Cross-Domain Missingness-Aware Time-Series Adaptation With Similarity Distillation in Medical Applications

Baoyao Yang, Mang Ye, Qingxiong Tan, and Pong C. Yuen, *Senior Member, IEEE*

***Abstract*—Medical time series of laboratory tests has been collected in electronic health records (EHRs) in many countries. Machine-learning algorithms have been proposed to analyze the condition of patients using these medical records. However, medical time series may be recorded using different laboratory parameters in different datasets. This results in the failure of applying a pretrained model on a test dataset containing a time series of different laboratory parameters. This article proposes to solve this problem with an unsupervised time-series adaptation method that generates time series across laboratory parameters. Specifically, a medical time-series generation network with similarity distillation is developed to reduce the domain gap caused by the difference in laboratory parameters. The relations of different laboratory parameters are analyzed, and the similarity information is distilled to guide the generation of target-domain specific laboratory parameters. To further improve the performance in cross-domain medical applications, a missingness-aware feature extraction network is proposed, where the missingness patterns reflect the health conditions and, thus, serve as auxiliary features for medical analysis. In addition, we also introduce domain-adversarial networks in both feature level and time-series level to enhance the adaptation across domains. Experimental results show that the proposed method achieves good performance on both private and publicly available medical datasets. Ablation studies and distribution visualization are provided to further analyze the properties of the proposed method.**

***Index Terms*—Medical data, time series, unsupervised domain adaptation (UDA).**

I. Introduction

TO IMPROVE the quality and cost effectiveness of healthcare, electronic health record (EHR) systems [1] have been developed and widely adopted in many medical institutes. Recent studies [2] showed that mining the EHRs provided intrinsic information of medical data, which helps physicians in disease diagnosis. Therefore, many machine-learning algorithms [3], [4] have been proposed on EHR data toward addressing clinical questions. Among these EHRs, the results of laboratory tests are a common reference for the diagnosis of many diseases. With the popularity of deep learning algorithms [5]–[7], convolutional networks [8]–[10] were built using the results of multiple laboratory tests to estimate the status of patients and/or predict the risk of mortality. Although promising results have been achieved by these networks, the laboratory tests collected at a single time point are not informative enough for future disease/mortality prediction. This is because the results of laboratory tests would change over time, and the trend of laboratory tests reflects the conditions of patients.

In order to better analyze the patient's condition, a time series of laboratory tests [11] is collected and the advance analytics on medical time series are provided. Good performance has been presented in [12]–[14] by incorporating the temporal information extracted from multivariate medical time series. But these methods assumed that all laboratory parameters are the same in the training and test datasets. This assumption may not hold in practical scenarios because different medical institutes may be equipped with different testing devices, resulting in the use of different laboratory parameters in different datasets to measure the same substance in blood/urine. For example, the *Sysmex blood cell analyzer* records the amount of platelets in blood using "*PLT*." But this information is measured by "*PT*" in another dataset that is collected using the *ACL TOP LAS testing system*. Consequently, the problem of *laboratory parameter bias* occurs, where some laboratory parameters are common between the training and test datasets while others are different across datasets. Furthermore, the number of laboratory parameters may vary between the training and test datasets. Under these scenarios, the pretrained model is not applicable to a new dataset that contains different/unseen laboratory parameters.

To tackle this challenge, one may suggest a training recognition model for time series of each laboratory parameter. However, the time series of some test laboratory parameters is not available in the training dataset and, thus, the models for these inconsistent laboratory parameters are unobtainable. A more elegant approach is unsupervised domain adaptation (UDA) [15] that solves the problem of domain shift. UDA has been widely studied in computer vision [16], [17] and

Manuscript received November 8, 2019; revised May 5, 2020; accepted July 19, 2020. Date of publication August 14, 2020; date of current version May 19, 2022. This work was supported in part by the Science Faculty Research Grant of Hong Kong Baptist University, Hong Kong Research Grants Council General Research Fund under Grant RGC/HKBU12200518; and in part by the Health and Medical Research Fund Project under Grant 07180216. This article was recommended by Associate Editor M. Han. *(Corresponding author: Pong C. Yuen.)*

The authors are with the Department of Computer Science, Hong Kong Baptist University, Hong Kong (e-mail: byyang@comp.hkbu.edu.hk; mangye@comp.hkbu.edu.hk; csqxtan@comp.hkbu.edu.hk; pcyuen@comp.hkbu.edu.hk).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCYB.2020.3011934.

Digital Object Identifier 10.1109/TCYB.2020.3011934

2168-2267 © 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: University of Southern California. Downloaded on October 11,2023 at 02:46:45 UTC from IEEE Xplore. Restrictions apply.

natural language processing [18], [19], where they mainly focus on aligning cross-domain data distributions within the feature space. For time-series data, Purushotham *et al.* [20] and Yan *et al.* [21] also incorporated the temporal dependencies for domain-invariant feature learning. However, the existing UDA methods cannot handle the problem of laboratory parameter bias, where the domain differences are mainly caused by the difference in laboratory parameters. As shown in Fig. 1, the time series of different laboratory parameters is present in highly different patterns. As a consequence, the domain gap between samples with different laboratory parameters cannot be modeled as a distribution shift between the source and target domains. Therefore, this domain gap could hardly be reduced by the existing UDA methods that align data distributions across domains. In addition, medical time series usually suffer from the missing data problem [22] caused by the incomplete data collection and the irregularities in medical treatment. It increases additional difficulties for the adaptation across different laboratory parameters.

