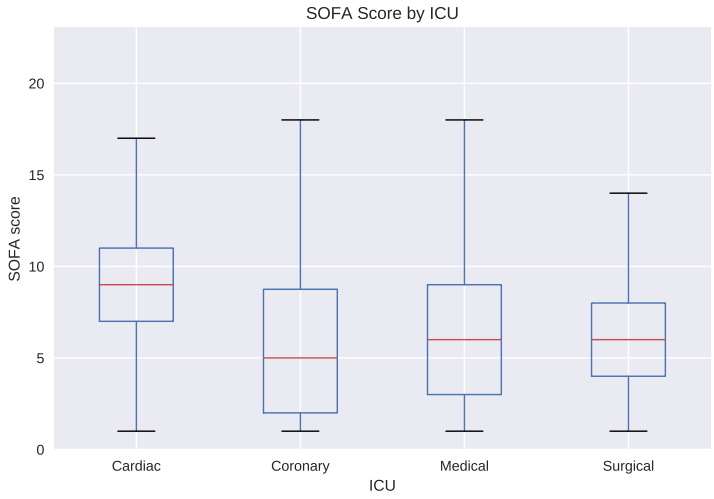

Figure 3.4. Relative frequency in which physiological parameters are measured in different ICU domains.

that physiological data differ greatly between patients admitted to different ICUs, but some features also have a common range across one or more ICUs, thus reinforcing our main hypothesis that transfer learning can indeed be applied to improve mortality prediction.

Figure 3.5 shows the relative frequency in which physiological parameters are measured within each ICU domain. As can be seen, some ICU domains have a significantly larger number of observations than others (e.g., $PaCO_2$ and PaO_2 are much more frequently measured in the Cardiac ICU, while TroponinT is much more frequently measured in the Coronary ICU).

3.2 Network Architecture

In this section we introduce the deep model architectures we evaluated to perform mortality prediction, eventually selecting those with the best results. We compared several architectures, from using only a Convolutional Neural Network [Krizhevsky et al., 2012] or recurrent layer, to combining both, and adding intermediate layers, such as Dropout layers. Convolutional and recurrent components offer a complementary perspective of the patient condition, as follows: the convolutional layer emphasizes the local interaction between physiological parameters, while the recurrent layer is designed to capture long range information and forget unimportant local information.

Table 3.1. Average patient physiological data. Mean, first and third quartiles within each physiological parameter. Mortality rate is concentrated in the Medical ICU (49.6% of all the deaths).

	Cardiac	Coronary	Medical	Surgical
N	874	577	1,481	1,067
Age	67.91 (56−79)	69.22 (59−81)	62.83 (51−78)	60.50 (48−76)
Male	530 (60.6%)	333 (57.7%)	753 (50.8%)	630 (59.0%)
Mortality Rate	4.9% (7.8%)	14.0% (14.6%)	18.6% (49.6%)	14.5% (28.0%)
Albumin (g/dL)	2.92 (2.4−3.5)	3.31 (2.9−3.6)	2.92 (2.5−3.3)	2.99 (2.5−3.5)
Alkaline phosphatase (IU/L)	74.93 (46−83)	92.44 (59−102)	126.15 (64−138)	91.43 (52−96)
Alanine transaminase (IU/L)	89.16 (18−45)	128.28 (19−78)	164.87 (16−61)	191.52 (17−84)
Bilirubin (mg/dL)	1.01 (0.4−1.1)	0.87 (0.4−0.9)	2.44 (0.4−1.6)	1.85 (0.5−1.5)
Blood urea nitrogen (mg/dL)	18.76 (12−21)	29.92 (16−36)	32.59 (14−42)	20.36 (11−24)
Cholesterol (mg/dL)	150.14 (114−174)	163.59 (134−189)	141.04 (111−169)	157.87 (122−184)
Creatinine (mg/dL)	1.04 (0.7−1.1)	1.58 (0.8−1.6)	1.64 (0.7−1.7)	1.12 (0.7−1.1)
Invasive diast. press. (mmHg)	58.85 (51−66)	62.65 (53−74)	54.97 (48−70)	59.65 (52−72)
Fractional inspired O2	0.91 (1.0−1.0)	0.82 (0.5−1.0)	0.72 (0.5−1.0)	0.72 (0.5−1.0)
Serum glucose (mg/dL)	129.28 (103−145)	165.74 (114−191)	155.02 (104−175)	148.85 (114−167)
Serum bicarbonate (mmol/L)	23.41 (22−25)	23.31 (21−26)	22.74 (19−26)	23.44 (21−26)
Hematocrit (%)	29.32 (25.3−32.8)	34.48 (30.7−37.8)	31.82 (27.9−36)	33.01 (29.1−36.8)
Heart rate (bpm)	85.43 (79−91)	84.32 (69−97)	95.61 (80−110)	87.83 (74−100)
Serum potassium (mEq/L)	4.49 (4−4.7)	4.28 (3.8−4.5)	4.19 (3.6−4.5)	4.07 (3.6−4.3)
Lactate (mmol/L)	2.76 (1.5−3.3)	2.76 (1.4−3)	2.58 (1.3−2.8)	2.65 (1.3−3.1)
Serum magnesium (mmol/L)	2.22 (1.8−2.4)	1.90 (1.7−2.1)	1.95 (1.6−2.1)	1.80 (1.5−2)
Invasive mean press. (mmHg)	78.86 (69−86)	86.14 (73−99)	86.58 (68−96)	87.13 (73−98)
Serum sodium (mEq/L)	138.42 (136−140)	137.82 (135−140)	138.96 (136−142)	139.33 (137−142)
Non-invasive diast. press. (mmHg)	52.21 (44−59)	61.15 (49−72)	62.03 (50−72)	62.42 (52−73)
Non-invasive mean press. (mmHg)	71.53 (62−79)	78.93 (67−89)	80.55 (68−91)	82.78 (71−94)
Non-invasive syst. press. (mmHg)	110.88 (96−125)	117.46 (101−134)	121.78 (104−138)	126.72 (108−145)
Partial press. of art. CO2 (mmHg)	41.20 (36−45)	40.61 (35−45)	42.50 (34−48)	41.01 (35−45)
Partial press. of art. O2 (mmHg)	295.46 (218−387)	181.58 (89−248)	147.68 (78−185)	188.24 (101−250)
Arterial pH (0-14)	7.39 (7.35−7.44)	7.84 (7.31−7.43)	7.44 (7.3−7.42)	7.46 (7.32−7.43)
Platelets (cells/nL)	170.36 (117−208)	241.44 (181−283)	230.89 (143−287)	219.19 (150−268)
Respiration rate (bpm)	17.55 (14−20)	19.74 (16−23)	21.10 (17−24)	18.95 (16−21)
O2 saturation in hemoglobin (%)	97.48 (97−98)	96.25 (96−98)	94.84 (94−98)	96.99 (97−98)
Invasive systolic press. (mmHg)	117.16 (105−127)	117.65 (100−139)	107.45 (95−137)	123.33 (108−148)
Temperature (°C)	35.57 (35.5−36.6)	36.38 (36−37.1)	36.77 (36.2−37.4)	36.51 (36.1−37.4)
Troponin-I (μg/L)	6.77 (0.8−10.1)	10.05 (0.8−12.4)	5.59 (0.8−7)	7.02 (0.4−6.7)
Troponin-T (μg/L)	1.51 (0.04−0.59)	2.78 (0.17−2.8)	0.33 (0.04−0.25)	0.22 (0.03−0.14)
Urine output (mL)	497.92 (120−615)	365.62 (100−500)	255.39 (70−325)	389.29 (100−500)
White blood cell (cells/nL)	12.98 (9.2−15.5)	12.31 (8.5−14.3)	13.33 (7.8−17)	12.37 (8.4−15.1)

