IEEE Journal of Translational Engineering in Health and Medicine

Received 11 August 2023; revised 21 November 2023; accepted 13 December 2023.
Date of publication 18 December 2023; date of current version 26 December 2023.

Digital Object Identifier 10.1109/JTEHM.2023.3344035

Contrastive Transfer Learning for Prediction of Adverse Events in Hospitalized Patients

HOJJAT SALEHINEJAD[1,2], (Senior Member, IEEE), ANNE M. MEEHAN[3], PEDRO J. CARABALLO[3,4], AND BIJAN J. BORAH[1]

[1]Kern Center for the Science of Health Care Delivery, Mayo Clinic, Rochester, MN 55905, USA
[2]Department of Artificial Intelligence and Informatics, Mayo Clinic, Rochester, MN 55905, USA
[3]Department of Medicine, Mayo Clinic, Rochester, MN 55905, USA
[4]Department of Quantitative Health Sciences, Mayo Clinic, Rochester, MN 55905, USA

CORRESPONDING AUTHOR: H. SALEHINEJAD (hojjat@ieee.org)

ABSTRACT Objective: Deterioration index (DI) is a computer-generated score at a specific frequency that represents the overall condition of hospitalized patients using a variety of clinical, laboratory and physiologic data. In this paper, a contrastive transfer learning method is proposed and validated for early prediction of adverse events in hospitalized patients using DI scores. Methods and procedures: An unsupervised contrastive learning (CL) model with a classifier is proposed to predict adverse outcome using a single temporal variable (DI scores). The model is pretrained on an unsupervised fashion with large-scale time series data and fine-tuned with retrospective DI score data. Results: The performance of this model is compared with supervised deep learning models for time series classification. Results show that unsupervised contrastive transfer learning with a classifier outperforms supervised deep learning solutions. Pretraining of the proposed CL model with large-scale time series data and fine-tuning that with DI scores can enhance prediction accuracy. Conclusion: A relationship exists between longitudinal DI scores of a patient and the corresponding outcome. DI scores and contrastive transfer learning can be used to predict and prevent adverse outcomes in hospitalized patients. Clinical impact: This paper successfully developed an unsupervised contrastive transfer learning algorithm for prediction of adverse events in hospitalized patients. The proposed model can be deployed in hospitals as an early warning system for preemptive intervention in hospitalized patients, which can mitigate the likelihood of adverse outcomes.

INDEX TERMS Contrastive learning, deterioration index, early warning system, transfer learning.

I. INTRODUCTION

PROACTIVELY identifying signs of deterioration in hospitalized patients is critical for preventing morbidity and mortality. Early warning systems in hospitals, as supportive tools for medical decision-making, can potentially avert adverse events including cardiac arrests, rapid response calls, resuscitation, intensive care unit (ICU) transfers, and death.

Epic's Deterioration Index (DI) is one of the most widely used early warning systems deployed in hundreds of hospitals across the United States [1], [2]. This system aims to detect patients who deteriorate and require higher levels of care. The deterioration index (DI) score ranges from 0 to 100, wherein patients are deemed low ($\leq$ 30), intermediate (30 $-$ 60) or high-risk ($\geq$ 60). The high-risk patients are at the greatest risk of encountering a composite adverse outcome which can be prevented by prompt interventions. This has been found to have fair performance and improve patient outcomes and reduce ICU admissions [2].

The DI system is currently running in many hospitals. Clinical providers use an absolute number threshold ($\geq$ 60) to determine if a patient may experience an adverse event and need immediate intervention. Health systems utilize the DI score in conflicting ways and with substantially disparate thresholds [1], [3]. The clinical providers' observation is that some patients may experience an adverse event without nearing a high-risk DI ($\geq$ 60), whereas others may reach a critical DI score and not have an adverse event. However, no prior peer-reviewed studies have used machine learning to demonstrate a relationship between DI score patterns and

patient outcome as well as the ability to predict adverse outcomes.

Machine learning has demonstrated superior performance in many real-world applications such as COVID-19 lung detection prognosis using chest computerized tomography (CT) scans [4], in-hospital mortality prediction among diabetes ICU patients [5], cervical spine fracture detection on CT scans [6], human activity recognition [7], synthesizing pathology in chest X-rays images [8], and identification of metastatic breast cancer [9]. The implementation of supervised machine learning in real-world healthcare settings presents inherent challenges, primarily attributed to the scarcity of large-scale annotated datasets. This challenge becomes particularly pronounced due to the insufficient number of samples available for rare conditions and events (e.g., adverse events). Moreover, the process of annotating large healthcare datasets for practical applications is a resource-intensive and costly endeavor. Unsupervised and self-supervised learning can perform clustering and prediction tasks with unlabeled data. However, training of such models is very challenging due to complex pattern of data, particularly in a high-dimensional feature space. Contrastive learning (CL) is one of the most popular approaches under self-supervised learning, which generates augmentations of the input to diversify representations [10], [11], [12].

Exploiting the time series nature of DI scores, this paper proposes a CL method for feature representation of DI scores in an unsupervised fashion. The features are then used to train a simple classifier for automated prediction of adverse events based on the retrospective DI scores of a hospitalized patient. As we will demonstrate, one of the key advantages of this approach is the utilization of transfer learning on large-scale unannotated data for enhanced feature representation by the CL model. This approach can enhance generalization performance of the CL model in feature representation from DI scores. In practice, the proposed system can be deployed on top of the Epic's DI system as a clinical assistive tool to alert care givers of potential risk of adverse events in hospitalized patients.

The contributions of this paper include the following. 1) Contrastive representation of DI score for prediction of adverse events in hospitalized patients; 2) study of transfer learning with general time series data in enhancing performance of contrastive learning in adverse event prediction; 3) demonstrating a direct relationship between the DI score patterns of hospitalized patients and their corresponding outcomes, solely relying on unsupervised contrastive and deep supervised models as non-linear functions, without the use of any additional predictors.

The remainder of this paper is organized as follows. Section II provides a background on DI score and contrastive representation learning. The proposed method for contrastive transfer learning from DI scores is discussed in Section III. The experiments and analysis are provided in Section IV. Translation to clinical practice is discussed in Section V and the paper is concluded in Section VI.

