Predicting Hospital Readmission via Cost-sensitive Deep Learning

Haishuai Wang, Zhicheng Cui, Yixin Chen, Michael Avidan,
Arbi Ben Abdallah, Alexander Kronzer

Abstract—With increased use of electronic medical records (EMRs), data mining on medical data has great potential to improve the quality of hospital treatment and increase the survival rate of patients. Early readmission prediction enables early intervention, which is essential to preventing serious or life-threatening events, and act as a substantial contributor to reduce healthcare costs. Existing works on predicting readmission often focus on certain vital signs and diseases by extracting statistical features. They also fail to consider skewness of class labels in medical data and different costs of misclassification errors. In this paper, we recur to the merits of convolutional neural networks (CNN) to automatically learn features from time series of vital sign, and categorical feature embedding to effectively encode feature vectors with heterogeneous clinical features, such as demographics, hospitalization history, vital signs and laboratory tests. Then, both learnt features via CNN and statistical features via feature embedding are fed into a multilayer perceptron (MLP) for prediction. We use a cost-sensitive formulation to train MLP during prediction to tackle the imbalance and skewness challenge. We validate the proposed approach on two real medical datasets from Barnes-Jewish Hospital, and all data is taken from historical EMR databases and reflects the kinds of data that would realistically be available at the clinical prediction system in hospitals. We find that early prediction of readmission is possible and when compared with state-of-the-art existing methods used by hospitals, our methods perform significantly better. For example, using the general hospital wards data for 30-day readmission prediction, the area under the curve (AUC) for the proposed model was 0.70, significantly higher than several baselines. Based on these results, a system is being deployed in hospital settings with the proposed forecasting algorithms to support treatment.

Index Terms—Readmission Prediction; Deep Learning; Electronic Medical Records; Cost-sensitive; Categorical Feature Embedding.

1 Introduction

BIG-DATA based predictive algorithms in medical community has been an active research topic since the Electronic Medical Records (EMRs) captured rich clinical and related temporal information [20]. The applications of machine learning to solve important problems in healthcare, such as predicting readmission [21], [25], have the potential to revolutionize clinical care and early prevention [30].

Background and Significance: A hospital readmission is defined as admission to a hospital within a specified time frame after an original admission. Different time frames such as 30-day, 90-day, and 1-year readmissions have been used for research purposes. Readmission may occur for planned or unplanned reasons, and at the same hospital as original or admission at a different one [8]. Readmission prediction is significant for two reasons: quality and cost of health care. High readmission rate reflects relatively low quality and also has negative social impacts on the patients and on the hospital [14]. Nearly 20 percent of hospital patients are readmitted within 30 days of discharge, a $35 billion problem for both patients and the healthcare system. Avoidable readmissions account for around $17 billion a year [12]. Consequently, readmission is becoming more important as an indicator for evaluating the overall healthcare effectiveness. Identifying patients at high risk for readmission early during hospitalization may aid efforts in reducing readmissions [23]. It is significant to predict readmission early in order to prevent it.

We propose to develop, validate and assess machine learning, forecasting algorithms that predict readmission for individual patients. The forecasting algorithms will be based on data consolidated from heterogeneous sources, including the patient's electronic medical record, the array of physiological monitors in the operating room and the general hospital wards, and evidence-based scientific literature.

There are some existing forecasting algorithms being used to predict readmission [2], [3], [8], [19], [21]. However, these algorithms have some shortcomings, making them inapplicable to our datasets and objectives:

1. They predict patients without considering the misprediction costs of different categories. In a readmission prediction problem where the occurred cases (minority class) are usually quite rare as compared with normal populations (majority class), the recognition goal is to detect patients with readmission. A favorable classification model is one that provides a higher identification rate on the minority class (Positive Prediction Value) under a reasonably good prediction rate on the majority class (Negative Prediction Value).

2. Time-series is commonly used in the medical domain since medical equipments record vital signs with certain time interval. They first extract discriminative features from the original time series and then use off-the-shelf classifiers to predict, which is ad-hoc and separates the feature extraction part with the classification part, resulting in limited

- *H. Wang is with the Department of Biomedical Informatics, Harvard University, Boston, MA, 02115. E-mail: haishuai_wang@hms.harvard.edu*
- *Z. Cui and Y. Chen are with the Department of Computer Science and Engineering, Washington University in St. Louis.*
- *M. Avidan, A. B. Abdallah and A. Kronzer are with the Department of Anesthesiology at Washington University School of Medicine.*

Manuscript received XXX; revised XXX.

accuracy performance.

3. They use inefficient feature encoding and limited patient characteristics are related to a certain disease. A straightforward way for EMR feature encoding is to extract or create feature vector manually. The hand-crafted features are usually a mix of numerical and categorical features, which poses a challenge for directly applying classifiers as they can only deal with numerical inputs by design. Furthermore, the values of categorical features are indicators of categories instead of true values, resulting in they can not be input to classifiers directly. Thus, an effective feature encoding method is required to improve prediction accuracy. Besides, with the increasing use of EMRs, more existing patient characteristics can result in more effective prediction [19].

4. Though our goal is to make predictions for various medical datasets, such as data from general wards or operating rooms, they have not provided an integrated clinical decision support system for hospitals to predict readmission from heterogeneous, multi-scale, and high-dimensional data.

Nowadays, deep learning has been one of the most prominent machine learning techniques [1], [11], [18], [22]. Deep neural networks have multiple hidden layers structure and each hidden layer has non-linear activation functions. Therefore, deep learning has capability to model data with non-linear structures and learn high-level representation of features. In other words, deep learning aims to model high-level abstractions in the data using nonlinear transformations. Such abstractions can then be used to interpret the data, or to build better predictive models. Through stacking multiple layers, the model is able to capture much richer structures and learn significantly more complicated functions. Convolutional Neural Networks (CNN) is reported as a successful technique for time series classification [7], [9], [24] because of its ability to automatically learn complex feature representation using its convolutional layers. Thus, CNN is able to handle time series data without requiring any handcrafted features.

We aim to apply deep learning techniques to develop better models for early readmission prediction. At the same time, we need to consider the imbalanced or skewed class distribution problem, which yields varying costs information for different types of misclassification errors. In this paper, we present cost-sensitive deep learning models for clinical readmission prediction using data collected by monitoring different vital signs, demographics and lab results. Specifically, we first automatically learn the feature representation from the vital signs time series using CNN, and simultaneously construct the feature vectors by categorical feature embedding. Without loss of generality, we also extract statistical features from time series (such as first order and second order features) and feed into the feature vector. Then, we combine the learned time series features from CNN and feature vectors from categorical feature embedding as input to a Multi Layer Perceptron (with multiple hidden layers). At the output layer, a cost-sensitive prediction formulation is used to address the imbalanced challenge. A cost-sensitive prediction can be obtained using Bayesian optimal decision based on a cost matrix. The cost matrix denotes the uneven identification importance between classes, so that the proposed approach put on weights on learning towards the rare class associated with higher misclassification cost. The method we develop in this paper is focused on a much broader class of patients (ward patients and surgery patients), and deployed in a real system for supporting treatment and decision making. Model performance metrics are compared to state-of-the-art approaches. Our method outperforms the existing methods on real clinical datasets, and is being deployed on a real system at a major hospital.