Our first model was a single recurrent layer, more specifically a LSTM layer that sought to capture the tendencies between the patient states each time. Long-Short Term Memories are largely used in time-dependent problems, because of its great ability to deal with series data, so its a natural choice in this case. As our patient can be understood as a series of points moving in a high dimensional space, the LSTM will be able to create a representation based on this movement, which is then used to perform a prediction, although it may overlook the feature codependency in a single time step.

We also tried a Convolution-only model, that captured the relationship between features on a single time period, and then treated all time periods as one. This approach

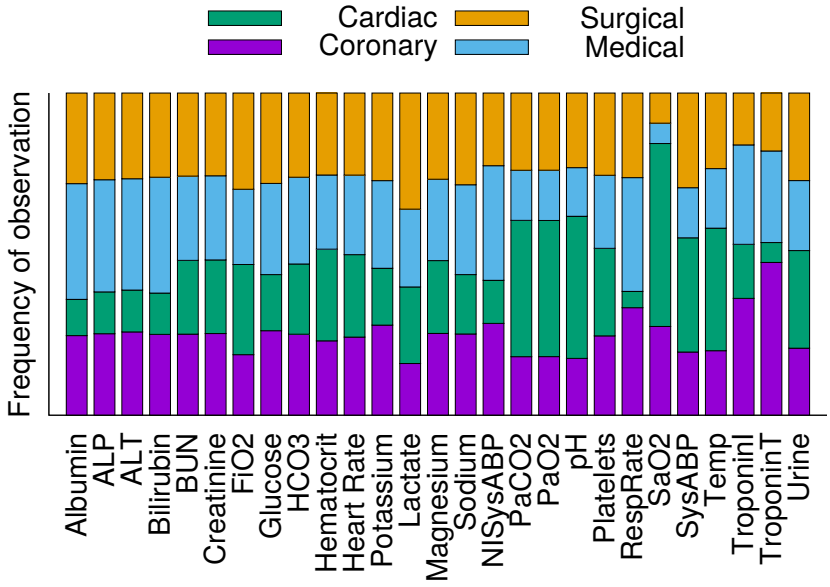

Figure 3.5. Relative frequency in which physiological parameters are measured in different ICU domains.

is not as intuitive as the later, but shows a surprisingly better performance. Here we create several filters that combine the patient observations only locally, i.e., it does not combine features across time. Alongside a Max Pooling layer, the model extracts some information about risk regions in the patient's feature space over this local combination and, finally, all this information is flattened into a single vector that is the patient representation, then used to predict his outcome. Although this method does not explicitly create a representation based on the patient's time series, by creating a flatting representation with all the time steps we are also encapsulating time information.

Finally, we have the model that employs a CNN layer followed by a max-pooling layer, thus extracting correlations between physiological parameters measured in the same time period and exploring their simultaneous effects. For instance, it may find that if both temperature and heart rate are high on the same time period, the odds of survival decrease. In a complementary way, the recurrent layer (LSTM) is devoted to learn how changes in observations for a patient affect the corresponding outcome. Intuitively, the recurrent layer captures temporal dependencies, enabling the estimation of a patient's progress from the current patient state. For instance, if the heart rate was low at the beginning of the stay and then becomes very high, then the odds of survival decrease. Finally, a dense layer takes the output of the recurrent layer and predicts the patient outcome. This model is shown in Figure 3.6.

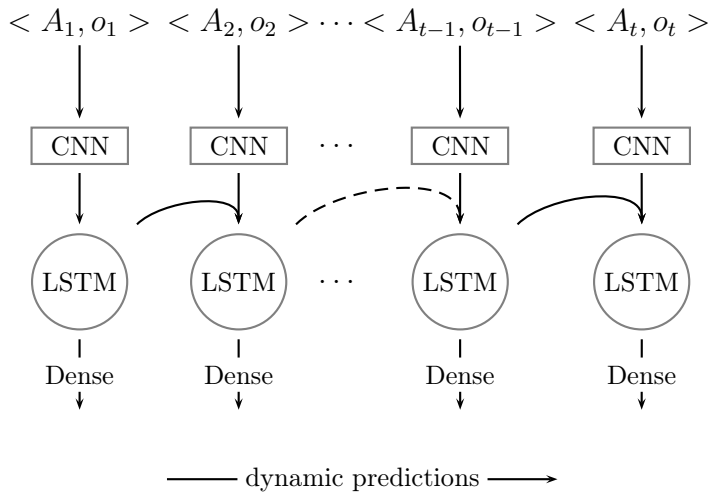

Figure 3.6. Network architecture for predicting patient outcomes over time. Each convolutional (CNN) layer is followed by a LSTM layer and different feature transference approaches are designed using this architecture.