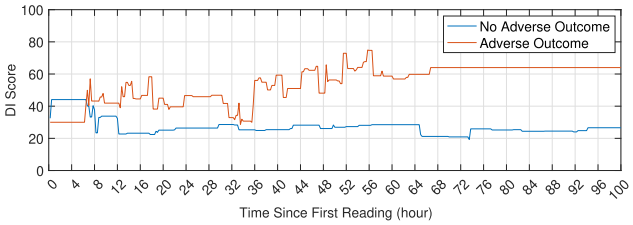


FIGURE 1. DI scores of two randomly selected patients with and without an adverse outcome. They had different length of stays, where the patient with an adverse outcome died approximately 68 hours after the first DI score reading. A DI score is regularly calculated at 15-minute intervals until discharge or death. The DI scores are plotted for 100 hour after the first DI score reading.

II. BACKGROUND

A. DETERIORATION INDEX

Adverse events in hospitals such as all-cause mortality, cardiac arrest, transfer to intensive care, and evaluation by the rapid response team has been estimated to be approximately 17% [13]. Epic DI generates a patient risk score immediately after hospital admission and it is then regularly calculated based on most recent available data at 15-minute intervals until discharge or death. Figure 1 shows two examples of the DI scores for two patients, one with an adverse outcome and the other without an adverse outcome. The risk score is calculated and then updated at 15-minute interval based on routinely recorded physiological, clinical, and laboratory parameters within Epic's electronic health record. Some of the predictors in determining DI score are age, nursing assessments (neurological assessment, cardiac rhythm, oxygen requirement, Glasgow Coma Scale), vital sign measurements (temperature, systolic blood pressure, pulse, oxygen saturation, respiratory rate), and laboratory values (hematocrit, white blood cell count, blood urea nitrogen, potassium, sodium, blood pH, platelet count) [1].

Limited research has been conducted on the DI score and its relationship with adverse outcomes in clinical settings. DI scores were augmented with chest radiographs in [2] to predict clinical deterioration (death or the need for ICU-level therapies including invasive or non-invasive mechanical ventilation, heated high flow nasal cannula, or vasopressor support) for hospitalized patients with acute dyspnea. The general objective was to improve accuracy of the DI score by incorporating patients' imaging data, who required supplemental oxygen during their hospitalization and had at least one chest radiograph performed during the first 48 hours. This study excluded patients that experienced clinical deterioration or discharge within four hours of presentation or experienced clinical deterioration prior to their first radiograph. The proposed machine learning model was completely supervised. The imaging features were extracted by training a DenseNet-121, pre-trained on the CheXpert [14] and MIMIC-DICOM [15] data. The combination of imaging features with deterioration index scores and time-dependent variables were used to train a feed-forward neural network with a single hidden layer of five nodes and

a single-prediction output. It is shown that incorporation of imaging data can increase accuracy in identifying patient with acute dyspnea at low risk of experiencing an adverse outcome.

B. CONTRASTIVE REPRESENTATION LEARNING

The most common machine learning approach for classification applications is based on supervised learning, which generally requires large-scale labelled datasets. However, many datasets in healthcare are limited-imbalanced in nature, which means limited number of data samples are available per some or all data classes [16]. In addition, it is time consuming and expensive to annotate large-scale datasets for training complex machine learning models, particularly deep supervised learning models. Unsupervised learning targets representation and clustering of data samples based on the underlying pattern in the data without utilizing the corresponding labels.

Contrastive learning (CL) has gained significant prominence as a leading approach in the field of unsupervised learning, with a particular focus on self-supervised learning [12]. CL methods have demonstrated notable performance in many applications such as diagnosis of Alzheimer's disease using brain positron emission tomography (PET) [17], human activity recognition [18], [19], tissue segmentation in histopathological images [20], [21], whole slide image classification [22], ultrasound images analysis [23], underwater image enhancement [24], medical image segmentation [25], and optical coherence tomography (OCT) fluid segmentation [26]. For various applications, CL in self-supervised learning has outperformed supervised learning [12], [27], [28].

The main idea behind CL methods is to diversify representation of an input by generating augmentations, compared to augmentations of the other inputs [11], [12]. This process is generally conducted by mapping similar sample pairs into a feature embedding space such that the similar samples are closer to each other and the dissimilar ones are far away. As a simple example, let two different augmentations of a sample input, which belong to the same class, be considered as positive samples. A sample from another input, which can be positive or negative, is then compared with the positive samples. The CL loss function adjust the distance between the feature vectors based on the similarity or dissimilarity of the samples. Selection or augmentation of the positive pairs is one of the major steps in CL, which distinguishes contribution of various CL methods from each other. It is possible to construct different augmentations of an input time series or image using various sampling techniques. In time series, representations with the same timestamp from two overlapping views of an input time series can be considered as positive, while those at different timestamps from the same time series as negatives [29]. This approach uses a mixed instance-wise and temporal-wise CL approach, where negative samples can be selected from the same instance and from other instances.

FIGURE 2. A DI score $x_{n,t}$ such that $t \in (1, \ldots, T_n)$ is generated at every 15-minute interval for a patient n. DI scores are collected 5.5 hours after admission until the event (adverse/no adverse).

In computer vision augmentation, techniques such as random rotating, cropping, and flipping of the different views of an input [12], [30] can generate different positive and negative pairs at instance and spatial levels.

III. METHODS AND PROCEDURES

This study was reviewed and approved by the Institutional Review Board. We propose using an unsupervised encoder with hierarchical CL for feature extraction from DI scores followed by a simple classifier.

A. DEFINITIONS

Let $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$ represent a set of retrospective data samples from N subjects such that $\mathbf{x}_n = (x_{n,1}, \ldots, x_{n,T_n})$ is the DI scores, where $x_{n,t} \in [0, 100]$ and $y_n \in \{0, 1\}$ is the outcome, where T_n is the number of DI scores for subject $n \in \{1, \ldots, N\}$. We model each DI score collection as a time series as illustrated in Figure 2. For a subject with no adverse outcome, $y_n = 0$, and for one with an adverse outcome, $y_n = 1$.

B. RULE-BASED METHOD

Clinical providers use an absolute number threshold η to determine if a patient may experience an adverse event and need immediate intervention. Below, two rule-based protocols based on this threshold are discussed.