The remainder of this paper is organized as follows. Section 2 describes the data we used and the problem definition. We introduce the proposed feature extracting and learning approach in Section 3. Section 4 shows the proposed predictive model with the learned features. In Section 5, we conduct the experiments on the real-world EMR datasets and compare the proposed method with benchmark approaches. The results are discussed in Section 6 and the system deployment is demonstrated in Section 7. We review traditional readmission prediction methods and feature learning from EMRs in Section 8. Lastly, we draw conclusions in Section 9.

2 Data Description

The work described in this paper was done in partnership with Washington University School of Medicine and Barnes-Jewish Hospital, one of the largest hospitals in the United States. We used two real datasets from Barnes-Jewish Hospital. A large database is from the general hospital wards (GHWs) between July 2007 and July 2011. GHWs gathered data from various sources, including more than 30 vital signs (pulse, shock index, temperature, heart rate etc.) from routine clinical processes, demographics, real-time bedside monitoring and existing electronic data sources from patients at the general hospital wards (GHWs) at Barnes-Jewish Hospital. The readmission class distribution is imbalanced, which makes the prediction task very difficult.

Another dataset is operating room pilot data (ORP), which is derived from heterogeneous sources, including the patient's electronic medical record, the array of physiological monitors in the operating room, laboratory tests, and evidence-based scientific literature. The ORP includes more than 40 vital signs during surgery (such as heart rate which are recorded every minute) and patients' pre-operation information such as demographics, past hospitalization history, surgery information and tests. The demographic features in our data include patients' age, gender, height, weight, race and so on. The surgery information includes surgery type, anesthesia type, and etc.

The purpose is to develop forecasting algorithms that mine and analyze the data to predict the patients' outcomes (specifically, whether or not they would be re-admitted). The forecasting algorithms will be based on data collected from general wards or operating rooms. The algorithm will facilitate patient-specific clinical decision support (such as early readmission prediction) to enable early intervention. The system is being implemented and deployed in the Barnes-Jewish Hospital.

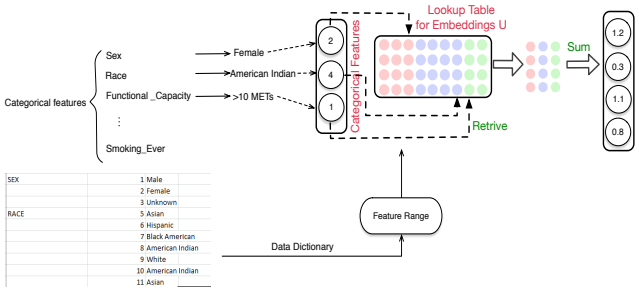

Fig. 1. The architecture of feature embedding for categorical EMR data. Each categorical feature is represented as a d-dimensional vector, where d is user defined. The number of columns in the lookup table is based on the features range calculated from the data dictionary. Different color represents different categorical feature. Each categorical feature would retrieve its corresponding embedding in the lookup table as its new feature representation by index.

3 Preprocessing and Features

Data exploration and preprocessing, and feature extraction are critical steps for the success of any application domain. They are especially important for our clinical domain since the data are noisy, complex, and heterogeneous. Thus, prior to feature encoding, several preprocessing steps are applied to eliminate outliers and find an appropriate feature representation of patient's states.

We first preprocess the dataset by removing the outliers. The raw data typically contain many reading and input errors because information are recorded by nurses and there are inevitably errors introduced by manual operations. We list the acceptable ranges of every feature based on the domain knowledge of the medical experts in our team. Then we perform a sanity check of the data and replace the abnormal values that are outliers by the mean value of the entire population.

Second, not all patients have values for all signs in a real clinical data, and many types of clinical features involved in lab tests are not routinely performed on all patients. We use the medium value of a sign over the entire historical dataset to fill the missing values.

Finally, we normalize the data to scale the values in each bucket of every vital sign so that the data values range in the interval [0,1]. Such normalization is helpful for prediction algorithms such as deep learning.

A key aspect in any application of data mining is building effective features for classification or prediction. Before building our model, we first worked with the physicians from Barnes-Jewish Hospital as well as studied prior work to determine useful features, since the input of our model is based on the feature embedding from raw medical data. Based on the characteristics of our datasets, we have discrete features, continuous features, and time series features which record the vital values at different time. The continuous features (such as patient's height) can be concatenated into the feature vector directly since we have the normalization process during preprocessing. However, values of categorical features are indicator of a category instead of true values, and time series of vital sign is usually high-dimensional especially for a long monitor period. Thus, these features are inapplicable to a classifier directly. Feature encoding is required for categorical features and effective features need to be extracted from the time series feature before being added to the vector. In this paper, we use the statistical features extracted from various of data types (i.e., numerical, categorical, and time series) in the datasets, at the same time, we adopt convolutional neural networks to automatically learn discriminative features from vital sign. In this way, the built features not only contain statistical information but also hold temporal and local information as well as the overall trend of time series.

3.1 Categorical Feature Embedding

Since neural networks can only deal with numerical inputs by design, the categorical features can not be input to neural networks directly. One-hot encoding is the most popular way to convert a categorical feature to a numerical one. By one-hot encoding, the new representation is a vector with one element being one and all others being zero. However, typically one-hot encoding results in high-dimensional sparse vectors if a categorical feature has plenty of categories. To encode categorical features efficiently, we propose a categorical feature embedding approach that first converts the categorical part to a numerical hidden representation which is then concatenated with the original numerical part. This new representation will be fed into the remaining neural network and generate the final output.

We use a lookup table $\mathbb{U}$ that contains a numerical embedding for each category in the categorical part. Figure 1 shows the lookup table component. There are independent zones (represented by different color in Figure 1) in the lookup table to represent different features. Categories from the same feature locates in the same zone, and the number of embedding in each zone equals to the number of categories of the corresponding categorical feature. Each column in the lookup table represents a category of a feature, and each row is a d-dimensional embedding for a category. The dimensional of d is user defined.

Assume there are P categorical features of $\mathbf{x}^C$, where $\mathbf{x}^C = [[\mathbf{x}^C]_1, \cdots, [\mathbf{x}^C]_P]$. And the p^{th} categorical feature has K_p categories, such that $[\mathbf{x}^C]_p \in \{1, 2, \cdots, K_p\}$. The total number of all categories is $K = \sum_{p=1}^{P} K_p$.