To address the aforementioned issues, this article presents a novel missingness-aware time-series adaptation method with similarity distillation in medical applications. Specifically, we assume that the labeled training medical time-series dataset is the source domain and the unlabeled test dataset is the target domain. A time-series generation network with similarity distillation is developed to model the transformation between the time series of different laboratory parameters across the source and target domains. The domain gap between different laboratory parameters is reduced by generating a time series of target-domain specific laboratory parameters. With the assumption that samples with similar time series in one laboratory parameter have a high probability of containing similar time series in another laboratory parameter, these similarities are distilled to reflect the relations of different laboratory parameters. The distilled similarities are ranked and used as a guidance in the medical time-series generation network. In particular, a similarity ranking criterion is designed to maintain the consistency of similarity rankings between time series of different laboratory parameters. In this manner, the reliability of the generated target time series is improved since the relations of different laboratory parameters are preserved during the generation.

In addition, to handle the missingness problem in medical time series, we introduce a missingness-aware feature extraction module with co-occurrence convolution kernels to learn the discriminate features by integrating the extracted missingness patterns. Rather than impute the missing data [23], our proposed strategy is beneficial for medical applications since the missingness patterns reflect the health conditions of patients. In addition, the generation framework is constrained by a well-designed domain-adversarial network in both time-series level and feature level to enhance the adaptation across domains. The generated target time series are restricted to be indistinguishable from the real target time series by the domain-adversarial networks in both feature and time-series levels.

The contributions can be summarized as follows.

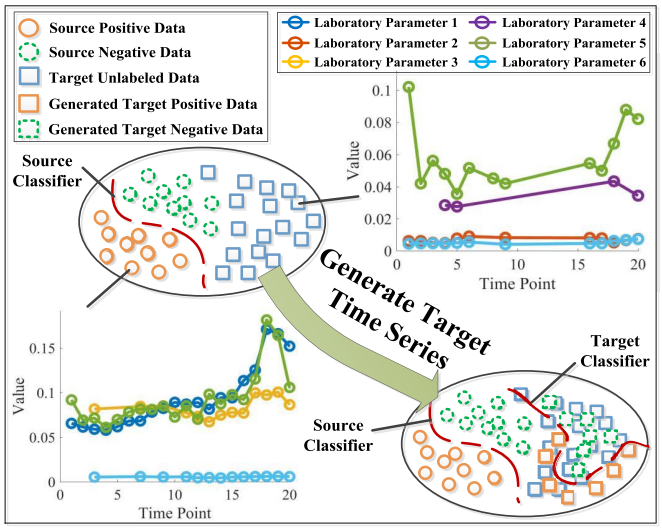

Fig. 1. Illustration of cross-domain time-series adaptation. Both source and target time series contain laboratory parameters 3–6. The laboratory parameter 1 is only recorded in the source domain while laboratory parameter 2 is only included in the target samples. We generate the time series of target laboratory parameters from the source time series in order to train a classifier in the target domain.

1) We propose a medical time-series adaptation method that generates target time series to solve the problem of laboratory parameter bias across medical datasets.
2) We develop a cross-domain medical time-series generation network with similarity distillation for domain adaptation across different laboratory parameters. This network is an unsupervised sequence-to-sequence generation network that does not require corresponding time-series pairs in the source and target domains. The similarities of time series are distilled to maintain the relations between different laboratory parameters during the generation of medical time series.
3) We propose a missingness-aware feature extraction network to extract missingness patterns from medical time series as auxiliary features for cross-domain medical applications.
4) We introduce domain-adversarial networks to guide the generation model learning, which captures the information in both feature level and time-series level to enhance the adaptation across domains.

The remainder of this article is organized as follows. In Section II, we give a review of the existing data analysis algorithms for time series in medical applications. Section II also reviews the existing domain adaptation methods. The proposed cross-domain medical time-series adaptation method is then introduced and discussed in Section III. The experimental results will be presented and analyzed in Section IV, followed by a conclusion given in Section V.

II. Related Work

In this section, we first review the existing data analysis algorithms on medical time series. An introduction of the existing domain adaptation methods is provided afterward. We will

Authorized licensed use limited to: University of Southern California. Downloaded on October 11,2023 at 02:46:45 UTC from IEEE Xplore. Restrictions apply.