Rule-based any: In this protocol, if at any time $t \in (0, 1, \ldots, T_n)$ a DI score value $x_{n,t}$ passes the threshold η for patient n, the predicted outcome is perceived as an AE, defined as

$$\tilde{y}_n = \begin{cases} 1 & \text{if } x_{n,t} \geq \eta \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

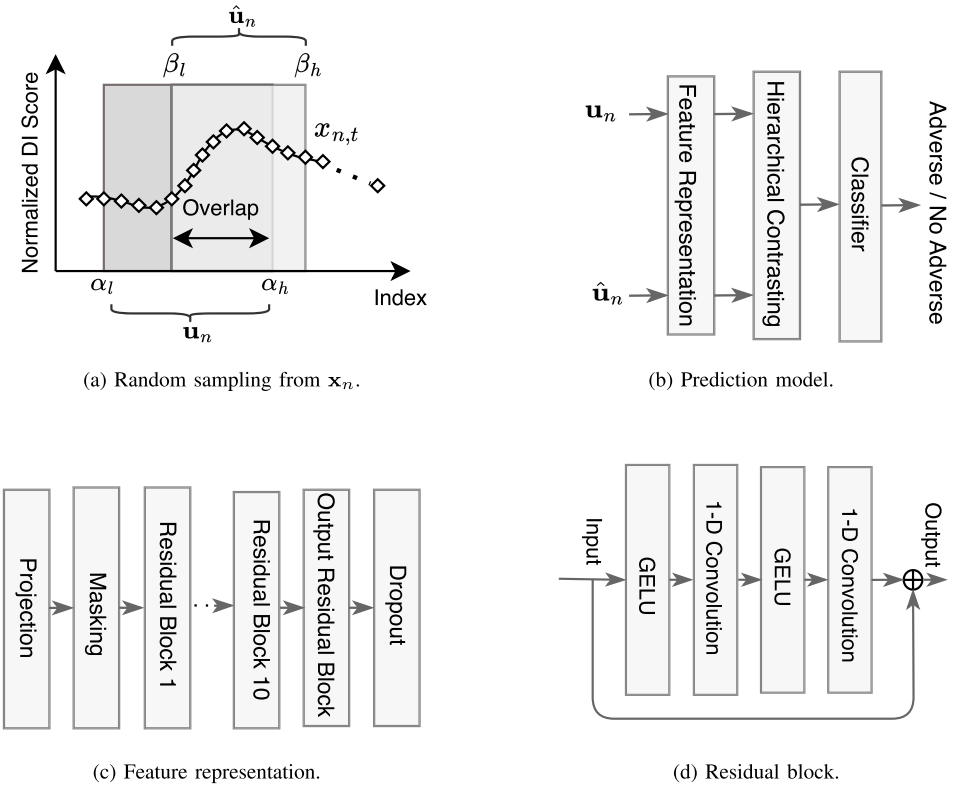
where $\tilde{y}_n$ is the predicted outcome.

Rule-based last: Within this protocol, if the last DI score x_{n,T_n} of patient n passes the threshold η, the predicted outcome is considered as an AE, defined as

$$\tilde{y}_n = \begin{cases} 1 & \text{if } x_{n,T_n} \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

C. CONTRASTIVE LEARNING METHOD

A DI score series $\mathbf{x}_n$ is normalized and sampled by randomly generating two overlapping cropping intervals $[\alpha_l, \alpha_h]$ and $[\beta_l, \beta_h]$ to select the sequential subsets of the DI scores $\mathbf{u}_n$ and $\hat{\mathbf{u}}_n$, respectively, such that $0 < \alpha_l \leq \beta_l \leq \alpha_h \leq \beta_h \leq T_n$,



(a) Random sampling from $\mathbf{x}_n$.

(b) Prediction model.

(c) Feature representation.

(d) Residual block.

FIGURE 3. The CL model for feature extraction from DI scores.

as demonstrated in Figure 3(a). The contextual representations on the overlapped segment in the interval $[\beta_l, \alpha_h]$ should be consistent along the time for two context reviews.

As Figures 3(b) and 3(c) show, the sample vectors $\mathbf{u}_n$ and $\hat{\mathbf{u}}_n$ are passed to a feature representation module which projects the vectors at timestamp t as

$$\mathbf{z}_{n,t} = \mathbf{w}u_{n,t} + \mathbf{b}, \tag{3}$$

and

$$\hat{\mathbf{z}}_{n,t} = \mathbf{w}\hat{u}_{n,t} + \mathbf{b}, \tag{4}$$

where $\mathbf{w}, \mathbf{b} \in \mathbb{R}^{F_p}$ and F_p is the dimensionality of the feature space. A random binary mask is applied to the represented features to generate an augmented context view as

$$\mathbf{z}_{n,t} := \mathbf{z}_{n,t} \odot \mathbf{m}, \tag{5}$$

and

$$\hat{\mathbf{z}}_{n,t} := \hat{\mathbf{z}}_{n,t} \odot \mathbf{m}, \tag{6}$$

where $\odot$ is the element-wise multiplication, $\mathbf{m} \in \mathbb{Z}_{0,1}^{F_p}$, and $m_i \sim$ Bernoulli(0.5). Similar to the dropout regularization method in training deep neural networks [31], the timestamp masking and random cropping are only functional during training.

The masked features are then passed to a series of 10 residual convolution blocks where the block $l \in (1, \ldots, 10)$ has two 1-D convolution layers with Gaussian error linear units (GELUs) [32], kernel size k, channel size F_l, and a dilation of 2^l, Figure 3(d). The output residual block maps the extracted features to a higher dimensional space of $F_o > F_l$ defined as $\mathbf{r}_{n,t}$ and $\hat{\mathbf{r}}_{n,t}$ for (5) and (6), respectively. The features are then passed to a dropout layer. We use a temporal contrastive loss to learn discriminative representations over time, defined as

$$\mathcal{T}_{n,t} = -log \frac{e^{\mathbf{r}_{n,t} \cdot \hat{\mathbf{r}}_{n,t}}}{\Sigma_{t'=1}^{T'} (e^{\mathbf{r}_{n,t} \cdot \hat{\mathbf{r}}_{n,t'}} + \mathbb{1}_{[t \neq t']} e^{\mathbf{r}_{n,t} \cdot \mathbf{r}_{n,t'}})}, \tag{7}$$

where T' is the length of the overlap between $\mathbf{u}_n$ and $\hat{\mathbf{u}}_n$ and $\mathbb{1}$ is the indicator function [29]. An instance-wise contrastive loss is also defined as

$$\mathcal{E}_{n,t} = -log \frac{e^{\mathbf{r}_{n,t} \cdot \hat{\mathbf{r}}_{n,t}}}{\Sigma_{j=1}^{B} (e^{\mathbf{r}_{j,t} \cdot \hat{\mathbf{r}}_{n,t}} + \mathbb{1}_{[n \neq j]} e^{\mathbf{r}_{n,t} \cdot \mathbf{r}_{j,t}})}, \tag{8}$$

where B is the batch size. Finally, the overall loss of the hierarchical contrasting is defined as

$$\mathcal{L} = \frac{1}{N \cdot T} \sum_{n=1}^{N} \sum_{t=1}^{T} (\mathcal{T}_{n,t} + \mathcal{E}_{n,t}). \tag{9}$$

The loss function is then minimized using an optimization algorithm as discussed in Section IV-E.