Naturally, this is a high variance data, since all features come from measuring something as complex as a human being, which leads to the model quickly overfitting the training set. In order to prevent this, we applied several dropout layers [Srivastava et al., 2014], specifically after the input, max pooling and LSTM layers. A dropout layer will choose a random set of neurons at each batch and disable them during training. This will make the other neurons (that were not disabled) generalize more, simulating the effect of training multiple smaller networks and averaging them during test. We drop from 20% to 30% of all neurons on each layer. We also apply L2 regularization [Wager et al., 2013] to the LSTM inner cell neurons and the fully connected layer at the end of the model. This regularization will force each neuron to keep their activation weights low, thus generalizing more. Our loss function was binary cross-entropy, because of its good performance for classification problems with two classes. This loss function is given by the following formula:

$$\ell(\lambda) = -\frac{1}{n}\sum_{i=1}^{n}[y_i log(p_i) + (1 - y_i) log(1 - p_i)] + \lambda \sum_{j=1}^{k} w_j^2$$

where λ is the set of weights, n is the number of samples in the batch, y_i is the true output of the ith patient, p_i is the predicted output for the ith patient, and k is the number of neurons to regularize. We optimize the network weights using Adam [Kingma and Ba, 2014], a stochastic optimization method with adaptive momentum, that is able to quickly achieve low error on the training set.

The final component tested in the neural architecture was the activation function of each layer. A few activation functions have been tried, being those the following.

Linear

$$f(x) = x$$

Sigmoid

$$f(x) = \frac{1}{1+e^{-x}}$$

Tanh

$$f(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$$

Rectifier Linear Unit (ReLU)

$$f(x) = max(0, x)$$

Scaled Exponential Linear Unit (SELU)

$$f(x) = \lambda x, \text{ if x} > 0, \alpha e^{x} - \alpha, \text{ otherwise,}$$

being x the neuron output. Since this is a binary classification problem, ranged from 0 to 1, we chose to sustain a sigmoid activation on the output layer. This will scale any output to the (0, 1) interval.

In order to choose a set of model parameters that perform well for this task, a hand tuning method was applied. This means that we manually executed tests with different parameter sets and chose the set that performed best, making adjustments based on the output of previous executions. Those parameters include the number of neurons on each layer, activation functions (as mentioned above), regularization type and amount, dropout percentage, along with some layer-specific parameters, such as kernel size, for convolutional layers and pool size for max pooling layers.

In summary, our models work by passing each observation through a spatial feature extractor and then the sequence model captures how the extracted spatial features are associated with patient outcomes over time. Also, a dropout operation is performed after each layer of the network.

As not all the descriptors and time-series were available for all records, we had to deal with the problem of missing values. If one feature (either a descriptor or a time-series) was never recorded for a given record, we used the approach called "imputation" and replaced its values with zero. Because of the normalization step, this approximately corresponds to replacing the missing raw variable with a measure of central tendency, which corresponds to the arithmetic mean for Gaussian-distributed variables and to the geometric mean for log-normal ones. In some cases, the time-series measurements were taken only in the first 24 hours or only during the next 24 hours. In this case,

replacing with zero all the features related to the period with missing measurements could possibly create a non-existing improvement or deterioration trend. Instead, we duplicate the values from the available period, assuming stationarity conditions as default in absence of further measurements.

3.3 Feature Transferability

Our goal is to train multi-domain models to predict patient outcomes over time, which is based on patient observations associated with multiple ICU domains. Although patients associated with a given ICU domain may be better represented by specific features, there still exist some common features that permeate all other ICU domains.

The main intuition that we exploit for feature transferability is that the features must eventually transition from general to specific along our model. Besides, feature transferability drops significantly in higher layers with increasing domain discrepancy [Yosinski et al., 2014]. In other words, the features computed in higher layers must depend strongly on a specific domain, and prediction effectiveness suffers if this domain is discrepant from the target one. Since we are dealing with many domains simultaneously, we tested multiple transference approaches, which are detailed as follows:

A1: No layer is kept frozen during fine-tuning, i.e., errors are back-propagated through the entire network during fine-tuning.

A2: Only the convolutional layer is kept frozen during fine-tuning.

A3: Convolutional and LSTM layers are kept frozen during fine-tuning, i.e., errors are back-propagated only thought the fully-connected layers during fine-tuning.

A4: Only the convolutional layer is kept frozen during fine-tuning and other layers have their weights randomly initialized for fine-tuning.

A5: Convolutional and LSTM layers are kept frozen during fine-tuning and weights in fully-connected layers are randomly initialized for fine-tuning.

3.4 Switch Layer

As mentioned before, in this scenario we deal with high variance data. With the Switch Layer, our hypothesis is that there could be inner distributions on the data, i.e., there could be groups of patients that behave similar, being the optimum scenario those

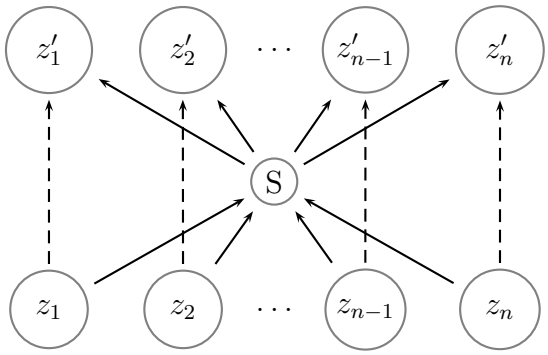

Figure 3.7. Switch Layer. Each circle is a neuron, each full arrow is a trainable weight and each dashed arrow is a constant weight.

groups matching exactly the ICU domains. Thus, this layer was designed to learn those groups during training, and then it uses this knowledge to modify outputs and improve prediction. It takes the output of the previous layer in the network and creates a dense representation for this output, using this representation to modify the output itself again. Figure 3.7 shows a diagram for the Switch Layer, where z_i is the ith neuron of the previous layer, s is the dense representation and z'_i is ith neuron modified by the switch. Also, the full arrows represent trainable weights and the dotted arrows indicate a copy (the same as setting the weight to 1).

Mathematically, the switch layer can be expressed as the following:

$$S = \phi_z(Z \cdot W_z + b_z)$$
$$Z' = \phi_s(S \cdot W_s + b_s) * Z$$

where Z is the previous layer output , W is the set of trainable weights, b is the bias, ϕ is the activation function, S is the switch output (also the layer inner representation), Z' is the modified output.