Consequently,the lookup table is a matrix $\mathbb{U} \in \mathcal{R}^{d \times K}$. For instance, in Figure 1, $d = 4$ and $K = 9$ as $P = 3, K_1 = 3, K_2 = 2, K_3 = 4$. The corresponding embedding in the lookup table for each categorical feature can be retrieved, and the lookup table can be updated and learned based on backpropagation. After the feature embedding, categorical features have new feature representation with numerical values. Mathematically, let $\mathbf{u}_i$ abe the i^{th} column vector in lookup table $\mathbb{U}$, and $q(j)$ be the index that the j^{th} categorical feature would use as index for retrieving. Let $A_j = \sum_{p=1}^{j} K_p$ be the total number of previous categories up to categorical feature j. The value of j^{th} categorical feature $[\mathbf{x}_j^C]$, would retrieve embedding by index

$$q(j) = A_{j-1} + [\mathbf{x}^C]_j \tag{1}$$

After embedding retrieval, we obtain P number of d dimensional embeddings. Then element-wise summation is applied to get the representation of all categorical features. Suppose the new representation for categorical features is $g(\mathbf{x}^C)$. Then we have

$$g(\mathbf{x}^C) = \sum_{j=1}^{P} \mathbf{u}_{q(j)} \quad (2)$$

This new representation would then be fed into the next layer in the neural networks, and all the embeddings are updated and learned through backpropagation.

3.2 Feature Extraction from Time Series of Vital Sign

To capture the temporal effects in time series, we use a bucketing technique. For the time series data of each patient, we divided it into 2 buckets based on the care time (for ORP) or room start time (for GHWs), and compute the features in each bucket. Then, we extract first order features and second order features from patients' vital sign time series in each bucket. The details of first order and second order feature from time series are as follows.

3.2.1 First Order Features

We use some traditional statistical features as the first order features. Specifically, the first order features include maximum, minimum, mean (μ), standard deviation(σ), skewness and kurtosis in each bucket. Skewness is a measure of symmetry of the probability distribution of a real-valued random variable. The larger absolute value of skewness means the greater deviation of its distribution. Kurtosis is a measure of whether the data are heavy-tailed or light-tailed relative to a normal distribution. A larger absolute value of the kurtosis represents greater difference between the steepness of its distribution and the normal distribution. The formula of mean, standard deviation, skewness and kurtosis are:

$$\mu = \frac{\sum_{i=1}^{N} x_i}{N}, \quad \sigma = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \mu)}{N}} \quad (3)$$

$$Skewness = \frac{\sum_{i=1}^{N}(\mathbf{x}_i - \mu)^3}{(N-1)\sigma^3} \quad (4)$$

$$Kurtosis = \frac{\sum_{i=1}^{N}(\mathbf{x}_i - \mu)^4}{(N-1)\sigma^4} - 3 \quad (5)$$

3.2.2 Second Order Features

The most commonly used second order features are co-occurrence features. The co-occurrence features in one-dimensional time series have been shown to perform better than other second-order features [16]. The data is firstly quantized into Q levels, and then a two dimensional matrix $\phi(i,j)(1 \le i,j \le Q)$ is constructed. Point (i,j) in the matrix represents the number of times that a point in the sequence with level i is followed, at a distance d_1, by a point with level j. The co-occurrence features we used are Energy (E_1), Entropy (E_2), Correlation ($\rho_{x,y}$), Inertia, and Local Homogeneity (LH). The features are calculated by the following equations:

$$E_1 = \sum_{i=1}^{Q}\sum_{j=1}^{Q} \phi(i,j)^2, \quad E_2 = \sum_{i=1}^{Q}\sum_{j=1}^{Q} \phi(i,j) * log(\phi(i,j))$$

$$\rho_{x,y} = \frac{\sum_{i=1}^{Q}\sum_{j=1}^{Q}(i-\mu_x)(j-\mu_y)\phi(i,j)}{\sigma_x \sigma_y}$$

where:

$$\mu_x = \frac{\sum_{i=1}^{Q} i \sum_{j=1}^{Q} \phi(i,j)}{Q}, \sigma_x^2 = \frac{\sum_{i=1}^{Q}(i-\mu_x)^2 \sum_{j=1}^{Q} \phi(i,j)}{Q}$$

$$\mu_y = \frac{\sum_{j=1}^{Q} j \sum_{i=1}^{Q} \phi(i,j)}{Q}, \sigma_y^2 = \frac{\sum_{j=1}^{Q}(j-\mu_y)^2 \sum_{i=1}^{Q} \phi(i,j)}{Q}$$

$$Inertia = \sum_{i=1}^{Q}\sum_{j=1}^{Q}(i-j)^2\phi(i,j), LH = \sum_{i=1}^{Q}\sum_{j=1}^{Q}\frac{\phi(i,j)}{1+(i-j)^2}$$

We set $Q = 5$ in our experiments. The extracted first order and second order features are concatenated into the one-hot vector as input to our model.

3.3 Convolutional Neural Network for Time Series Feature Learning

We use Convolutional Neural Network (CNN) to automatically learn features from time series (such as heart rate, temperature and blood pressure which are recorded every minute). In our setting, we regard CNN as feature extractor. The input time series is fed into CNN model, containing several convolutional layers, activation layers and max-pooling layers to learn features.

The convolutional layer contains a set of learnable filters which are updated using the backpropagation algorithm. Convolution operation can capture local temporal information from the time series. We use the same filter size through all convolutional layers.

The activation layer introduces the non-linearity into neural networks and allows it to learn more complex model. We adopt $tanh(\cdot)$ as our activation function in all activation layers.

The max-pooling layer aims to provide an abstracted form of the representation by down-sampling. At the same time, it reduces the computational cost by reducing the number of parameters to learn and provides basic translation invariance to the internal representation.

We used two convolutional layers for time series feature learning, and followed by two pooling layers. The max-pooling is used for down-sampling. Each max-pooling layer is inserted in-between successive convolutional layers in the CNN architecture. The filter size for two convolution layers is 64 and 32, respectively. The convolved signal is pooled with pool size of 2.

The statistical features can be combined with features learnt from CNN, and furthure feed them into a multilayer perceptronn for readmission prediction task. In principle, our extracted and learnt features can be used as input to any classification algorithms.

4 Prediction Methodology

A main challenge in our application is that we have severely skewed datasets as there are much more non-readmission patients than those with readmission. For example, among 2565 records in the GHWs data, only 406 have a 30-day readmission. This extremely imbalanced class distribution makes the prediction task very difficult.