The trained encoder using (9) generates a feature vector $\mathbf{r}_{n,t} \in \mathbb{Z}^{F_o}$ at timestamp and encounter levels. Hence, for an encounter n over all the timestamps, the features representation is

$$\mathbf{R}_n = (\mathbf{r}_{n,1} \oplus \ldots \oplus \mathbf{r}_{n,T}), \tag{10}$$

where $\oplus$ is for concatenation and $\mathbf{R}_n \in \mathbb{Z}^{F_o,\times T}$. The final feature vector per encounter is

$$a_{n,i} = \max_{t\in\{1,\ldots,T\}} (r_{n,i,t}), \tag{11}$$

which is computed recursively for $i = (1, \ldots, F_o)$. Hence, $\{\mathbf{a}_1, \ldots, \mathbf{a}_N\}$ is the set of contrastive feature representations of the DI scores $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$. The set of features can then be used along with the corresponding data class labels to train a supervised classification model $\Phi(\cdot)$ as

$$P(y_n|\mathbf{a}_n) = \Phi(\mathbf{a}_n, y_n), \tag{12}$$

where $P(y_n = 1|\mathbf{a}_n)$ is the predicted probability for the encoutner n to experience an adverse outcome given the DI scores $\mathbf{x}_n$ such that $P(y_n = 0|\mathbf{a}_n) + P(y_n = 1|\mathbf{a}_n) = 1$. The predicted outcome is $\tilde{y}_n = 1$ if $P(y_n = 1|\mathbf{a}_n) \geq \theta$ and $\tilde{y}_n = 0$ otherwise, where θ is the cutoff probability by default set to 0.5. The optimal value of θ is discussed in the Section IV-D.

IV. RESULTS

A. DATA

The dataset was collected from adult patients ($\geq$ 18 years old), hospitalized in medical or surgical service at the Mayo Clinic, Rochester, MN, USA, during 8-23-2021 to 3-31-2022. For each occurrence, the first DI score was collected 5.5 hours (the buffer time in Figure 2) after admission until the outcome to ensure enough number of samples are available in the electronic health record (EHR) system and for the predictive model. There were a total of 25, 127 encounters with a distribution of 22, 325 ($\approx$ 88.85%) encounters without adverse event and 2, 802 ($\approx$ 11.15%) with adverse event. Similar to many other clinical applications such as intracranial hemorrhage detection [42], [43] and pathology detection in X-ray images [8], [14], the natural distribution of this dataset over the data classes is imbalanced. Hence, the following protocols were implemented for training and testing of the models.

1) TEST DATA

For each cross-validation fold, a balanced test dataset is randomly selected from the dataset, where 10% was selected from the data class with adverse event and 10% was selected from the data class with no adverse event.

2) TRAINING DATA

After selecting the test dataset, the remaining dataset is naturally imbalanced. Hence, for each cross-validation fold, a balanced training dataset is selected by incorporating the remaining encounters from the adverse event data class with the number of encounters randomly selected from the no adverse event data class. The concatenation of samples was shuffled before each training process.

3) PRE-TRAINING DATA

The large-scale UCR time series classification archive [44], containing 128 datasets, was used to pre-train the CL model.

TABLE 1. Distribution of the original dataset and selected train and test datasets by random sampling from the original dataset per independent cross-validation fold. 10% of the training data (504 samples) is used as the validation dataset for hyperparameters tuning.

Data Class	Original Distribution	Sampled from Original	
		Test	Train
Adverse Event	2802	280	2522
No Adverse Event	22325	280	2522
Total	**25127**	**560**	**5044**

The model was pre-trained with the train and test datasets in an unsupervised fashion.

B. VALIDATION

1) METRICS

The machine learning models were evaluated using different metrics. The accuracy (Acc) performance metrics is defined as

$$Acc = \frac{TP + TN}{P + N}, \tag{13}$$

where TP is the true positive value, TN is the true negative value, P is the number of real encounters with adverse event, and N is the number of real encounters without adverse event. Since the test dataset is balanced, the accuracy metric is equivalent to the balanced accuracy. F1 Score is defined as

$$F1 = \frac{2 \times TP}{2 \times TP + FP + FN}, \tag{14}$$

where FP is the test encounter which is incorrectly predicted as adverse event and FN is the test encounter which is incorrectly predicted as having no adverse event. Specificity or true negative rate (TNR) is defined as

$$TNR = \frac{TN}{TN + FP}, \tag{15}$$

and sensitivity or true positive rate (TPR) is defined as

$$TPR = \frac{TP}{TP + FN}. \tag{16}$$

2) CROSS-VALIDATION

A 10-fold cross-validation was conducted and the average (Avg.) and standard deviation (Std.) of each performance metric were collected. In each independent run, the models were trained from scratch using a randomly selected training dataset and evaluated using a randomly selected balanced test dataset. A summary is provided in Table 1.

C. MODELS

1) CONTRASTIVE LEARNING

The CL models was trained as an encoder in an unsupervised fashion. The extracted features were used to train and evaluate Ridge regression [37], support vector machines (SVM) [38], random forest (RF) [39], Adaptive Boosting (AdaBoost) [40], and XGBoost [41] classifiers. In one setup, the CL model was trained from scratch using the DI score

TABLE 2. Performance of the rule-based protocol as baseline using a threshold of 60 in percentage. The results are averaged over 10-fold cross-validation.

Model	Performance Metric (Avg.±Std.)				
	Acc	PPV	TNR	TPR	F1-Score
Rule-based any	56.57±1.00	54.02±1.00	91.02±0.00	21.03±3.00	50.03±2.00
Rule-based last	52.20±1.00	50.83±1.00	99.00±0.90	3.20±1.00	36.42±2.00

TABLE 3. Performance of the supervised deep learning models as baseline. The results are averaged over 10-fold cross-validation. Values are in percentage.