We called this layer a switch because the inner neurons (S), when constrained to assume either 0 or 1, will turn each input neuron on or off, performing a feature selection and, therefore, acting as a switch. It was initially inspired by the dropout layer, as a way to learn what neurons to drop according to the incoming data.

What this layer does is learning how to represent the input neurons in the dense space and apply this representation to scale each neuron according to its learned importance. Thus, similar patients will have similar representations and will have their neurons scaled similarly, while different patients will be scaled differently. This scaling helps the prediction by making the feature space more separable. As this layer is trained, it should be able to differ more patients that are not similar and then create a

more well-defined representation space. This inner activation is very similar to a fully connected layer, so one can interpret it as a classifier. It indeed can be seen this way and the learned classification is therefore used to scale the output.

The layer does not modify the input dimension, and can be applied to any layer output, including the network input. It also can be stacked, creating several representations in a row. In theory, a multi-layer Switch can work, but it was not tested in this work, since we only propose this layer here.

Chapter 4

Experimental Results

In this chapter we present our main questions about the problem, along with the baselines we used to compare our models' performance. We also show the main results obtained in several scenarios and discuss their implications.

In particular, our experiments aim to answer the following research questions:

RQ1: Does domain adaptation improve mortality prediction? Do models that are specific to each ICU domain improve the state-of-the-art models for mortality prediction?

RQ2: Which feature transference approach is more appropriate to each ICU domain?

RQ3: How accurate are dynamic predictions?

RQ4: How meaningful are the mortality risk spaces created from patient trajectories?

4.1 Baselines

We considered the following methods in order to provide a baseline comparison:

- Traditional classifiers: Support Vector Machines (SVM), Random Forest (RF), Logistic Regression (LR), Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA) and AdaBoost. The main objective of using these baselines is to compare our model with shallow ones.

- Training on Target (TT): A CNN−LSTM model is trained using only the target domain data. No source domain data is used. The main objective of using this baseline is to assess the benefits of different feature transfer approaches.

- Deep architecture [Che et al., 2015]: A deep network that uses prior-based regularization. The main objective of using this baseline is to compare our model with state-of-the-art results on the PhysioNet data.

4.2 Setup

The metric used to evaluate the effectiveness of our models is the standard Area Under the ROC Curve (AUC), as adopted by Che et al. [2015]. Like Johnson et al. [2012], we also used five-fold cross validation and relevant hyper-parameters were found using a further internal cross-validation. The results reported are the average of the five runs, which are used to assess the overall performance of the models. To ensure the relevance of the results, we assess the statistical significance of our measurements by means of a pairwise t-test [Sakai, 2014] with $p-value \leq 0.05$. Hereinafter, we refer to our model as $CNN-LSTM$.

First, we trained several architectures to identify which one would work better for this particular problem. We trained each architecture using three methods:

DA Domain Adaptation training, where we apply one of the techniques described on Subsection 3.3

TT Specific training, where only the target ICU is trained

GT General training, where all ICU are trained together

All the models were trained using the same set of hyper-parameters. We used a hand-tuning approach, and the set of hyper-parameters that presented the best overall AUC results is the following:

- 64 filters on the CNN;
- CNN kernel size of 5;
- CNN stride of 1;
- Max pooling of size 4;
- 70 neurons on the inner LSTM cell;
- SELU activation for both CNN and LSTM;
- L2 Regularization for both LSTM and Dense layers.

4.3 Domain Adaptation

In this section, we compare the performance of each training approach and discuss their results, searching to explain why each approach works better for each ICU domain.

Table 4.1. AUC comparison between Convolutional, Recurrent and CNN-LSTM models

	CNN			LSTM			CNN-LSTM		
ICU Domain	DA	TT	GT	DA	TT	GT	DA	TT	GT
Cardiac	0.828	0.737	0.866	0.786	0.740	0.812	0.833	0.773	**0.876**
Coronary	0.771	0.742	0.802	0.785	0.731	0.807	0.809	0.744	**0.833**
Medical	0.754	0.739	0.747	0.732	0.714	0.742	**0.757**	0.732	0.737
Surgical	**0.813**	0.787	0.812	0.752	0.753	0.769	0.807	0.791	0.801
Average	0.791	0.751	0.807	0.764	0.735	0.782	0.802	0.760	**0.812**

Table 4.1 displays the best result for each ICU domain and each model (Convolution Neural Network (CNN), LSTM Network and CNN-LSTM Network), considering each training method. Those results give us some insights about the models' behavior for each ICU. For instance, the specific training (M2) never outperformed the other two methods (more discussion on this later). Moreover, although the best overall performance was obtained with the CNN-LSTM model, we can see that the CNN model reaches a very close result with general training (M3). This indicates that, even though the convolution layer does not explicitly capture time series dependencies, the network's ability to look at all time steps at once comes close to the LSTM sequence representation for this problem. We can also note that, for the Surgical ICU, the best result is the one without the LSTM layer, which means that the time dependency in this case is not as important as for the other ICUs.

If we look at the results for each model individually and ignore the specific training (that underperformed on every experiment), we can see that the range of the AUC does not change too much and each ICU follows a distinct range. The Cardiac ICU always show the highest AUCs, followed by the Coronary ICU, then by the Surgical ICU and finally by the Medical ICU. We can interpret this as each ICU having its own difficulty in mortality prediction, which can be also explained by the patient profile in each ICU. More specific ICUs, such as Cardiac and Coronary, that deal with a single kind of disease or procedure, have more similar patients, thus making the prediction easier, since the death causes are usually alike. On the other hand, general purpose ICUs, such as the Surgical and Medical ones, tend to have a more diverse type of patient and the causes of death are a lot broader, making it more difficult to accurately predict mortality.

4.4 Switch Layer

In this section, we apply the Switch layer to the models described in Section 4.3 and evaluate their performance in the same manner. Using the same architectures than before, we introduce switch layers in the input and after the LSTM layer. For the CNN model, that does not have a LSTM layer, we apply the switch layer after a Flatten layer. In doing so, the model should be able to modify the input, improving the features that impact the most on the output, or alter the high level representations created by the previous layers in order to improve the final classification.