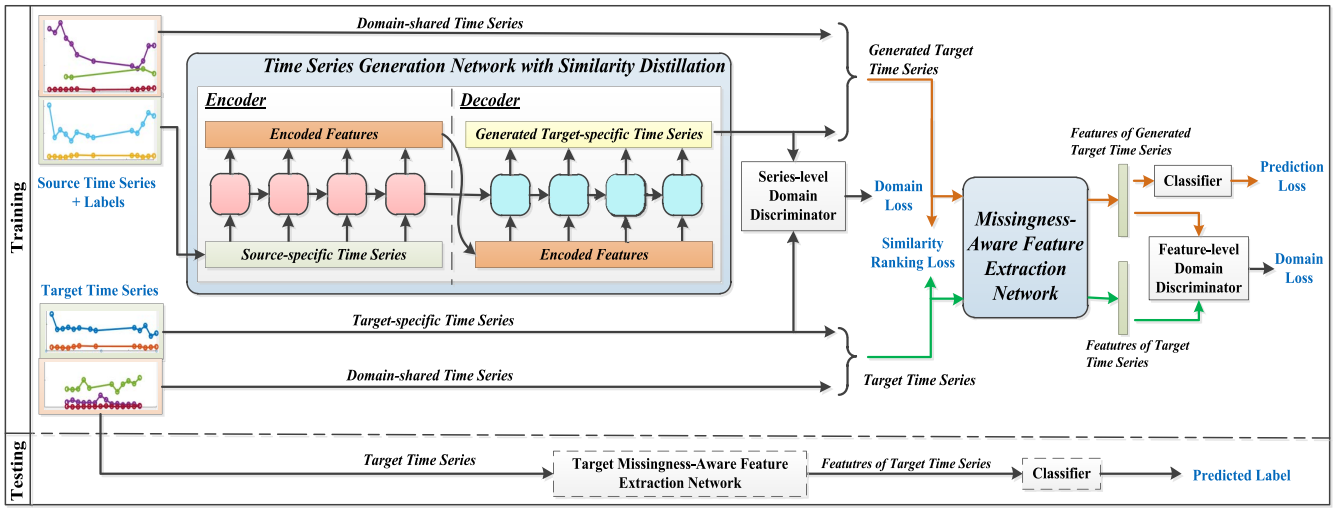

Fig. 2. Overview of the proposed method. In the training phase, a time-series generation network with similarity distillation is trained to generate target time series. The generated target time series are then used to learn a missingness-aware feature extraction network to extract discriminative features for cross-domain disease/mortality prediction. In the testing phase, features are extracted from target time series using the trained missingness-aware feature extraction network to predict labels.

introduce the existing domain adaptation methods for medical time series at the end of this section.

A. Advanced Data Analytics on Medical Time Series

Medical time series of laboratory tests are effective information in the EHRs. Thus, data mining algorithms [24]–[26] are proposed to find discriminative patterns from medical times series in order to help clinical diagnoses. Disease recognition/prediction can then be achieved by searching these discriminative patterns from the medical time series. Moreover, statistical attributes [27] are extracted, and features [28] are designed to represent discriminable information in medical time series. Machine-learning models (e.g., k-nearest neighbor, support vector machine, etc.) are then applied to the extracted features for disease analysis. In addition to handcrafted features, deep-learning networks [29], [30] have been built on multivariate medical time series toward solving medical problems in recent years. The deep-learning networks automatically learn features from medical time series, which are more discriminative for a specific recognition task. As medical time series are sequence data, most of these networks are developed based on the long short-term memory (LSTM) [31] network to learn and extract discriminative features with temporal dependencies. But the frequency of medical treatment varies from patients to patients, which leads to the problem of irregularity in medical time series. To handle the irregularity issue, data preprocessing algorithms are proposed to align the irregular medical time series by data imputation [23] or time wrapping [32]. The performance of models learned by the aligned data would be improved. Besides, Che *et al.* [4] argued that the data irregularity in medical time series reflects patients' health condition. Thus, they exploit the missingness information of medical time series to impute the missing data for mortality prediction.

Although promising results have been reported in recent researches, these analysis algorithms are designed for the cases where the same laboratory parameters are recorded in the training and test time series. Without the consideration of dataset bias, the existing algorithms are inapplicable in practical applications where the laboratory parameters are inconsistent in the training and test datasets.

B. Domain Adaptation Methods

Dataset bias between the training and test datasets is a common phenomenon in a variety of machine-learning applications. Therefore, many domain adaptation algorithms [16], [17], [33], [34] have been developed to increase the generalization ability of recognition models in the test datasets. For semisupervised domain adaptation, the labels of a few target data are given. Daumé *et al.* [35] obtained the adaptation by concatenating the labeled source and target data. Other semisupervised domain adaptation methods [36], [37] propose to refine the source model to the target domain using the target labeled data.

However, label information may be unavailable in the test datasets in practical applications. UDA [15] was proposed to solve the problem of domain shift without target labels. Gong *et al.* [38] and Aljundi *et al.* [39] estimated label information as target-domain landmarks to learn the model in the target domain. But the performance of these methods highly dependent on the quality of label estimation. Distribution alignment is another approach that does not estimate target labels. Source and target data are projected into subspaces. The distribution alignment is achieved by aligning the geometric bases of subspaces [40] and/or aligning data representations in the subspaces [41]. The domain-invariant features could also be obtained using the deep-learning networks [42], [43] with distribution alignment criteria [44]. Recently, techniques of adversarial learning [45] have been introduced. Domain-adversarial networks [46], [47] have been developed to learn the domain-invariant features.