Model	Performance Metric (Avg.±Std.)				
	Acc	PPV	TNR	TPR	F1-Score
LSTM-FCN [33]	71.64±3.96	71.48±5.80	70.29±8.45	73.00±3.17	72.09±2.94
GRU-FCN [34]	72.50±4.10	**75.74±3.26**	**78.00±8.01**	67.00±14.25	70.28±7.23
InceptionTime [35]	73.29±4.75	72.96±3.40	72.86±3.15	73.71±8.84	73.21±5.96
XceptionTime [36]	**74.86±3.24**	74.72±3.51	74.43±4.47	**75.29±4.03**	**74.96±3.20**

dataset. In another setup, it was pre-trained using the UCR data and then fine-tuned with the DI score dataset.

2) BASELINE DEEP SUPERVISED LEARNING

For sake of comparison with supervised methods as a baseline, the long-short-term memory fully convolutional network (LSTM-FCN) [33], gated recurrent unit fully convolutional network (GRU-FCN) [34], InceptionTime [35], and XceptionTime [36] models were trained and evaluated. These models were trained from scratch in a supervised manner using the dataset discussed in Subsection IV-A.

D. TRAINING SETTINGS

The hyperparameter tuning was conducted using 10% of the training data as the validation dataset, which was different from the test dataset. For the contrastive encoder, the number of projection features was $F_p = 64$, the number of features per layer was $F_l = 64$ for $l \in (1, 2, \ldots, 10)$, the number of output features was $F_o = 320$, kernel size was $k = 3$, the learning rate was 0.001 (grid searched in $\{0.0001, 0.001, 0.01, 0.1\}$) with the Adam optimizer [45], and the batch size was $B = 8$. The unsupervised encoder was trained on the training data using the hierarchical contrasting [29]. The dropout rate in the last layer of feature extraction phase was 0.1. The SVM [38] model was built with a radial basis function (RBF) with a regularization parameter of 0.1 (grid searched in $\{0.001, 0.01, 0.1, 1, 10\}$). The regularization parameter of the Ridge regression classifier was set to 0.01 after a grid search in $\{0.001, 0.01, 0.1, 1\}$.

Regarding the supervised deep learning models, the LSTM-FCN model has a single LSTM layer with 100 units and dropout rate of 0.8 and three 1-D convolution layers with 128, 256, and 128 kernels, batch normalization, and rectified linear unit (ReLU) activation function [33]. The learning rate for all the models was set to 0.01 after grid search with Adam optimizer and early-stopping was applied. The GRU-FCN model has a similar setup but with GRU instead of LSTM units. The InceptionTime and XceptionTime were tuned based on the recommendation in [35] and [36], respectively.

The hyper-parameter tuning was conducted using 10% of the training data (i.e. 504 samples) as the validation dataset, which was different from the test dataset. All the models were implemented in Python and PyTorch [46] and trained on two NVIDIA A6000 GPUs with 256GB of RAM and 64 CPU cores.

E. PERFORMANCE ANALYSIS

1) CLASSIFICATION PERFORMANCE

Table 2 shows the performance of baseline rule-based methods using the threshold $\eta = 60$, as discussed in Section III-B. The classification accuracy of *rule-based any* and *rule-based last* methods are 56.57% and 52.20%, respectively. The area under the curve (AUC) of the *rule-based any* model is 56.73% and for the *rule-based last* model is 51.67%.

Performance of the supervised deep learning models on the test dataset after 10-fold class validation is presented in Table 3. These models were trained as baseline for comparison with the CL approach. The results show that XceptionTime has a higher overall performance than the other models with an accuracy of 74.86% and F1-Score of 74.96%. The LSTM-FCT and FRU-FCN models have lower performance with F1-Score of 72.09% and 70.28%, respectively.

Classification performance results of the CL model with and without pre-training are presented in Table 4. The results on the test dataset show that the CL model, without pre-training, has better performance than the supervised learning models. Particularly, the combination of CL with XGBoost has an accuracy of the 79.79% which is about 5% better than the XceptionTime.

Pre-training the CL model with UCR dataset further improves the classification performance of the XGBoost classifier by about 2%. Similarly, it enhances the performance of the AdaBoost and Ridge regression classifiers about 4%.

Figure 4 shows the AUC of the baseline models in comparison with the CL model and pre-trained CL models. The XceptionTime model has an AUC of 0.82 which is the best performance among the supervised models. The CL model with XGBoost has an AUC of 0.86 and with

IEEE Journal of Translational Engineering in Health and Medicine

TABLE 4. Performance of the CL model with Ridge, SVM, RF, AdaBoost, and XGBoost classifiers. The CL model was evaluated with and without pre-training. The results are averaged over 10-fold cross-validation. Values are in percentage.

Model	CL Pre-Train	Performance Metric (Avg.±Std.)				
		Acc	PPV	TNR	TPR	F1-Score
Ridge [37]	N	75.57±0.96	77.76±1.88	79.43±2.49	71.71±1.72	74.59±0.90
	Y	**78.57±1.84**	77.65±1.83	76.86±2.51	80.29±3.26	**78.92±1.94**
SVM [38]	N	80.07±1.11	80.55±1.14	80.86±1.17	79.29±1.34	79.91±1.15
	Y	**81.43±1.96**	82.37±1.84	82.86±2.20	80.00±3.81	**81.13±2.25**
RF [39]	N	80.86±4.12	80.08±4.63	79.43±5.19	82.29±3.55	81.15±3.92
	Y	**81.64±1.57**	79.98±2.61	78.71±3.73	84.57±2.60	**82.07±1.42**
AdaBoost [40]	N	76.43±2.85	77.32±3.13	78.00±3.44	74.86±3.62	76.04±2.99
	Y	**80.71±2.22**	80.23±2.32	79.86±2.83	81.57±3.51	**80.86±2.29**
XGBoost [41]	N	79.79±2.01	79.84±2.25	79.71±3.66	79.86±5.75	79.73±2.56
	Y	**81.86±1.17**	80.92±1.62	80.29±2.24	83.43±2.39	**82.23±1.23**

RF has an AUC of 0.88. The RF model in Table 4 also shows a competitive performance with XGBoost due to its higher *TPR*. Pre-training of the CL models enhances the AUC values of all classifiers. Particularly, XGBoost trained on top of the contrastive transfer learning model has an AUC of 0.91, which is 0.05 higher than its counterpart without pre-trained CL.

Overall, the results show that unsupervised pre-training of the CL model with large-scale time series datasets can boost the classification accuracy of the models for prediction of adverse events.