Table 4.2. AUC Scores for Models With and Without Switch Layer

	Without Switch			With Switch		
ICU Domain	Model	Mode	Score	Model	Mode	Score
Cardiac	CNN-LSTM	M3	0.876	CNN-LSTM-SW	M1	**0.881**
Coronary	CNN-LSTM	M3	0.833	CNN-LSTM-SW	M3	**0.837**
Medical	CNN-LSTM	M1	0.757	CNN-SW	M1	**0.762**
Surgical	CNN	M1	0.812	CNN-SW	M1	**0.827**
Average			0.818			**0.827**

Table 4.2 shows the best results for each ICU with and without the Switch layer. We observe that while without Switch layers we have domain adaptation (M1 mode) performing better on two ICU domains, Medical ICU and Surgical ICU, when applying those layers, domain adaptation also performs better on Cardiac ICU domain. This indicates that the shared feature space created by this layer benefits the domain adaptation, allowing more similarities between the ICU domains.

It is also clear from those results that the Switch layer improves the mortality prediction, with gains ranging from 0.4% to 1.5%, according to the ICU domain. Each ICU has a different gain, indicating that the Switch behaves differently for each domain, creating feature spaces that are either more simple or more complex, depending on the complexity of the domain. On average, the layer improves the AUC score by 0.9%.

4.5 Answering Our Research Questions

In this section, we take the best results from Sections 4.3 and 4.4 and use them to answer the research questions proposed at the beginning of this chapter. Here we compare our results with the state-of-the-art ones, expand the experiments on domain adaptation, show how the models' predictions behave in the patient's early admission hours and explain how we can construct a semantic space to provide an intuitive way to visualize the model predictions for medical experts.

4.5.1 Domain Adaptation and the State of the Art

The first experiment in this section is devoted to answer RQ1, i.e., how well does domain adaptation help to predict mortality. We present a comparison between our best models and other shallow and deep models. Table 4.3 shows AUC numbers for predictions performed using information acquired within the first 48 hours after the patient admission. Predictions performed by the baseline models were simply separated according to the ICU domain in which the corresponding patient was admitted, so that we can report AUC numbers for each ICU domain. On the other hand, the CNN-LSTM-SW model employs domain adaptation and, thus, is composed of four sub-models that are specific to each of the four domain ICUs. Clearly, domain adaptation improves the accuracy of our models and consistently outperform all baselines considered in this work. Overall, our model shows an AUC number of 0.827, which is considered to provide excellent clinical utility in the field of mortality prediction [Johnson et al., 2014].

Table 4.3. AUC numbers for shallow and deep models. Numbers in bold indicate the best models for each ICU domain.

Model	Cardiac	Coronary	Medical	Surgical
AdaBoost	0.572	0.551	0.510	0.531
SVM	0.627	0.572	0.503	0.532
LR	0.629	0.601	0.510	0.517
LDA	0.632	0.602	0.516	0.513
RF	0.610	0.578	0.587	0.623
QDA	0.689	0.668	0.567	0.610
[Che et al., 2015]	0.853	0.802	0.760	0.785
CNN-LSTM-SW	**0.881**	**0.837**	**0.762**	**0.827**

4.5.2 Domain Adaptation Approaches

The next set of experiments is concerned with RQ2. This is an extension of the results shown in Table 4.1, with the approaches discussed in Section 3.3. We present a comparison between the TT model and models learned following our five feature transfer approaches, along with the results for not performing Domain Adaptation. In this last case, we train on all ICU together and evaluate only on the target ICU. Table 4.4 shows AUC numbers for predictions performed using information acquired within the first 48 hours after the patient admission. Feature transfer is never detrimental when compared

with the TT and no Domain Adaptation models, and they provide substantial gains that are up to 6.7% (Cardiac), 8.3% (Coronary), 8.3% (Medical), and 11.0% (Surgical) when compared to the first one. These gains seem to be related to the mortality rate associated with each target ICU domain − gains are higher for domains with higher mortality rates.

Table 4.4. AUC numbers for different feature transference approaches. Numbers in bold indicate the best transference approach for each target ICU domain.

	TT	No DA	A1	A2	A3	A4	A5
Cardiac	0.821	0.876	0.852	**0.881**	0.829	0.849	0.858
Coronary	0.769	**0.837**	0.800	0.823	0.798	0.817	0.786
Medical	0.722	0.757	0.754	**0.763**	0.744	0.759	0.736
Surgical	0.727	0.812	0.821	**0.827**	0.778	0.818	0.788
Average	0.746	0.802	0.792	**0.804**	0.774	0.798	0.770

Finally, we can see from Table 4.4 that the best transfer approach varies depending on the target ICU domain, but for most of them, freezing only the convolutional layer works better. Randomly initializing the weights for fine-tuning does not show to be the best approach, since A4 and A5 were not the best performers for any target ICU domain, and those were the approaches that randomly initialized the weights. It seems that the temporal patterns play an important role when comes to ICU specifics, since A2 was the best for approach for most of them. For the Coronary ICU, however, not performing the domain adaptation gives the best results, which indicates that this ICU does not share many common factors with the other ones, taking the most advantage when they are all trained as one.

4.5.3 Early Predictions

The set of experiments presented now is devoted to answer RQ3. Figure 4.1 shows AUC numbers obtained with predictions performed using information acquired within the first y hours after the patient admission. As expected, the AUC increases as more information is acquired. From the first 5 to 20 hours, the slopes associated with Cardiac and Coronary domains increase much faster than the slopes associated with Medical and Surgical domains. It is also important to note that although the AUC increases as more data is introduced in the model, its value for the first measure, i.e., after 5 hours of admission, it is still high, indicating that the model can still predict the patient's outcome reasonably well, gaining more confidence as the time passes.

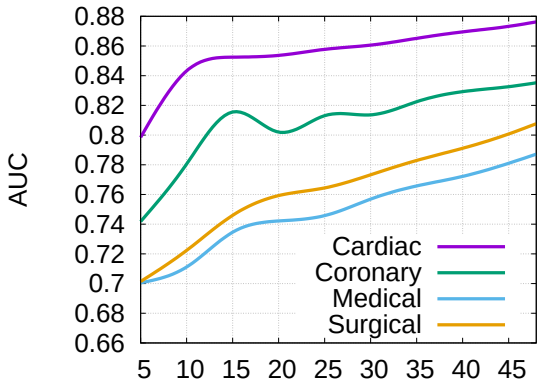

Figure 4.1. CNN−LSTM−SW AUC numbers for predictions performed using information within the first y hours after the patient admission ($5 \leq y \leq 48$).