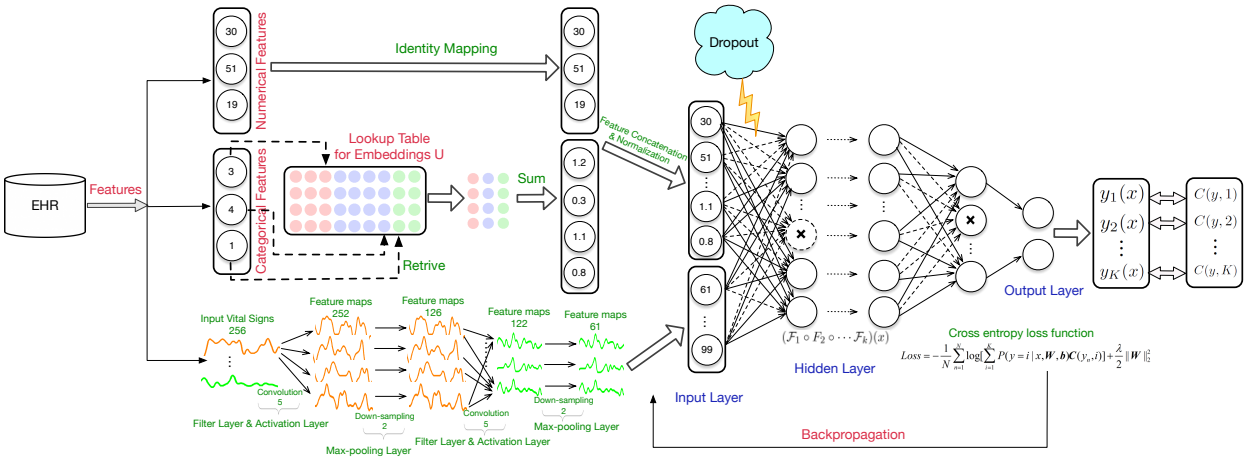

Fig. 2. CSDNN overall framework. There are three components in the architecture, including feature extraction/learning part, neural network part, and prediction part. All three parts are connected through backpropagation algorithm. Identity mapping is used for numerical features, and lookup table is for categorical feature embedding. In the meanwhile, we use convolutional neural networks to automatically learn features from time series of vital sign. During prediction, CSDNN considers different costs of misclassification errors with a cost matrix **C** in the output layer. Once acquiring predicted outcome **y**, the predicted errors can be calculated with cost matrix **C** according to the loss function in Eq. (9). All the parameters including the categorical embedding matrix $\mathbb{U}$, CNN parameters for time series feature learning as well as the weights and biases of each hidden layer are learned jointly by backpropagation. Regularization techniques including dropout and L2 regularization are also applied to improve the generalization ability of the network.

4.1 Classification Algorithms

In the medical domain, the cost of misdiagnosing abnormal patient as healthy is different with misdiagnosing healthy as abnormal patient. In most cases, the proportion of normal patients is larger than abnormal patients (e.g., readmission and ICU patients). Therefore, in our datasets, we have two crucial issues during classification. One is imbalanced outcomes and another one is low sensitivity of abnormal patients. Standard classifiers, however, pay less attention to rare cases in an imbalanced dataset. Consequently, test patients belonging to the small class are misclassified more often than those belonging to the prevalent class.

To over this problem, we formalize it as a cost-sensitive classification problem. Cost-sensitive classification considers the varying costs of different misclassification types. A cost matrix encodes the penalty of classifying samples from one class as another. Bayesian optimal decision can help obtain the cost-sensitive prediction . Eq. 6 shows the predicted class label that reaches the lowest expected cost:

$$y_{pred} = \arg\min_{1\le k\le K} \sum_{i=1}^{K} P(y=i|\mathbf{x},\mathbf{W},\mathbf{b})\mathbf{C}(k,i) \tag{6}$$

where $\mathbf{C}(k,i)$ denotes the cost of predicting a sample from class k as class i. K is the total number of classes. In our case, K equals 2 since this is a binary readmission classification. The diagonal elements in the cost matrix are the weights of corresponding categories, others are zero. Larger value in the cost matrix impose larger penalty. In the experiments, we set values in the cost matrix based on parameter study method. The $P(y=i|\mathbf{x},\mathbf{W},\mathbf{b})$ is to estimate the probability of class i given $\mathbf{x}$. The probability estimator can be any classifiers which the outputs are probability. In this work, we use a modified cross entropy loss function that embeds the cost information. We denote the deep neural network (DNN) with cost sensitive as CSDNN for short.

The CSDNN consists of one input layer, one output layer and multiple hidden layers. There are m neurons in the input layer, where m is the dimension of input feature vector. The hidden layers are fully-connected with the previous layer. Each hidden layer h uses $\mathbf{W}_h$ as a fully-connected weight matrix and $\mathbf{b}_h$ as a bias vector that enters the neurons. Then, for an input feature vector $\mathbf{x}$, the output of the hidden layer is $\mathcal{H}(\mathbf{W}_h\mathbf{x}+\mathbf{b}_h)$, where the activation function $\mathcal{H}$ can be sigmod or tanh. We used tanh in the experiments because it typically yields faster training (and sometimes better local minima), that is, $\mathcal{H}(\alpha) = (e^{\alpha}-e^{-\alpha})/(e^{\alpha}+e^{-\alpha})$. After H hidden layers, the DNN describes a complex feature transform function by computing:

$$\mathcal{F}(\mathbf{x}) = \mathcal{H}(\mathbf{W}_H \cdot \mathcal{H}(\cdots \mathcal{H}(\mathbf{W}_2 \cdot \mathcal{H}(\mathbf{W}_1\mathbf{x}+\mathbf{b}_1)+\mathbf{b}_2)\cdots)+\mathbf{b}_H) \tag{7}$$

Then, an output layer is placed after the H-th hidden layer. From hidden layer to output layer is a softmax function to output the probability of feature vector $\mathbf{x}$ belonging to each category. Hence, there are K neurons (outputs) in the output layer, where the i-th neuron with weights $\mathbf{W}_o^i$ and bias $\mathbf{b}_o^i$ (the subscript o represents the parameters in the output layer). The estimate of probability of class i given $\mathbf{x}$ can be formulated as follows:

$$\begin{aligned} P(y=i|\mathbf{x},\mathbf{W},\mathbf{b}) &= softmax_i(\mathbf{W}_o\mathcal{F}(\mathbf{x})+\mathbf{b}_o) \\ &= \frac{\exp(\mathbf{W}_o^i\mathcal{F}(\mathbf{x})+\mathbf{b}_o^i)}{\sum_{k=1}^{K}\exp(\mathbf{W}_o^k\mathcal{F}(\mathbf{x})+\mathbf{b}_o^k)} \end{aligned} \tag{8}$$

To learn and optimize the parameters of the model, we set the cross entropy as the loss function and minimize the loss function with respect to $\{\mathbf{W}_h\}_{h=1}^H$, $\{\mathbf{b}_h\}_{h=1}^H$, $\mathbf{W}_o$ and $\mathbf{b}_o$. The loss function over the training set is as follows:

$$Loss = -\frac{1}{N}\sum_{n=1}^{N} log\big[\sum_{i=1}^{K} P(y=i|\mathbf{x}_n,\mathbf{W},\mathbf{b})\mathbf{C}(y_n,i)\big] + \frac{\lambda}{2}\|\mathbf{W}\|_2^2 \tag{9}$$

where N is the total number of patients, y_n is the readmission indicator for the n-th patient where 1 indicates readmission and 0 control, and $P(y = y_n|\mathbf{x}_n, \mathbf{W}, \mathbf{b})$ is the n-th patient calculated by the model. The class number K equals 2 for readmission prediction. The loss minimization and parameter optimization can be performed through the backpropagation using mini-batch stochastic gradient descent.