2) CONTRASTIVE ENCODER LATENT SPACE ANALYSIS

One of the critical hyperparameters in tuning the CL model is the number of output features F_o. Figure 5 shows the prediction accuracy of the best model (i.e. contrastive transfer learning with XGBoost) for $F_o \in \{40, 80, 160, 320, 640\}$ after 10-fold cross-validation according to Section IV-B. For $F_o = 40$ and $F_o = 80$ the average performance and standard deviation are $69.23 \pm 3.8\%$ and $75.13 \pm 3.1\%$, respectively. Increasing the number of output features to $F_o = 320$ achieves the highest performance at $81.86 \pm 1.17\%$. Further increase of F_o to 640 slightly decrease the performance to $81.34 \pm 0.9\%$.

3) CONFIDENCE SCORE OF CLASSIFIER

Figure 4 shows the specificity and sensitivity plots of the CL model with XGBoost versus different probability cutoff values. The plots show the cutoff probability value of $\theta = 0.59$ and $\theta = 0.43$ for the CL model without pre-training and with pre-training, respectively. Pre-training of the CL model shifts θ towards the lower end of the probability cutoff which improves the TNR and TPR values of the model, Table 4.

Figure 7 shows the confidence score, which is the prediction probability in (12), of the CL and XGBoost classifier for the encounters in the test dataset. This plot shows the normalized count of encounters per data class and the corresponding prediction probability.

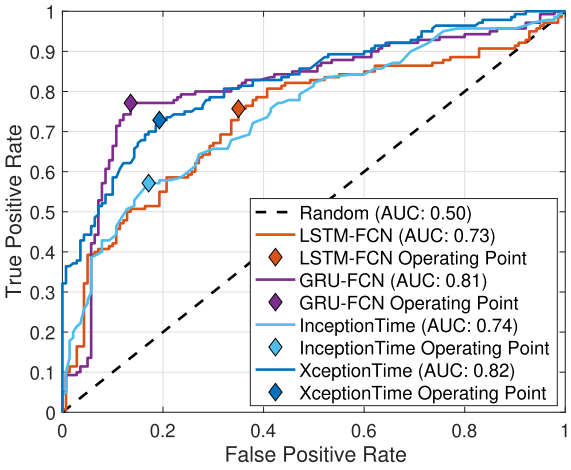
The first observation is that the classifier without pre-training has predicted 70.71% of the adverse event encounters correctly with a probability of $P \geq 0.9$. For the no adverse events this number is 58.57%. Similarly, the pre-trained CL model with XGBoost has predicted 71.43% of the encounters with an adverse event correctly with a probability of $P \geq 0.9$. For the no adverse events this number is 72.14% which is significantly ($\approx$ 14%) higher than the CL model without pre-training. This is also observable by comparing the cutoff thresholds in Figure 4, where threshold of pre-trained model is lower than the not pre-trained model which increases the *TPR* as shown in Table 4.

The second observation is the number of encounters that the models have predicted with low confidence. Particularly, the classifier without pre-training has predicted 7.86% of the adverse event encounters incorrectly with a confidence of $P \leq 0.1$. For the no adverse events this number is 15.00%. Similarly, the pre-trained model has predicted 13.57% of the encounters with an adverse event incorrectly with a confidence of $P \leq 0.1$, which is about 5% more than the not pre-trained model. For the no adverse events this number is 10.00% which is 5%) less than the CL model without pre-training.

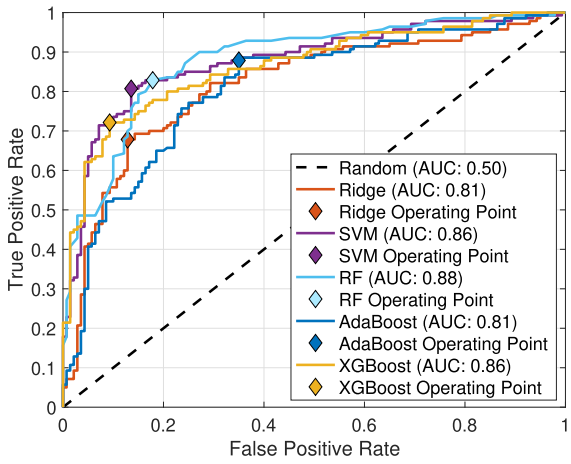
V. TRANSLATION TO CLINICAL PRACTICE

This study is originated from the necessity to enhance the current clinical efficacy of the DI score systems within hospitals, driven by feedback from clinicians. At present, the DI score is automatically updated every 15 minutes based on the latest variables information, visible on the Epic summary screen of the patients in the hospital. This screen shows hospital services with all the patients cared for by a provider listed in individual rows and patient specific data displayed in columns. Clinicians have observed situations in which patients with a critical DI score (≥ 60) did not encounter AEs, while others with AEs did not attain the critical threshold score.

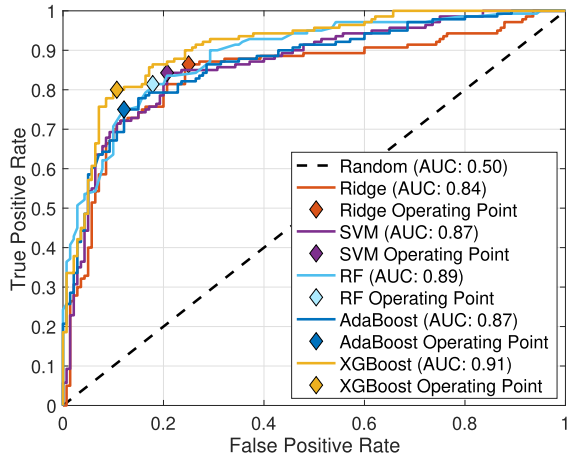
The future focus of the outlined CL approach is to compute and integrate the new CL model DI prediction in a column alongside the current DI score in the clinical workflow for our hospital users, including medical and nursing staff. Technically, existing institutional information technology resources can be leveraged along with expertise in implementing other ML models in our EHR to implement the proposed solution.



(a) Supervised deep models.



(b) CL without pre-training.



(c) CL with pre-training.

FIGURE 4. AUC plots of the supervised deep learning models and the CL model with and without pre-training.

The algorithm will operate using resources interfaced with our clinical data, and the outcome will be displayed within the clinical workflow as described earlier.