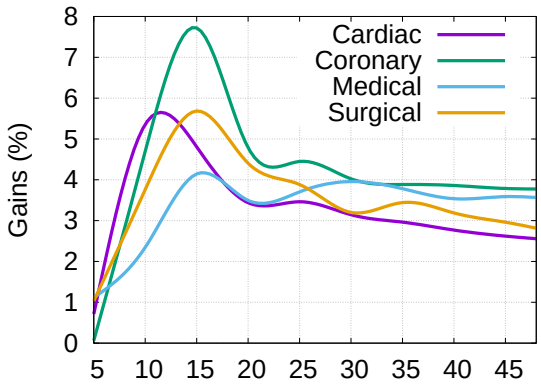

Figure 4.2. Gains over [Che et al., 2015] at different prediction times ($5 \leq y \leq 48$).

Figure 4.2 shows the gains obtained when compared with the work by Che et al. [2015] at different prediction times. The early predictions performed by the CNN-LSTM-SW architecture are much more accurate than the early predictions performed by Che et al. [2015], particularly in the first hours after the admission. The 10−20 hours period concentrates the more impressive gains, which vary from 4% (Medical) to almost 8% (Coronary).

4.5.4 Patient Dynamics

The last set of experiments is concerned with RQ4, i.e., they aim to assess how meaningful are the mortality risk spaces. Figure 4.3 shows risk spaces for each ICU domain, before and after applying the neural network. These spaces are obtained by gather-

ing patient trajectories, i.e., the coordinates (CNN-LSTM-SW representations) along with the predicted outcome at each time. Risk spaces can also be obtained from raw data and, in this case, the coordinates are simply the entire feature-vector. Risk spaces created from CNN-LSTM-SW representations are much more meaningful than the corresponding spaces obtained from raw data.

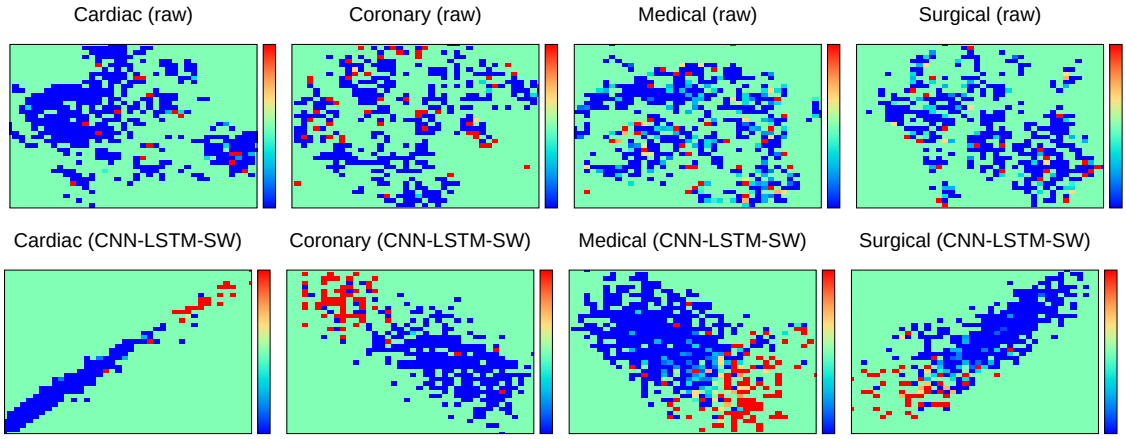

Figure 4.3. Mortality risk space for different ICU domains. Regions in red are risky.

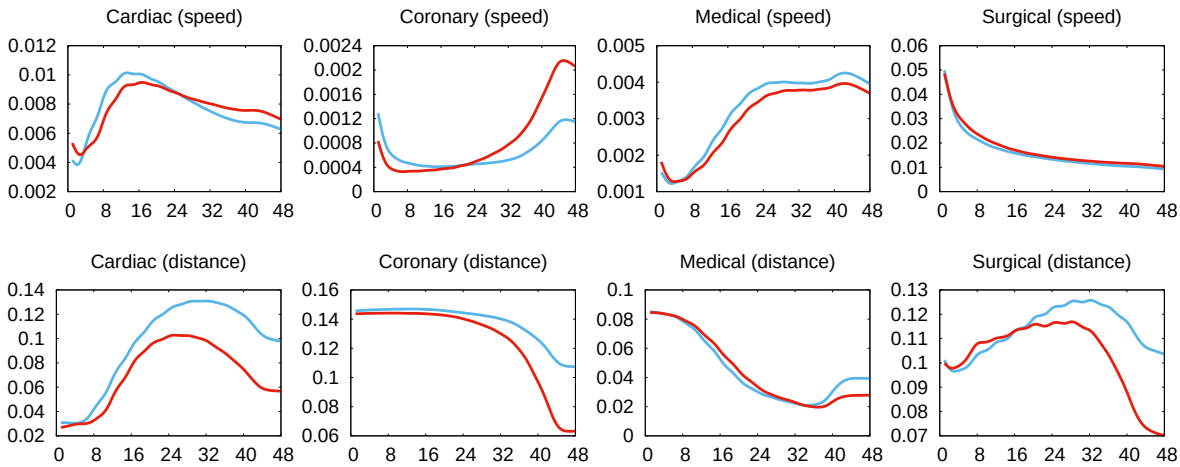

Figure 4.4. Dynamics of 48-hour trajectories in different ICU domains. Red curves are computed from trajectories associated with patients that have died. Blue curves are computed from trajectories associated with patients that survived.

Time is also encoded in the risk spaces and, thus, we can exploit dynamics, such as the proximity to mortality risky regions or the speed in which the patient condition changes. Figure 4.4 shows such dynamics in mortality risk spaces obtained from CNN−LSTM representations. Dynamics associated with the mortality risk space for

the Cardiac and Coronary ICU domains, for instance, are highly discriminative since red and blue curves are separated in the first hours after the patient admission. This may explain the high AUC numbers obtained in these domains. Patients show distinct dynamics, depending on the ICU domain. Patients admitted to the Cardiac and Surgical units, for instance, move much faster than patients admitted to the Coronary and Medical units. Also, the speed increases over time for patients admitted to the Coronary and Medical units.