The CSDNN framework is shown in Figure 2. For the numerical features, we use identity mapping to map numerical values to a feature vector. The lookup table is learned for categorical feature embedding. Then, all the features from identity mapping, categorical feature embedding, CNN as well as statistical feature are concatenated and normalized as input vector to a multilayer perceptron. The cost matrix is applied to the loss function in Eq. (9) during prediction phase. We use two hidden layers MLP. There are 128 hidden units for the first hidden layer and 64 units in the second hidden layer. We also use dropout to avoid over-fitting. To estimate parameters of models, we utilize gradient-based optimization method to minimize the loss function. Since backpropagation is an efficient and most widely used gradient-based method in neural networks [29], we use backpropagation algorithm to train our CSDNN model. As mini-batch gradient descent could converge faster than full-batch for large scale datasets, we adopt mini-batch gradient descent with momentum to efficient update the parameters.

5 EXPERIMENTS AND EVALUATION

5.1 Datasets and Setup

We evaluate performance of proposed CSDNN algorithm on two real datasets from Barnes-Jewish Hospital. One data is from general hospital wards (GHWs) while another one is pilot data from operating room (ORP). The two datasets are described in Section 2, and more details are as follows:

GHWs data: We aim to predict 30-day and 60-day readmission on the GHWs data. There are 41,503 patient visits in the GHWs data, and 2,565 have the outcomes of readmission or not. In this dataset, each patient is measured for 34 indicators, including demographics, vital signs (pulse, shock index, mean arterial blood pressure, temperature, and respiratory rate), and lab tests (albumin, bilirubin, BUN, creatinine, sodium, potassium, glucose, hemoglobin, white cell count, INR, and other routine chemistry and hematology results). A total of 406 patients are readmitted within 30 days and 538 instances are readmitted within 60 days.

ORP data: We aim to predict 30-days and 1-year readmission in the ORP data (there is no 60-day outcomes in this dataset). There are 700 patients in the pilot data with more than 50 pre-operation features and 26 intra-operation vital signs of each patient. Since there are plenty of null outcomes in the pilot dataset, we remove the patients with null outcomes. A total of 157 patients are readmitted within 1 year and 124 patients are readmitted within 30 days.

For both datasets, we randomly select 60%, 15%, and 25% from readmission and non-readmission patients as training data, validation data and test data, respectively. We choose the best parameters through validation data. Based on the data distribution and parameter study, we set the cost of misclassifying readmission patients to non-readmission patients is twice as many as misclassifying non-readmission patients to readmission patients on the GHWs data, and 1.5 times on the ORP data.

TABLE 1
Confusion Matrix

		True Condition	
		Positive	Negative
Predicted Condition	Positive	True Positive (TP)	False Positive (FP)
	Negtive	False Negative (FN)	True Negative (TN)

5.2 Evaluation Criteria

Following the most common procedure for evaluating models for early predicting readmission, we use: ROC Receives Operating Characteristic) Curve, AUC (Area Under ROC Curve), Accuracy, Sensitivity (Recall), Specificity (Precision), PPV (Positive Predictive Value), and NPV (Negative Predictive Value) to evaluate the proposed method. To compute the measurements, we can use a confusion matrix illustrated in Table 1 to summarize the results of testing the algorithm. From Table 1, the accuracy, specificity, sensitivity, F1-score, NPV and PPV can be calculated from Eq. (10).

$$
\begin{aligned}
&Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \\
&Sensitivity = \frac{TP}{TP + FN},\ Specificity = \frac{TN}{TN + FP} \\
&PPV = \frac{TP}{TP + FP},\ NPV = \frac{TN}{TN + FN} \\
&F1 = 2 \cdot \frac{PPV \cdot Sensitivity}{PPV + Sensitivity} = \frac{TP + TN}{TP + TN + FP + FN}
\end{aligned}
\tag{10}
$$

Baselines: We evaluate CSDNN for comparison with existing approaches used in hospitals. From the literature study, the existing predictive methods for readmission are mainly based on feature extraction for specific disease or dataset, and then input the extracted features to classifiers. The most widely used classifiers are Support Vector Machine (SVM), Logistic Regression(LR), Decision Tree (DT), Random Forest (RF) and Artificial Neural Networks (ANN). In spite of the settings of our problem are not exactly the same with all the baselines, we implement baselines based on their methodologies used in the state-of-the-art approaches for readmission prediction. Specifically, Mao et al. [16] proposed an integrated data mining approach with the statistical features (in Sections 3.2.1 and 3.2.2) but without CNN feature learning. They applied an exploratory undersampling [15] method to deal with the class-imbalance problem, and used RF as classifier and obtain good performance. Somanchi et al. [19] extracted features from heterogeneous data source (such as demographics and vitals), and employed SVM as classifier for cardiac arrest early prediction. Kim et al. [13] used extra physiological variables extracted from an APACHE critical care system, and shows DT classifier achieves the best performance. Almayyan [2] selected discriminative features using PSO and

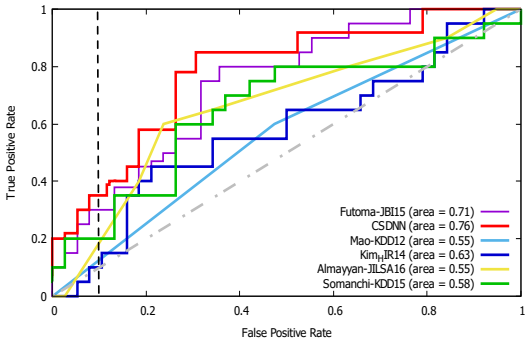

Fig. 3. ROC curves of 1-year readmission prediction on the ORP dataset.

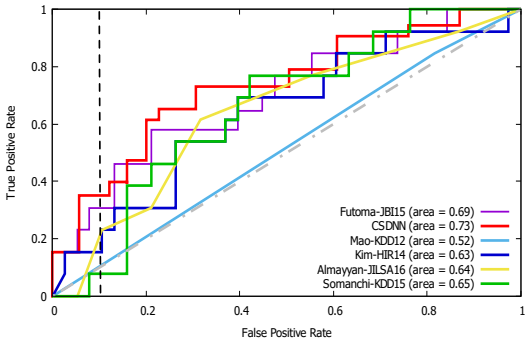

Fig. 4. ROC curves of 30-day readmission prediction on the ORP dataset.

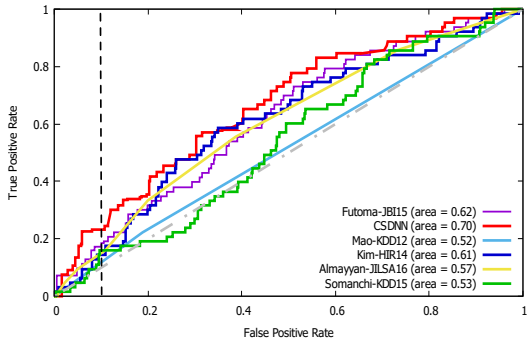

Fig. 5. ROC curves of 30-day readmission prediction on the GHWs dataset.