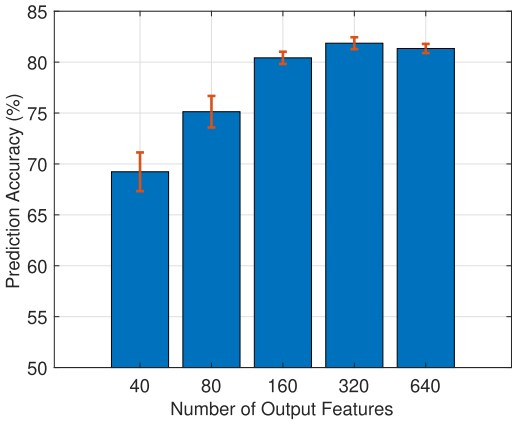
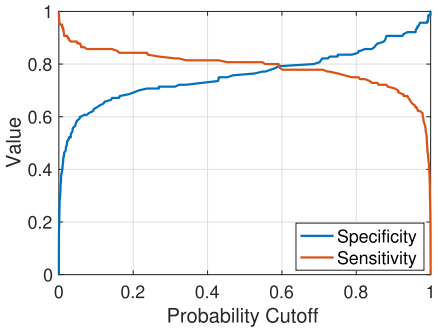
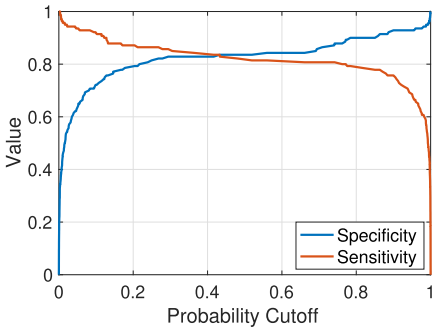


FIGURE 5. Prediction accuracy of the XGBoost classifier for different number of output features in the pre-trained CL encoder.



(a) Without pre-training.



(b) With pre-training.

FIGURE 6. Specificity and sensitivity of the CL and XGBoost model with respect to different probability cutoff values.

The initial implementation will involve a silent mode period to validate the model and ensure accuracy and safety. Prior to and during the initial implementation, providers and nurses will undergo appropriate education and training. The ultimate objective is for providers to utilize information from the DI score and our new CL model prediction in medical decision-making to identify patients at risk of clinical deterioration.

Evaluation of the new CL model's utility will encompass subjective feedback from clinical users and objective measurement of AEs and outcomes in hospital areas where the new CL model is employed, compared to control areas

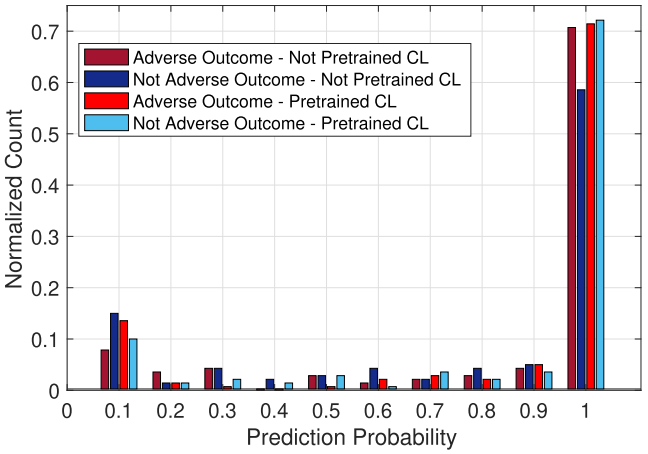


FIGURE 7. Normalized number of patients and corresponding confidence score (prediction probability) of the CL model with XGBoost classifier.

using the existing DI score. This implementation strategy is capable of easy generalization and potential adoption by other institutions.

A key enabler of clinical translation of any proposed intervention is its cost-effectiveness. Cost-effectiveness assessment during the validation phase of the model in the silent mode can be conducted. Specifically, the difference in the number of AEs averted between the standard of care (i.e. the rule-based method) versus our CL approach, can be utilized to quantify the saved cost associated with CL detected AEs.

VI. CONCLUSION

In this paper, for the first time in the literature, we propose an unsupervised CL approach with a classifier for prediction of adverse events in hospitalized patients using retrospective DI scores. The model was trained and validated on real-world data. In addition, its classification performance was compared with the baseline deep supervised time series classification models. The results indicated the CL approach with a classifier outperforms the supervised models. Furthermore, we demonstrated that contrastive transfer learning, involving pre-training the CL model with large-scale unlabeled data, enhances the prediction accuracy concerning patient outcomes. This model holds the potential for implementation as an assistive tool for care providers, enabling early intervention to mitigate adverse outcomes among hospitalized patients.

REFERENCES

[1] K. Singh et al., "Evaluating a widely implemented proprietary deterioration index model among hospitalized patients with COVID-19," *Ann. Amer. Thoracic Soc.*, vol. 18, no. 7, pp. 1129–1137, Jul. 2021.

[2] E. Mu, S. Jabbour, A. V. Dalca, J. Guttag, J. Wiens, and M. W. Sjoding, "Augmenting existing deterioration indices with chest radiographs to predict clinical deterioration," *PLoS ONE*, vol. 17, no. 2, Feb. 2022, Art. no. e0263922.

[3] C. Ross, "Hospitals are using ai to predict the decline of COVID-19 patients—Before knowing it works," STAT, Boston, MA, USA, Tech. Rep., 2020.

[4] E. H. Lee et al., "Deep COVID DeteCT: An international experience on COVID-19 lung detection and prognosis using chest CT," *NPJ Digit. Med.*, vol. 4, no. 1, p. 11, Jan. 2021.

[5] J. Theis, W. L. Galanter, A. D. Boyd, and H. Darabi, "Improving the in-hospital mortality prediction of diabetes ICU patients using a process mining/deep learning architecture," *IEEE J. Biomed. Health Informat.*, vol. 26, no. 1, pp. 388–399, Jan. 2022.

[6] H. Salehinejad et al., "Deep sequential learning for cervical spine fracture detection on computed tomography imaging," in *Proc. IEEE 18th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2021, pp. 1911–1914.

[7] H. Salehinejad and S. Valaee, "LiteHAR: Lightweight human activity recognition from WiFi signals with random convolution kernels," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2022, pp. 4068–4072.

[8] H. Salehinejad, E. Colak, T. Dowdell, J. Barfett, and S. Valaee, "Synthesizing chest X-ray pathology for training deep convolutional neural networks," *IEEE Trans. Med. Imag.*, vol. 38, no. 5, pp. 1197–1206, May 2019.

[9] D. Wang, A. Khosla, R. Gargeya, H. Irshad, and A. H. Beck, "Deep learning for identifying metastatic breast cancer," 2016, *arXiv:1606.05718*.

[10] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, pp. 539–546.