Combining the results shown in Figure 4.3 and Figure 4.4, it is possible to provide an intuitive way of visualizing the model's output, by representing a patient as a particle moving through the mortality risk space. If the particle is moving towards a risky zone, the patient's survival probability is decreasing, otherwise it is increasing.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

ICU mortality prediction is a domain-specific problem. Thus, a prediction model learned from a sub-population of patients is likely to fail when tested against data from another population. In this dissertation, we investigated this problem by considering four sub-populations of patients that were admitted to different ICU domains. We were able to show that patients within a specific ICU domain are epidemiologically and physiologically different from patients within other domains. Nevertheless, patients across ICU domains still share basic characteristics. This motivated us to propose improved mortality prediction models based on domain adaptation. Specifically, we applied deep learning to create models that learned domain invariant representations from time series ICU data while transferring the complex temporal latent dependencies between ICU sub-populations. The proposed models employ local and temporal feature extractors, through a combination of convolutional and recurrent neural networks, being thus able to perform dynamic predictions during the ICU stay, potentially leading to earlier diagnosis and a more effective therapy. The proposed models are also able to be dynamically updated, improving their predictions with each new information about the patient's status. We showed that specific models built with domain adaptation outperforms the general models in three of the four ICU studied, making use of all the available data but still specializing on the target ICU domain.

We also propose a novel neural layer, capable of learning internal representations that are then used to modify the input data itself, creating thus a feature space that is easier to be classified by the model's final layers. The proposed layer is also agnostic to the problem, i.e., it could be used in other problems too and should be able to improve deep learning models in general. Finally, our models produce a mortality risk space,

and the dynamics associated with patient trajectories are meaningful and can be very discriminative, enabling clinicians to track risky trends and to gain more insight into their treatment decisions or interventions by observing how the patient's representation changes over time in the mortality risk space. Our models provide impressive gains (4% to 8%) for early predictions, i.e., predictions within the first 5-20 hour period after admission. Significant gains (2% to 4%) are also observed for predictions performed based on information acquired during the first 48 hours after admission. Although it may take several ours to train (each experiment took from 2 to 48 hours of training on a NVIDIA GeForce GTX TITAN X), the models can score patients almost instantly (under a second), which makes them viable to production and real time environments.

5.2 Future Work

In this dissertation, we analyzed the data of 4000 patients admitted to four different ICU, in the form of time series that reflects the status of each patient in the first 48 hours of stay. This data is available in the Physionet's MIMIC Database. However, we lack some important information about the patient's treatment, such as to which procedures was the patient submitted and what kind of interventions the doctors suggested, not to mention the patient's actual cause of death. Those informations are available in a latter databased, the Physionet's MIMIC-III Clinical Database [Johnson et al., 2016b], which comprises a more extensive set of information about 53,423 patients and which we intend to use in our future works.

Our studies have shown that domain adaptation indeed helps to create more robust models in order to predict mortality, but we were not able to explain why the model make such predictions or what is the main factors that eventually leads a patient to surviving or not. We identify as future work the following improvements:

- Provide means to explain the models' output. Given a single patient's status, we should be able to describe what are the most relevant features that the model analyzes to output its prediction. In doing so, we could provide a much clearer assistance to intensivists, being able to not only sort their patients by the attention needed, but also showing where among the patient's status to pay attention, leading to a better understating of the patient's condition.

- Suggest which interventions should be made. If the model identifies that a patient's survival probability is decreasing, it should be able to provide to intensivists the most likely effective interventions that should restore the patient's health.

- Show probable death causes. Once the model starts to predict that the patient may not survive, it should rank the most probable causes of death, helping the intensivists to prevent them as soon as possible.
- Define which exams are really necessary. By understanding which variables are important and predictive to the model, we can define what needs to be measured and what does not, since some variables are hard to obtain.
- Predict targets beyond in-hospital death, such as ICU staying time, out-hospital death and readmission. It is possible to study how well the model is able to predict those targets, as well as how each target affects the others.

So far we were able to build a model that can accurately predicts that a patient will or will not survive the ICU stay in the first few hours. Our main interests as future works is to help medical experts to better understand the model's output and be more effective on the patients' treatment.

Bibliography

Alemayehu, B. and Warner, K. (2004). The lifetime distribution of health care costs. *Health Services Research*, 3(39):627--642.

Barajas, K. C. and Akella, R. (2015). Dynamically modeling patient's health state from electronic medical records: A time series approach. In *Proc. of the 21st ACM SIGKDD*, pages 69--78.

Bera, D. and Nayak, M. (2012). Mortality risk assessment for ICU patients using logistic regression. *Computing in Cardiology*, pages 493--496.

Bhattacharya, S., Rajan, V., and Shrivastava, H. (2017). ICU mortality prediction: A classification algorithm for imbalanced datasets. In *Proc. of the 31st AAAI*, pages 1288--1294.

Bonomi, L. and Jiang, X. (2017). A mortality study for ICU patients using bursty medical events. In *Proc. of the 33rd IEEE ICDE*, pages 1533--1540.

Cai, X., Concha, O., Coiera, E., Martín-Sánchez, F., Day, R., Roffe, D., and Gallego, B. (2016). Real-time prediction of mortality, readmission, and length of stay using electronic health record data. *JAMIA*, 23(3):553--561.

Che, Z., Kale, D., Li, W., Bahadori, M., and Liu, Y. (2015). Deep computational phenotyping. In *Proc. of the 21st ACM SIGKDD*, pages 507--516.

Citi, L. and Barbieri, R. (2012). Physionet 2012 challenge: Predicting mortality of ICU patients using a cascaded SVM-GLM paradigm. *Computing in Cardiology*, pages 257--260.

Gall, J.-R. L., Lemeshow, S., and Saulnier, F. (1993). A new simplified acute physiology score (SAPS II) based on a european/north american multicenter study. *JAMA, Journal of the American Medical Association*, pages 2957--2963.

Ghassemi, M., Naumann, T., Doshi-Velez, F., Brimmer, N., Joshi, R., Rumshisky, A., and Szolovits, P. (2014). Unfolding physiological state: mortality modelling in intensive care units. In *Proc. of the 20th ACM SIGKDD*, pages 75--84.

Ghassemi, M., Pimentel, M., Naumann, T., Brennan, T., Clifton, D., Szolovits, P., and Feng, M. (2015). A multivariate time-series modeling approach to severity of illness assessment and forecasting in ICU with sparse, heterogeneous clinical data. In *Proc. of the 29th AAAI*, pages 446--453.