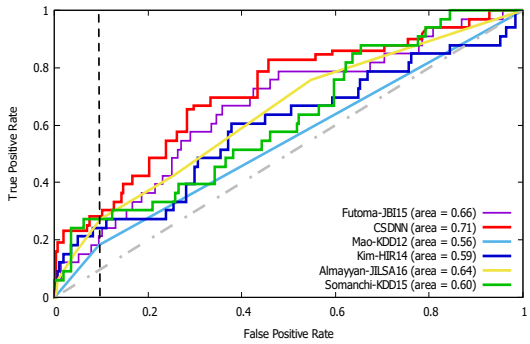

Fig. 6. ROC curves of 60-day readmission prediction on the GHWs dataset.

several feature selection techniques to reduce the features dimension, and applied random forest classifier to diagnose lymphatic diseases. Futoma [8] applied ANN for predicting early hospital readmission and achieved good predictive performance than regression methods.

For simplicity, we use Mao_KDD12, Somanchi_KDD15, Kim_HIR14, Almayyan_JILSA16, and Futoma_JBI15 for short to represent the benchmark approaches.

6 Results and Discussion

Results: Tables 3-5 and Figures 3-6 present the performance of the different predictive approaches on the GHWs and the ORP datasets. In comparison to the state-of-the-art baselines on the test set, we observed that our model (CSDNN) performs better than baselines in terms of AUC and PPV. The PPVs of our model are approximately twice the best value of that found in the baselines on the GHWs dataset. Obviously, the PPV is statistically significantly improved by using cost-sensitive deep learning. This is critical since the misclassification costs of readmission patients is more serious. Our goal is to make the predictions for readmission patients as precise as possible under high NPV, which enables the hospital to intervene early, as well as adjust the schedules for physicians and nurses to optimize overall quality of care for all patients. As we can observe from the ROC curves in Figures 3-6, under the same false positive rate, we are able to predict readmission with high true positive rate, which is better than that of in the baselines.

Discussion: We achieved high accuracy mainly because we used both sufficient statistical features and automatically learned time series features by convolutional neural networks (CNN), as well as we consider the misclassification costs to improve PPV. Compared with traditional statistical features, CNN can learn a hierarchical feature representation from raw data automatically, which make it possible to improve the accuracy of feature-based methods. Cost-sensitive deep learning approach ensures the prediction of rare but high misclassification cost class, which are developed by introducing cost items into the learning framework. However, this may affect the prediction for non-readmission patients (indicated by NPV). Thus, the PPV and NPV need to be tradeoff. As we believe the cost of a false positive is considerably higher than a false negative, relatively low NPV may be a tolerable tradeoff.

Sensitivity analysis: For any test, there is usually a tradeoff between the different measures. This tradeoff can be represented using a ROC curve, which is a plot of sensitivity or true positive rate, versus false positive rate (1-specificity). For practical deployment in hospitals, a high specificity (e.g. >90%) is needed. The ROC figures also show the results of all algorithms with specificity being fixed close to 0.90. Even at this relatively high specificity, the CSDNN approach can achieve a sensitivity of around 35% on the ORP data. The sensitivity of ORP data is relatively higher than GHWs data, because the ORP data is a small pilot data and not very imbalance compared with GHWs data.

7 System Deployment

The work described here was done in partnership with Barnes-Jewish Hospital, one of the largest hospitals in the United States. Based on our performance, the results is good enough to deploy a decision support system with the proposed predictive algorithms to support treatment. The purpose of the clinical decision support system is to identify prognostic factors and suggest interventions based

TABLE 2
30-day readmission prediction on the GHWs dataset.

Method	Accuracy	Specificity	Sensitivity	F1-Score	AUC	NPV	PPV
Somanchi_KDD15	0.83	0.85	0.08	0.15	0.53	0.88	0.19
Mao_KDD12	0.72	0.86	0.18	0.30	0.52	0.85	0.20
Kim_HIR14	0.85	0.85	0.00	0.00	0.61	0.89	0.08
Almayyan_JILSA16	0.84	0.85	0.11	0.19	0.57	0.86	0.15
Futoma_JBI15	0.84	0.86	0.23	0.36	0.62	0.87	0.16
CSDNN	0.89	0.89	0.26	0.44	0.70	0.85	0.37

TABLE 3
60-day readmission prediction on the GHWs dataset.

Method	Accuracy	Specificity	Sensitivity	F1-Score	AUC	NPV	PPV
Somanchi_KDD15	0.87	0.91	0.19	0.31	0.60	0.90	0.25
Mao_KDD12	0.84	0.91	0.19	0.31	0.56	0.90	0.18
Kim_HIR14	0.90	0.90	0.00	0.00	0.59	0.96	0.08
Almayyan_JILSA16	0.90	0.91	0.23	0.37	0.61	0.98	0.03
Futoma_JBI15	0.89	0.91	0.23	0.37	0.66	0.95	0.06
CSDNN	0.92	0.93	0.26	0.45	0.71	0.91	0.31

TABLE 4
1-year readmission prediction on the ORP dataset.

Method	Accuracy	Specificity	Sensitivity	F1-Score	AUC	NPV	PPV
Somanchi_KDD15	0.64	0.70	0.16	0.26	0.58	0.87	0.20
Mao_KDD12	0.67	0.74	0.21	0.32	0.55	0.81	0.15
Kim_HIR14	0.61	0.77	0.31	0.44	0.63	0.82	0.37
Almayyan_JILSA16	0.71	0.76	0.38	0.51	0.55	0.87	0.08
Futoma_JBI15	0.68	0.78	0.35	0.48	0.71	0.80	0.32
CSDNN	0.77	0.79	0.41	0.54	0.76	0.82	0.42

TABLE 5
30-day readmission prediction on the ORP dataset.

Method	Accuracy	Specificity	Sensitivity	F1-Score	AUC	NPV	PPV
Somanchi_KDD15	0.65	0.72	0.00	0.00	0.65	0.87	0.06
Mao_KDD12	0.65	0.74	0.22	0.34	0.52	0.82	0.15
Kim_HIR14	0.61	0.80	0.24	0.37	0.63	0.86	0.22
Almayyan_JILSA16	0.73	0.78	0.25	0.38	0.64	0.85	0.23
Futoma_JBI15	0.78	0.83	0.41	0.55	0.69	0.82	0.25
CSDNN	0.82	0.87	0.49	0.64	0.73	0.88	0.35

on novel feature extracting and learning algorithms using heterogeneous data.

We are building up a system to deploy our CSDNN algorithm for early readmission prediction. The system architecture is demonstrated in Figure 7. The system is an internet based tool for medical data analysis and outcome prediction, for example, readmission prediction via our CSDD algorithm. There are four key components in the system: 1) Data acquisition. There are user-friendly interfaces to guide user how to submit a job and how to train a model. Physicians can upload historical EMR data to the system following the sample data format. 2) Data preprocessing. After uploading the training data by physicians, the system preprocesses the raw data with several modules, including feature extracting, feature encoding and feature learning. 3) Model selection. Users can select which model will be trained. Since we integrated several models into the system for different tasks, users should select one model for specific purpose. In our case, CSDNN should be selected to predict readmission. Once a model is selected, the system will train the model using the uploaded data. 4) Prediction. Once the training phase is over, test data can be fed into the system. Test data will be analyzed using the trained model (CSDNN). Finally, the system shows the results to indicate whether the patient is readmission or not.