Authorized licensed use limited to: University of Southern California. Downloaded on October 11,2023 at 02:46:45 UTC from IEEE Xplore. Restrictions apply.

Saito *et al.* [48] and Lee *et al.* [49] then improved the discrimination in the domain-invariant space by introducing multiple classifiers. Furthermore, Sankaranarayanan *et al.* [50] achieved domain adaptation by generating data across domains. Along this direction, several generative networks [51], [52] are developed. The generated target data are regarded as target label information so that the target model can be trained using the generated target data.

Although domain adaptation techniques have achieved rapid development in recent years, most of them have been developed for the applications of computer vision and natural language processing. These methods may not be suitable for direct application in time series in medical applications because they ignore the specific patterns that reflect the health status of patients in medical time series.

C. Domain Adaptation for Medical Time Series

Some researchers realized that the problem of dataset bias also exists in time series in medical applications and, thus, cross-domain medical time-series adaptation algorithms have been developed. Alves *et al.* [53] found that medical time series could vary by age, conditions, and interventions of patients. They develop multiple models specific to different populations and learn the transferable features across different population groups. On the other hand, Gupta *et al.* [54] achieved the domain adaption by refining the source model using some target labeled medical time series. For unsupervised cases where target labels are unavailable, the domain adaptation is obtained by learning domain-invariant features across domains. Purushotham *et al.* [20] extended the domain-adversarial networks [46] using variational recurrent neural networks [55] in order to capture the temporal information in medical time series. Instead, da Costa *et al.* [56] learned the domain-invariant features with temporal information by LSTM networks [57]. Besides, Yan *et al.* [21] tried to extract domain features that are specific to each domain. The domain-invariant subspace is achieved by maximizing the domain independence among the domain features.

The above methods mainly study the extraction of temporal information from time series in domain adaptation, which may not be helpful to solve the problem of changing laboratory parameters, where time series are presented in highly different patterns across domains. In addition, these methods are infeasible for heterogeneous cases, where the numbers of laboratory parameters are various in the source and target domains. Therefore, it is necessary to develop a novel time-series adaptation method to address the problem of laboratory parameter bias in medical applications.

III. Proposed Method

This section presents the proposed cross-domain medical time-series adaptation method. As shown in Fig. 2, the multivariate medical time series in the source and target domains is divided into domain-shared and domain-specific time series according to the types of laboratory parameters. The source- and target-domain-shared time series have the same laboratory parameters, and the source-specific and target-specific time series are different in the laboratory parameter.

In the training phase, a time-series generation network with similarity distillation is trained to transform the source-specific time series to the target-specific time series using both source and target time series. The generated target-specific time series, together with the domain-shared time series in the source domain and the time series in the target domain are then passed to a missingness-aware feature extraction network to learn the features for disease/mortality prediction. Two domain discriminators *(time-series-level and feature-level domain discriminators)* followed by domain losses are introduced to reduce the domain gap between the source and target domains. We also design a similarity ranking loss to guide the generation of target-specific time series. To preserve the discrimination in the feature space, the extracted source features are constrained by a classifier with a source-domain prediction loss.

In the testing phase, features are extracted from the target-domain medical time series using the trained missingness-aware feature extraction network. The trained classifier is then employed to predict labels for the extracted target features. The networks of time-series generation and missingness-aware feature extraction will be introduced in Sections III-A and III-B, respectively. The main symbols to be used in this article are summarized in Table I.

TABLE I
List of Symbols to Be Used in This Article

$\boldsymbol{X}_s$	Source time series	$\boldsymbol{X}_t$	Target time series
$\boldsymbol{y}_s$	Source labels	C	Classifier
M_s^{cnn}	Source CNN	M_t^{cnn}	Target CNN
M_s^{lstm}	Source LSTM	M_t^{lstm}	Target LSTM
$\boldsymbol{X}_s^{co}, \boldsymbol{X}_t^{co}$	Domain-shared time series in the source/target domain		
$\boldsymbol{X}_s^{sp}, \boldsymbol{X}_t^{sp}$	Domain-specific time series in the source/target domain		
$\hat{\boldsymbol{X}}_t^{sp}$	Generated target-specific time series		
$\boldsymbol{X}_s^{mp}, \boldsymbol{X}_t^{mp}$	Missing patterns of source/target time series		
M_{Seq}	Time series generator		
D_{Seq}	Series-level domain discriminator		
D_{fe}	Feature-level domain discriminator		
i, j, k	Index of source/target samples		
p_i, p_j	Index of laboratory parameters		
n_p	Number of domain-shared laboratory parameters		
n_s^{sp}, n_t^{sp}	Number of source-specific/target-specific laboratory parameters		
l_t	Length of time in the source and target time series		
n_s, n_t	Total number of source/target samples		