Gong, J., Naumann, T., Szolovits, P., and Guttag, J. (2017). Predicting clinical outcomes across changing electronic health record systems. In *Proc. of the 23rd ACM SIGKDD*, pages 1497--1505.

Gong, J., Sundt, T., Rawn, J., and Guttag, J. (2015). Instance weighting for patient-specific risk stratification models. In *Proc. of the 21st ACM SIGKDD*, pages 369--378.

Gruenberg, D. A., Shelton, W., Rose, S. L., Rutter, A. E., Socaris, S., and McGee, G. (2006). Factors influencing length of stay in the intensive care unit. *American Journal of critical care*, 15(5):502--509.

Hamilton, S. and Hamilton, J. (2012). Predicting in-hospital death and mortality percentage using logistic regression. *Computing in Cardiology*, pages 489--492.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735--1780.

Huddar, V., Desiraju, B. K., Rajan, V., Bhattacharya, S., Roy, S., and Reddy, C. (2016). Predicting complications in critical care using heterogeneous clinical data. *IEEE Access*, 4:7988--8001.

Johnson, A., Dunkley, N., Mayaud, L., Tsanas, A., Kramer, A., and Clifford, G. (2012). Patient specific predictions in the intensive care unit using bayesian ensemble. *Computing in Cardiology*, pages 249--252.

Johnson, A., Ghassemi, M., Nemati, S., Niehaus, K., Clifton, D., and Clifford, G. (2016a). Machine learning and decision support in critical care. *Proc. of the IEEE*, 104(2):444--466.

Johnson, A., Kramer, A., and Clifford, G. (2014). Data preprocessing and mortality prediction: the physionet/cinc 2012 challenge revisited. *Computing in Cardiology*, pages 157--160.

Johnson, A. E., Pollard, T. J., Shen, L., Li-wei, H. L., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L. A., and Mark, R. G. (2016b). Mimic-iii, a freely accessible critical care database. *Scientific data*, 3:160035.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Krajnak, M., Xue, J., Kaiser, W., and Balloni, W. (2012). Combining machine learning and clinical rules to build an algorithm for predicting ICU mortality risk. *Computing in Cardiology*, pages 401--404.

Krizhevsky, A., Sutskever, I., and Hinton, G. (2012). Imagenet classification with deep convolutional neural networks. In *Proc. of the 26th NIPS*, pages 1106--1114.

Le, J. G., Loirat, P., Alperovitch, A., Glaser, P., Granthil, C., Mathieu, D., Mercier, P., Thomas, R., and Villers, D. (1984). A simplified acute physiology score for icu patients. *Critical care medicine*, 12(11):975--977.

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

Lee, D. H. and Horvitz, E. (2017). Predicting mortality of intensive care patients via learning about hazard. In *Proc. of the 31st AAAI*, pages 4953--4954.

Luo, Y., Xin, Y., Joshi, R., Celi, L., and Szolovits, P. (2016). Predicting ICU mortality risk by grouping temporal trends from a multivariate panel of physiologic measurements. In *Proc. of the 30th AAAI*, pages 42--50.

McMillan, S., Chia, C., Esbroeck, A., Rubinfeld, I., and Syed, Z. (2012). ICU mortality prediction using time series motifs. *Computing in Cardiology*, pages 265--268.

McNeill, G. and Bryden, D. (2013). Do either early warning systems or emmergency response teams improve hospital patient survival? *Resuscitation*, 84(12):1652--1667.

Nori, N., Kashima, H., Yamashita, K., Kunisawa, S., and Imanaka, Y. (2017). Learning implicit tasks for patient-specific risk modeling in ICU. In *Proc. of the 31st AAAI*, pages 1481--1487.

Patel, V., Shortliffe, E., Stefanelli, M., Szolovits, P., Berthold, M., Bellazzi, R., and Abu-Hanna, A. (2009). The coming of age of artificial intelligence in medicine. *Artificial Intelligence in Medicine*, 46(1):5--17.

Prechelt, L. (1998). Automatic early stopping using cross validation: quantifying the criteria. *Neural Networks*, 11(4):761--767.

Sakai, T. (2014). Statistical reform in information retrieval? *SIGIR Forum*, 48(1):3--12.

Seshamani, M. and Gray, A. (2004). A longitudinal study of the effects of age and time to death. *Journal of Health Economics*, 2(23):217--235.

Silva, I., Scott, G. M. D., Celi, L., and Mark1, R. (2012). Predicting in-hospital mortality of ICU patients: The physionet/computing in cardiology challenge. *Computing in Cardiology*, pages 245--248.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929--1958.

Tabak, Y., Sun, X., Nunez, C., and Johannes, R. (2014). Using electronic health record data to develop inpatient mortality predictive model: Acute laboratory risk of mortality score (alarms). *JAMIA*, 21(3):455--463.

Vairavan, A., Eshelman, L., Haider, S., Flower, A., and Seiver, A. (2012). Prediction of mortality in an intensive care unit using logistic regression and hidden markov model. *Computing in Cardiology*, pages 393--396.

Vincent, J.-L., Moreno, R., Takala, J., Willatts, S., De Mendonça, A., Bruining, H., Reinhart, C., Suter, P., and Thijs, L. (1996). The sofa (sepsis-related organ failure assessment) score to describe organ dysfunction/failure.

Wager, S., Wang, S., and Liang, P. S. (2013). Dropout training as adaptive regularization. In *Advances in neural information processing systems*, pages 351--359.

Wang, D. and Nyberg, E. (2015). A long short-term memory model for answer sentence selection in question answering. In *Proc. of the 53rd ACL Conference*, pages 707--712.

Wu, M., Ghassemi, M., Feng, M., Celi, L., Szolovits, P., and Doshi-Velez, F. (2017). Understanding vasopressor intervention and weaning: risk prediction in a public heterogeneous clinical time series database. *JAMIA*, 24(3):488--495.

Wunsch, H., Guerra, C., Barnato, A. E., Angus, D. C., Li, G., and Linde-Zwirble, W. T. (2010). Three-year outcomes for medicare beneficiaries who survive intensive care. *Jama*, 303(9):849--856.

Xia, H., Daley, B., Petrie, A., and Zhao, X. (2012). A neural network model for mortality prediction in ICU. *Computing in Cardiology*, pages 261--264.

Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. (2014). How transferable are features in deep neural networks? In *Proc. of the 28th NIPS*, pages 3320--3328.