8 Related Work

A number of forecasting algorithms exist that use medical data for outcomes prediction [17], [20], [28]. To predict whether a patient is readmitted to hospital, existing dedicated efforts are mostly focused on extracting effective features and using accurate classifiers. In this section, we give a brief overview of research efforts done along early readmission prediction at hospital.

As readmission act as a substantial contributor of rising healthcare costs, predicting readmission has been identified

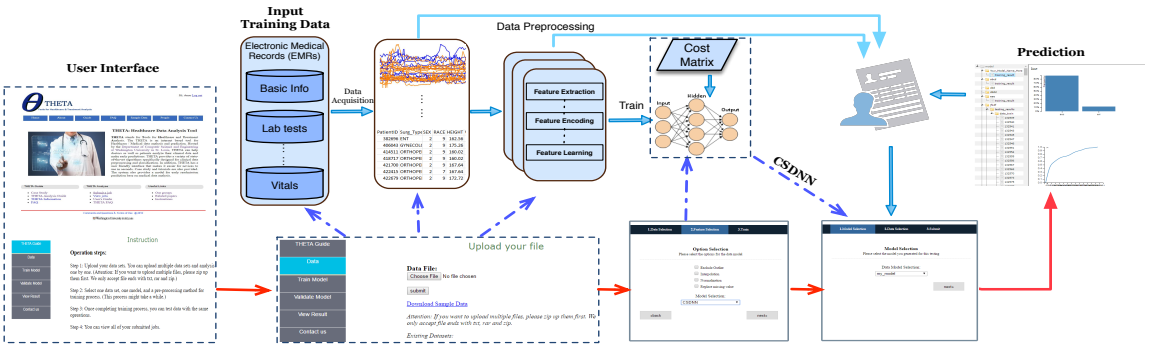

Fig. 7. An illustration of the system workflow. There are four key components in the system, including data acquisition, data preprocessing, model selection, and prediction. The system has user-friendly interfaces and detailed user guide. Physicians can follow the steps and the case study on the guide pages to predict readmission.

as one of the key problems for the healthcare domain. However, there are not many solutions known to be effective. He et al. [10] present a data-driven method to predict hospital readmission solely on administrative claims data. Nevertheless, their method is unable to incorporate clinical laboratory data in the model and as a result is not able to directly compare its performance with other approaches. Applying a comprehensive dataset that make generalization more reasonable. Therefore, [2], [6], [19] leverage a variety of data sources, including patient demographic and social characteristics, medications, procedures, conditions, and lab tests. However, they used the features designed for specific disease. Some conventional modeling techniques, such as support vector machine (SVM) or logistic regression are widely used for classification problems [26], [27]. [13], [16], [21] come up with more general statistical features used for predicting readmission with conventional modeling techniques. All of these methods relies on feature extraction and the ability of classifiers, which limit the performance of their methods.

To date, previous works on early readmission prediction by extracting statistical features from vital signs are inefficient feature representing methods, since they are hard to capture temporal patterns present in longitudinal time series data. Choi et al. [5] show deep learning models outperform the traditional modeling techniques in medical domain, and deep learning can be interpretable for healthcare analysis [4]. However, these works based on deep learning fail to consider the imbalanced data problem.

9 Conclusions

Readmission is a major source of cost for healthcare systems. Readmission not only degrades the quality of health care but also increases medical expenses. In this paper, we aim to identify those patients who are likely to be readmitted to the hospital. The identified patients can then be considered by health care personnel for application of preventive alternative measures. The goal is to deliver superior prediction quality, with good interpretability and high computational efficiency, that supports early readmission prediction.

Deep learning has been one of the most prominent machine learning techniques nowadays. Deep learning makes possible automatic feature learning from medical data. We propose to use both traditional statistical features via categorical feature embedding and learnt features via convolutional neural networks as input to a multilayer perceptron. This way can utilize the advantage of local information, temporal information and overall trends in vital signs time series. However, imbalance or skewed class distribution are challenges in medical data. For most cases, the recognition importance of positive instances is higher than that of negative instances. Therefore, we further propose a cost-sensitive deep learning model to address the imbalanced problem on medical data. The effectiveness of the proposed approach is validated on two real medical datasets from Barnes-Jewish Hospital. Our performance is good enough to warrant an actual clinical trial in hospital setting. Consequently, our model has been deployed in a real system for readmission prediction.

Acknowledgments

The work is supported in part by the DBI-1356669, SCH-1343896, III-1526012, and SCH-1622678 grants from the National Science Foundation of the United States.

References

[1] MM Al Rahhal, Yakoub Bazi, Haikel AlHichri, Naif Alajlan, Farid Melgani, and RR Yager. Deep learning approach for active classification of electrocardiogram signals. *Information Sciences*, 345:340–354, 2016.

[2] Waheeda Almayyan. Lymph diseases prediction using random forest and particle swarm optimization. *Journal of Intelligent Learning Systems and Applications*, 8(03):51, 2016.

[3] Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1721–1730. ACM, 2015.

[4] Edward Choi, Mohammad Taha Bahadori, Jimeng Sun, Joshua Kulas, Andy Schuetz, and Walter Stewart. Retain: An interpretable predictive model for healthcare using reverse time attention mechanism. In *Advances in Neural Information Processing Systems*, pages 3504–3512, 2016.

[5] Edward Choi, Andy Schuetz, Walter F Stewart, and Jimeng Sun. Using recurrent neural network models for early detection of heart failure onset. *Journal of the American Medical Informatics Association*, page ocw112, 2016.