A. Time-Series Generation Network With Similarity Distillation

Denote the source and target time series as $\boldsymbol{X}_s$ and $\boldsymbol{X}_t$, respectively. For each domain $d \in \{s, t\}$, time series $\boldsymbol{X}_d$ are divided into domain-shared and domain-specific time series according to the types of laboratory parameters, that is, $\boldsymbol{X}_d = \{\boldsymbol{X}_d^{co}; \boldsymbol{X}_d^{sp}\}$.[1] The domain-shared time series $\boldsymbol{X}_s^{co} \in \mathbb{R}^{n_p \times l_t \times n_s}$ and $\boldsymbol{X}_t^{co} \in \mathbb{R}^{n_p \times l_t \times n_t}$ have the same laboratory parameters. The notation n_p is the number of laboratory parameters shared across domains, l_t represents the time length of medical time series, and n_s and n_t are the numbers of samples in the source and target domains, respectively. The source-specific

[1]The symbol of {;} in this article denotes the concatenation of two tensors in the first dimension.

Authorized licensed use limited to: University of Southern California. Downloaded on October 11,2023 at 02:46:45 UTC from IEEE Xplore. Restrictions apply.

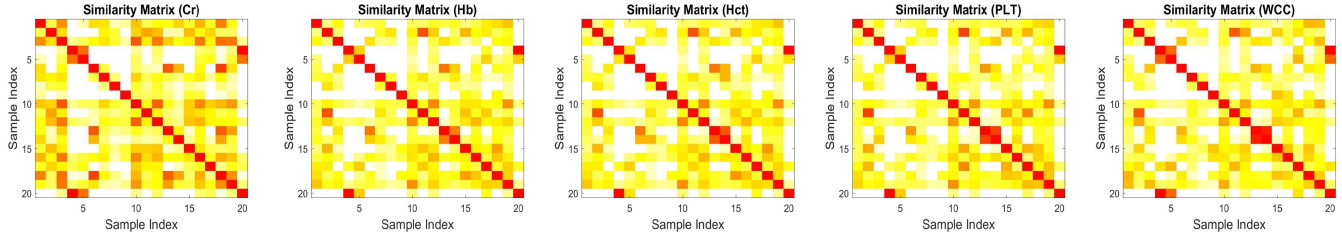

Fig. 3. Similarity matrices of different laboratory parameters. Twenty samples are randomly selected from the PWH dataset for illustration. The elements in these matrices show the similarities between the selected samples. Each subfigure shows the sample similarity for one type of laboratory parameter. The darker color represents higher similarity. The sample similarities are consistence among different laboratory parameters.

time series $\boldsymbol{X}_s^{sp} \in \mathbb{R}^{n_s^{sp} \times l_t \times n_s}$ and target-specific time series $\boldsymbol{X}_t^{sp} \in \mathbb{R}^{n_t^{sp} \times l_t \times n_s}$ vary in the laboratory parameter. The numbers of domain-specific laboratory parameters (n_s^{sp} and n_t^{sp}) can be different in the source and target domains. As mentioned in Section I, domain-specific time series $\boldsymbol{X}_s^{sp}$ and $\boldsymbol{X}_t^{sp}$ are present in highly different patterns and, thus, aligning data distributions between $\boldsymbol{X}_s^{sp}$ and $\boldsymbol{X}_t^{sp}$ could not nicely reduce the domain gap. Motivated by the sequence-to-sequence generation [58], this article proposes generating target-specific time series $\hat{\boldsymbol{X}}_t^{sp}$ from the source-specific time series $\boldsymbol{X}_s^{sp}$. We concatenate $\hat{\boldsymbol{X}}_t^{sp}$ with the source-domain-shared times series $\boldsymbol{X}_s^{co}$ to obtain the generated target time series $\hat{\boldsymbol{X}}_t = \{\boldsymbol{X}_s^{co}; \hat{\boldsymbol{X}}_t^{sp}\}$ as labeled data in the target domain. The generated target time series $\hat{\boldsymbol{X}}_t$ can then be used to learn the feature extraction network and train a classifier for the real target time series $\boldsymbol{X}_t$.

1) Network Architecture: As illustrated in Fig. 2, the proposed time-series generation network is developed based on the encoder–decoder framework. The source-specific time series $\boldsymbol{X}_s^{sp}$ are first inputted into an encoder with multiple-layer LSTMCells [58]. The encoded features are the concatenation of the hidden states of LSTMCells along the time axis l_t. We denote the encoded features as $\boldsymbol{Z}_s \in \mathbb{R}^{m_s \times l_t \times n_s}$, where m_s is the number of units in the LSTMCells. The encoded features and the output statues of the encoder are then passed to a decoder with multiple-layer LSTMCells. m_t units are contained in the LSTMCells of the decoder. The outputs of hidden states $\boldsymbol{Z}_t \in \mathbb{R}^{m_t \times l_t \times n_s}$ are densely connected to generate target-specific time series $\hat{\boldsymbol{X}}_t^{sp}$. The dimension of each generated target-specific time series is $n_t^{sp} \times l_t$, which is the same as the dimension of the real target-specific time series.