[11] N. Saunshi et al., "Understanding contrastive learning requires incorporating inductive biases," in *Proc. Int. Conf. Mach. Learn.*, 2022, pp. 19250–19286.

[12] C.-Y. Chuang, J. Robinson, Y.-C. Lin, A. Torralba, and S. Jegelka, "Debiased contrastive learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 8765–8775.

[13] N. Eldridge et al., "Trends in adverse event rates in hospitalized patients, 2010–2019," *JAMA*, vol. 328, no. 2, p. 173, Jul. 2022.

[14] J. Irvin et al., "CheXpert: A large chest radiograph dataset with uncertainty labels and expert comparison," in *Proc. AAAI Conf. Artif. Intell.*, 2019, vol. 33, no. 1, pp. 590–597.

[15] A. E. W. Johnson et al., "MIMIC-CXR, a de-identified publicly available database of chest radiographs with free-text reports," *Sci. Data*, vol. 6, no. 1, p. 317, Dec. 2019.

[16] H. Salehinejad, S. Valaee, T. Dowdell, and J. Barfett, "Image augmentation using radial transform for training deep neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 3016–3020.

[17] Y. Chen et al., "Contrastive learning for prediction of Alzheimer's disease using brain 18F-FDG PET," *IEEE J. Biomed. Health Informat.*, vol. 27, no. 4, pp. 1735–1746, Apr. 2023.

[18] K. Xu, J. Wang, L. Zhang, H. Zhu, and D. Zheng, "Dual-stream contrastive learning for channel state information based human activity recognition," *IEEE J. Biomed. Health Informat.*, vol. 27, no. 1, pp. 329–338, Jan. 2023.

[19] H. Salehinejad, N. Hasanzadeh, R. Djogo, and S. Valaee, "Contrastive representation of channel state information for human body orientation recognition in interaction with machines," in *Proc. IEEE 33rd Int. Workshop Mach. Learn. Signal Process. (MLSP)*, Sep. 2023, pp. 1–6.

[20] Z. Gao et al., "Unsupervised representation learning for tissue segmentation in histopathological images: From global to local contrast," *IEEE Trans. Med. Imag.*, vol. 41, no. 12, pp. 3611–3623, Dec. 2022.

[21] J. Shi, T. Gong, C. Wang, and C. Li, "Semi-supervised pixel contrastive learning framework for tissue segmentation in histopathological image," *IEEE J. Biomed. Health Informat.*, vol. 27, no. 1, pp. 97–108, Jan. 2023.

[22] Z. Zhu, L. Yu, W. Wu, R. Yu, D. Zhang, and L. Wang, "MuRCL: Multi-instance reinforcement contrastive learning for whole slide image classification," *IEEE Trans. Med. Imag.*, vol. 42, no. 5, pp. 1337–1348, May 2023.

[23] Y. Chen, C. Zhang, C. H. Q. Ding, and L. Liu, "Generating and weighting semantically consistent sample pairs for ultrasound contrastive learning," *IEEE Trans. Med. Imag.*, vol. 42, no. 5, pp. 1388–1400, May 2023.

[24] R. Liu, Z. Jiang, S. Yang, and X. Fan, "Twin adversarial contrastive learning for underwater image enhancement and beyond," *IEEE Trans. Image Process.*, vol. 31, pp. 4922–4936, 2022.

[25] Z. Liu, Z. Zhu, S. Zheng, Y. Liu, J. Zhou, and Y. Zhao, "Margin preserving self-paced contrastive learning towards domain adaptation for medical image segmentation," *IEEE J. Biomed. Health Informat.*, vol. 26, no. 2, pp. 638–647, Feb. 2022.

[26] X. He, L. Fang, M. Tan, and X. Chen, "Intra- and inter-slice contrastive learning for point supervised OCT fluid segmentation," *IEEE Trans. Image Process.*, vol. 31, pp. 1870–1881, 2022.

[27] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.

[28] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jul. 2006, pp. 1735–1742.

[29] Z. Yue et al., "Ts2Vec: Towards universal representation of time series," in *Proc. AAAI Conf. Artif. Intell.*, 2022, vol. 36, no. 8, pp. 8980–8987.

[30] Y. Tian, D. Krishnan, and P. Isola, "Contrastive multiview coding," in *Proc. 16th Eur. Conf. Comput. Vis.* Berlin, Germany: Springer-Verlag, Aug. 2020, pp. 776–794.

[31] A. Labach, H. Salehinejad, and S. Valaee, "Survey of dropout methods for deep neural networks," 2019, *arXiv:1904.13310*.

[32] D. Hendrycks and K. Gimpel, "Gaussian error linear units (GELUs)," 2016, *arXiv:1606.08415*.

[33] F. Karim, S. Majumdar, H. Darabi, and S. Chen, "LSTM fully convolutional networks for time series classification," *IEEE Access*, vol. 6, pp. 1662–1669, 2018.

[34] N. Elsayed, A. S. Maida, and M. Bayoumi, "Deep gated recurrent and convolutional network hybrid model for univariate time series classification," 2018, *arXiv:1812.07683*.

[35] H. I. Fawaz et al., "InceptionTime: Finding AlexNet for time series classification," *Data Mining Knowl. Discovery*, vol. 34, no. 6, pp. 1936–1962, Nov. 2020.

[36] E. Rahimian, S. Zabihi, S. F. Atashzar, A. Asif, and A. Mohammadi, "XceptionTime: A novel deep architecture based on depthwise separable convolutions for hand gesture classification," 2019, *arXiv:1911.03803*.

[37] A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.

[38] M. A. Hearst, S. T. Dumais, E. Osman, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intell. Syst. Appl.*, vol. 13, no. 4, pp. 18–28, Jul./Aug. 2008.

[39] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[40] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of online learning and an application to boosting," *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, Aug. 1997.

[41] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 785–794.

[42] A. E. Flanders et al., "Construction of a machine learning dataset through collaboration: The RSNA 2019 brain CT hemorrhage challenge," *Radiol., Artif. Intell.*, vol. 2, no. 4, Jul. 2020, Art. no. e209002.

[43] H. Salehinejad et al., "A real-world demonstration of machine learning generalizability in the detection of intracranial hemorrhage on head computerized tomography," *Sci. Rep.*, vol. 11, no. 1, p. 17051, Aug. 2021.

[44] H. A. Dau et al., "The UCR time series archive," *IEEE/CAA J. Autom. Sinica*, vol. 6, no. 6, pp. 1293–1305, Nov. 2019.

[45] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[46] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–11.