[6] Shahid A Choudhry, Jing Li, Darcy Davis, Cole Erdmann, Rishi Sikka, and Bharat Sutariya. A public-private partnership develops and externally validates a 30-day hospital readmission risk prediction model. *Online journal of public health informatics*, 5(2):219, 2013.
[7] Zhicheng Cui, Wenlin Chen, and Yixin Chen. Multi-scale convolutional neural networks for time series classification. *arXiv preprint arXiv:1603.06995*, 2016.
[8] Joseph Futoma, Jonathan Morris, and Joseph Lucas. A comparison of models for predicting early hospital readmissions. *Journal of biomedical informatics*, 56:229–238, 2015.
[9] John Cristian Borges Gamboa. Deep learning for time-series analysis. *arXiv preprint arXiv:1701.01887*, 2017.
[10] Danning He, Simon C Mathews, Anthony N Kalloo, and Susan Hutfless. Mining high-dimensional administrative claims data to predict early hospital readmissions. *Journal of the American Medical Informatics Association*, 21(2):272–279, 2014.
[11] WU Hongjie, CAO Chengyuan, XIA Xiaoyan, and LÜ Qiang. Unified deep learning architecture for modeling biology sequence. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2017.
[12] Stephen F Jencks, Mark V Williams, and Eric A Coleman. Rehospitalizations among patients in the medicare fee-for-service program. *N Engl J Med*, 360:1418–28, 2009.
[13] Sujin Kim, Woojae Kim, and Rae Woong Park. A comparison of intensive care unit mortality prediction models through the use of data mining techniques. *Healthcare informatics research*, 17(4):232–243, 2011.
[14] Eun Whan Lee. Selecting the best prediction model for readmission. *Journal of Preventive Medicine and Public Health*, 45(4):259, 2012.
[15] Xu-Ying Liu, Jianxin Wu, and Zhi-Hua Zhou. Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(2):539–550, 2009.
[16] Yi Mao, Wenlin Chen, Yixin Chen, Chenyang Lu, Marin Kollef, and Thomas Bailey. An integrated data mining approach to real-time clinical monitoring and deterioration warning. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1140–1148. ACM, 2012.
[17] Robert Moskovitch, Hyunmi Choi, George Hripcsak, and Nicholas P Tatonetti. Prognosis of clinical outcomes with temporal patterns and experiences with one class feature selection. *IEEE/ACM transactions on computational biology and bioinformatics*, 14(3):555–563, 2017.
[18] Moonshik Shin, Dongjin Jang, Hojung Nam, Kwang Hyung Lee, and Doheon Lee. Predicting the absorption potential of chemical compounds through a deep learning approach. *IEEE/ACM transactions on computational biology and bioinformatics*, 2016.
[19] Sriram Somanchi, Samrachana Adhikari, Allen Lin, Elena Eneva, and Rayid Ghani. Early prediction of cardiac arrest (code blue) using electronic medical records. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2119–2126. ACM, 2015.
[20] Jelena Stojanovic, Djordje Gligorijevic, Vladan Radosavljevic, Nemanja Djuric, Mihajlo Grbovic, and Zoran Obradovic. Modeling healthcare quality via compact representations of electronic health records. *IEEE/ACM transactions on computational biology and bioinformatics*, 14(3):545–554, 2017.
[21] Shanu Sushmita, Garima Khulbe, Aftab Hasan, Stacey Newman, Padmashree Ravindra, Senjuti Basu Roy, Martine De Cock, and Ankur Teredesai. Predicting 30-day risk and cost of" all-cause" hospital readmissions. In *Workshops at the Thirtieth AAAI Conference on Artificial Intelligence*, 2016.
[22] Christian Szegedy. An overview of deep learning. *AITP 2016*, 2016.
[23] Ying P Tabak, Xiaowu Sun, Carlos M Nunez, Vikas Gupta, and Richard S Johannes. Predicting readmission at early hospitalization using electronic clinical data: An early readmission risk score. *Medical care*, 55(3):267, 2017.
[24] Yujin Tang, Jianfeng Xu, Kazunori Matsumoto, and Chihiro Ono. Sequence-to-sequence model with attention for time series classification. In *Data Mining Workshops (ICDMW), 2016 IEEE 16th International Conference on*, pages 503–510. IEEE, 2016.
[25] Lior Turgeman and Jerrold H May. A mixed-ensemble model for hospital readmission. *Artificial intelligence in medicine*, 72:72–82, 2016.
[26] Haishuai Wang and Jun Wu. Boosting for real-time multivariate time series classification. In *AAAI*, pages 4999–5000, 2017.
[27] Haishuai Wang, Peng Zhang, Xingquan Zhu, Ivor Wai-Hung Tsang, Ling Chen, Chengqi Zhang, and Xindong Wu. Incremental subgraph feature selection for graph classification. *IEEE Transactions on Knowledge and Data Engineering*, 29(1):128–142, 2017.
[28] Lin Zhang, Hui Liu, Yufei Huang, Xuesong Wang, Yidong Chen, and Jia Meng. Cancer progression prediction using gene interaction regularized elastic net. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 14(1):145–154, 2017.
[29] Yi Zheng, Qi Liu, Enhong Chen, Yong Ge, and J Leon Zhao. Time series classification using multi-channels deep convolutional neural networks. In *International Conference on Web-Age Information Management*, pages 298–310. Springer, 2014.
[30] Shang-Ming Zhou, Fabiola Fernandez-Gutierrez, Jonathan Kennedy, Roxanne Cooksey, Mark Atkinson, Spiros Denaxas, Stefan Siebert, William G Dixon, Terence W ONeill, Ernest Choy, et al. Defining disease phenotypes in primary care electronic health records by a machine learning approach: a case study in identifying rheumatoid arthritis. *PloS one*, 11(5):e0154515, 2016.

Haishuai Wang is a research fellow at the Department of Biomedical Informatics at Harvard University. Before Harvard, he was a postdoctoral associate at the Department of Computer Scient and Engineering at Washington University in St. Louis. He received his Ph.D. degree in computer science from the Center of Artificial Intelligence at University of Technology Sydney, Sydney, Australia. His research focuses on data mining and machine learning as well as its applications on bioinformatics and social networks. He is an Associate Editor for IEEE Access, and Program Committee (PC) of the SIAM Data Mining Conference 2018 (SDM 2018) and NIPS ML4H 2017.

Zhicheng Cui is a Ph.D. student at the Department of Computer Science and Engineering at Washington University in St Louis. He received his B.E. degree from School of Computer Science and Technology at The University of Science and Technology of China in 2014. His research interests are deep learning, interpretable machine learning and security of machine learning. He received the Best Paper Award at the American Medical Informatics Annual Fall Symposium (AMIA-15), 2015.

Yixin Chen is a Professor of Computer Science at the Washington University in St. Louis. His research interests include data mining, machine learning, artificial intelligence, optimization, and cyber-physical systems. He received a Ph.D. in Computing Science from University of Illinois at Urbana Champaign in 2005. He received the Best Paper Award at the AAAI Conference on Artificial Intelligence (2010) and International Conference on Tools for AI (2005), and best paper nomination at the ACM KDD Conference (2009). His work on planning has won First Prizes in the International Planning Competitions (2004 & 2006). He has received an Early Career Principal Investigator Award from the Department of Energy (2006) and a Microsoft Research New Faculty Fellowship (2007). He is an Associate Editor for ACM Transactions of Intelligent Systems and Technology and IEEE Transactions on Knowledge and Data Engineering, and serves on the Editorial Board of Journal of Artificial Intelligence Research.

Michael Avidan is a Professor of Anesthesiology and Surgery at Washington University. He is also the Division Chief of Cardiothoracic Anesthesiology and Cardiothoracic Intensive Care. He received Diploma in Anesthesia (DA) from South African College of Medicine and MB-BCh from University of Witwatersrand, South Africa. His research interests include intraoperative awareness, postoperative cognitive alteration, brain monitoring during surgery, and postoperative outcomes.

Arbi Ben Abdallah is a Research Assistant Professor at the Washington University in St. Louis. Dr. Ben Abdallahs research interests include the assessment of human behavior when facing risk, and analyzing the cost effectiveness of medical interventions.

Alex Kronzer received a B.A. from St Olaf College studying mathematics and economics. He is a database analyst for Washington University in St. Louis where he assists the Anesthesiology Department with clinical research projects.