2) Training Constraints: Write the parameters of the time-series generator as M_{Seq}. Without corresponding pairs of the source and target time series, the generator M_{Seq} cannot be end-to-end learned as the existing sequence-to-sequence generation networks [58]. Inspired by the domain-adversarial learning [46], we relax the generation criterion to generate time series indistinguishable from the real target-specific time series. This criterion is applied in both *time-series level* and *feature level*. In other words, the generated target time series should not only have laboratory results similar to the real target data but should also be located close to the real target samples in the feature space.

In the *time-series level*, a domain discriminator D_{Seq} is introduced. The above-mentioned criterion is achieved if D_{Seq} could not correctly classify the generated series $\hat{\boldsymbol{X}}_t^{sp}$ from the real target series $\boldsymbol{X}_t^{sp}$. Mathematically, the time-series generator M_{Seq} is updated by

$$\begin{aligned}\min_{M_{\text{Seq}}} \; & L_{M_{\text{Seq}}}\left(\boldsymbol{X}_s^{sp}, \boldsymbol{X}_t^{sp}, M_{\text{Seq}}, D_{\text{Seq}}\right) \\ & = -\mathbb{E}_{\boldsymbol{X}_s^{sp}}\left[\log\left(1 - D_{\text{Seq}}\left(M_{\text{Seq}}\left(\boldsymbol{X}_s^{sp}\right)\right)\right)\right] \\ & \quad - \mathbb{E}_{\boldsymbol{X}_t^{sp}}\left[\log\left(D_{\text{Seq}}\left(\boldsymbol{X}_t^{sp}\right)\right)\right]. \end{aligned} \tag{1}$$

Similarly, the generation criterion in the *feature level* is obtained using a feature-level domain discriminator D_{fe}. As shown in Fig. 2, features of the generated and the real target time series are extracted by the missingness-aware feature extraction network (details will be introduced in Section III-B). Denote the parameters of feature extractors for the generated and the real target time series as M_s and M_t, respectively. The features $M_s(\hat{\boldsymbol{X}}_t)$ and $M_t(\boldsymbol{X}_t)$ are restricted to be domain indistinguishable by the domain discriminator D_{fe}. We optimize the generator (M_{Seq}) and the feature extractors (M_s and M_t) by

$$\begin{aligned}\min_{M_{\text{Seq}}, M_s, M_t} \; & L_M\left(\boldsymbol{X}_s, \boldsymbol{X}_t, M_{\text{Seq}}, M_s, M_t, D_{fe}\right) \\ & = -\mathbb{E}_{\boldsymbol{X}_s}\left[\log\left(1 - D_{fe}\left(M_s\left(\left\{\boldsymbol{X}_s^{co}; M_{\text{Seq}}\left(\boldsymbol{X}_s^{sp}\right)\right\}\right)\right)\right)\right] \\ & \quad - \mathbb{E}_{\boldsymbol{X}_t}\left[\log\left(D_{fe}(M_t(\boldsymbol{X}_t))\right)\right]. \end{aligned} \tag{2}$$

The time-series-level and feature-level domain discriminators (D_{Seq} and D_{fe}) are two adversarial networks. Hence, the discrimination ability of D_{Seq} and D_{fe} should be simultaneously improved while updating the time-series generator M_{Seq} and feature extractors M_s and M_t. The optimizations are written as

$$\begin{aligned}\min_{D_{\text{Seq}}} \; & L_{D_{\text{Seq}}}\left(\boldsymbol{X}_s^{sp}, \boldsymbol{X}_t^{sp}, M_{\text{Seq}}, D_{\text{Seq}}\right) \\ & = -\mathbb{E}_{\boldsymbol{X}_s^{sp}}\left[\log\left(D_{\text{Seq}}\left(M_{\text{Seq}}\left(\boldsymbol{X}_s^{sp}\right)\right)\right)\right] \\ & \quad - \mathbb{E}_{\boldsymbol{X}_t^{sp}}\left[\log\left(1 - D_{\text{Seq}}\left(\boldsymbol{X}_t^{sp}\right)\right)\right] \end{aligned} \tag{3}$$

$$\begin{aligned}\min_{D_{fe}} \; & L_D\left(\boldsymbol{X}_s, \boldsymbol{X}_t, M_{\text{Seq}}, M_s, M_t, D_{fe}\right) \\ & = -\mathbb{E}_{\boldsymbol{X}_s}\left[\log\left(D_{fe}\left(M_s\left(\left\{\boldsymbol{X}_s^{co}; M_{\text{Seq}}\left(\boldsymbol{X}_s^{sp}\right)\right\}\right)\right)\right)\right] \\ & \quad - \mathbb{E}_{\boldsymbol{X}_t}\left[\log\left(1 - D_{fe}(M_t(\boldsymbol{X}_t))\right)\right]. \end{aligned} \tag{4}$$

Constrained by (1)–(4), each source time series will be mapped to the vicinity of several target samples in the feature space, as illustrated in Fig. 1. Thus, the domain gap between the source and target domains is reduced. The source-domain label information can be propagated to the target domain via the generated target time series.

Authorized licensed use limited to: University of Southern California. Downloaded on October 11,2023 at 02:46:45 UTC from IEEE Xplore. Restrictions apply.

However, the constraints of (1) and (2) remain too weak to generate reliable time series in the target domain. Each generated time series can be aligned with arbitrary target samples. Consequently, a positive source time series may be mapped to the neighborhoods of negative target samples in the feature space, and vice versa. In order to improve the reliability of time-series generation, we propose utilizing knowledge from a medical perspective to guide the generation of target-specific time series. According to medical knowledge, we know that different substances in the blood/urine interact with each other, and thus the results of different laboratory parameters are intrinsically related. If two samples include similar time series of a certain laboratory parameter, these two samples are likely to have similar time series of other laboratory parameters. To further indicate this point, we calculate the sample similarity for each laboratory parameter and display the similarity matrices of different laboratory parameters in Fig. 3. It is shown in Fig. 3 that the sample similarities are consistent among different laboratory parameters.

Based on the above observation, this article distills the similarity information from medical time series to reflect the relations of different laboratory parameters. We aim to preserve these relations during the generation to enhance the reliability of the generated target-specific time series. Thus, the distilled similarities are constrained to be consistent between the domain-shared and domain-specific laboratory parameters. To achieve this goal, a similarity ranking loss is designed and applied as a constraint in the generation network. In particular, we rank the sample similarities between the domain-shared time series in the source and target domains. The similarities between the generated and the real target-specific time series are restricted to follow the same similarity ranking.

Mathematically, the similarities between source- and target-domain-shared time series are computed and recorded in a similarity matrix $\boldsymbol{B}^{co} \in \mathbb{R}^{n_s \times n_t}$. Each element $\boldsymbol{B}^{co}_{i,j}$ represents the similarity between the ith source-domain-shared time series $\boldsymbol{X}^{co}_{s,i}$ and the jth target-domain-shared time series $\boldsymbol{X}^{co}_{t,j}$. Similarly, the similarities between the generated and real target-specific time series are calculated and recorded in the similarity matrix $\boldsymbol{B}^{sp} \in \mathbb{R}^{n_s \times n_t}$. The ith row of $\boldsymbol{B}^{sp}_{i,:}$ records the similarities between the ith generated target-specific time series $\hat{\boldsymbol{X}}^{sp}_{t,i}$ and each real target-specific time series. The similarity ranking loss constrains that the elements in $\boldsymbol{B}^{sp}_{i,:}$ are ranked based on $\boldsymbol{B}^{co}_{i,:}$, that is, the similarities between $\boldsymbol{X}^{co}_{s,i}$ and each target-domain-shared time series. In other words, $\boldsymbol{B}^{sp}_{i,j} > \boldsymbol{B}^{sp}_{i,k}$, if $\boldsymbol{B}^{co}_{i,j} > \boldsymbol{B}^{co}_{i,k}$, where $j \neq k$, and j, k are the indices of target time series. The optimization function with similarity ranking loss can be written as

$$\min_{M_{\text{Seq}}} \; L_M\left(\boldsymbol{X}^{co}_s, \boldsymbol{X}^{co}_t, \boldsymbol{X}^{sp}_s, \boldsymbol{X}^{sp}_t, M_{\text{Seq}}\right) = \sum_{p,q}^{n_s n_t} \left\| \boldsymbol{B}^{co}_{(p)} - \boldsymbol{B}^{co}_{(q)} \right\|_2^2 - \left\| \boldsymbol{B}^{sp}_{(p)} - \boldsymbol{B}^{sp}_{(q)} \right\|_2^2 \tag{5}$$

where p and q represent the indices of elements in $\boldsymbol{B}^{co}$ and $\boldsymbol{B}^{sp}$, and n_s and n_t are the number of source and target samples, respectively.

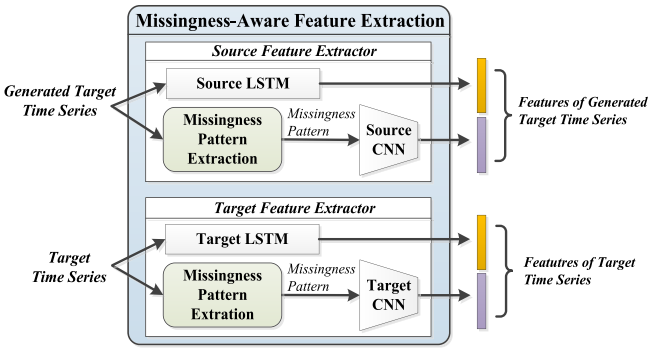

Fig. 4. Illustration of the missingnsess-aware feature extraction network.

As mentioned in Section I, medical time series contain missing data. To handle the missing data problem, an omission mechanism is employed in the calculation of the similarity matrices $\boldsymbol{B}^{co}$ and $\boldsymbol{B}^{sp}$. For two time-series samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, the similarity between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ is the summation of the similarities of each corresponding element in $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, that is, $\boldsymbol{b}_{i,j} = \sum_p \text{sim}(\boldsymbol{x}_{i,(p)}, \boldsymbol{x}_{j,(p)})$, where p is the index of each element. We omit the similarity of $\text{sim}(\boldsymbol{x}_{i,(p)}, \boldsymbol{x}_{j,(p)})$ if the values of $\boldsymbol{x}_{i,(p)}$ and/or $\boldsymbol{x}_{j,(p)}$ are missing. As the values of laboratory tests are commonly positive, the similarity $\boldsymbol{b}_{i,j}$ increases with the increase of numbers of elements that are valid in both $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. Cosine similarity is applied, and the omission mechanism is obtained by setting the value of missing data as 0. We obtain $\boldsymbol{B}^{co}_{i,j} = ([\boldsymbol{X}^{co}_{s,i}(\boldsymbol{X}^{co}_{t,j})']/[\|\boldsymbol{X}^{co}_{s,i}\|\|(\boldsymbol{X}^{co}_{t,j})\| + \epsilon])$ and $\boldsymbol{B}^{sp}_{i,j} = ([\hat{\boldsymbol{X}}^{sp}_{t,i}(\boldsymbol{X}^{sp}_{t,j})']/[\|\hat{\boldsymbol{X}}^{sp}_{t,i}\|\|(\boldsymbol{X}^{sp}_{t,j})\| + \epsilon])$, where ϵ is a small constant to avoid computational errors.

With (5), the distilled similarities are consistent between time series of domain-shared and target-specific laboratory parameters. Thus, the relations between different laboratory parameters are preserved during the time-series generation. The generated target-specific time series are more reliable with the guidance from the domain-shared laboratory parameters.

In addition, we preserve the model discrimination in the feature space. This is obtained by introducing a classifier C and minimizing the source-domain prediction loss

$$\min_{M_s, M_{\text{Seq}}, C} \; L_C\left(\boldsymbol{X}_s, M_s, M_{\text{Seq}}, C\right) = -\mathbb{E}_{(\boldsymbol{X}_s, \boldsymbol{y}_s)} \times \left[\sum_k \mathbb{1}_{[\boldsymbol{y}_s = k]} \log C\left(M_s\left(\left\{\boldsymbol{X}^{co}_s; M_{\text{Seq}}\left(\boldsymbol{X}^{sp}_s\right)\right\}\right)\right) \right]. \tag{6}$$

With the above optimizations, we learn the time-series generation network with similarity distillation and obtain the generated target-specific time series $\hat{\boldsymbol{X}}^{sp}_t = M_{\text{Seq}}(\boldsymbol{X}^{sp}_s)$. The generated target time series can be written as $\hat{\boldsymbol{X}}_t = \{\boldsymbol{X}^{co}_s; \hat{\boldsymbol{X}}^{sp}_t\}$. As $\hat{\boldsymbol{X}}^{sp}_t$ is similar to $\boldsymbol{X}^{sp}_t$, the domain gap caused by the difference of laboratory parameters is reduced.

B. Missingness-Aware Feature Extraction Network

In medical time series, missingness [4] reflects the frequency with which the patient sees a doctor. Thus, missingness is an important information in health analysis.

Authorized licensed use limited to: University of Southern California. Downloaded on October 11,2023 at 02:46:45 UTC from IEEE Xplore. Restrictions apply.

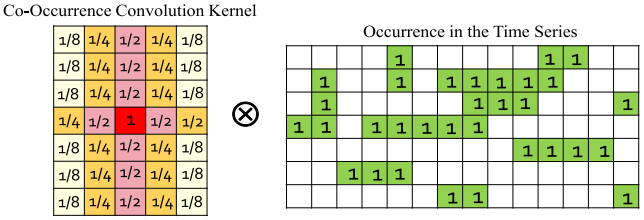

Fig. 5. Example of co-occurrence convolution kernel.

Unlike [4] that used missingness information as the reliability measure for time-series imputation and model update, this article proposes extracting patterns of missingness for disease/mortality prediction. The missingness information can be better preserved in the extracted features and directly contribute to the final decision. The network of missingnsess-aware feature extraction is illustrated in Fig. 4. In this network, features of the generated and the real target time series are extracted by two separated feature extractors, as domain shift may exist between the source- and target-domain-shared time series ($\boldsymbol{X}_s^{co}$ and $\boldsymbol{X}_t^{co}$). In each feature extractor, an LSTM network is contained to extract features with temporal information. At the same time, missingness patterns are extracted and a convolutional neural network (CNN) is employed to extract features from the missingness patterns for label prediction. However, the missingness information in the generated target-specific time series $\hat{\boldsymbol{X}}_t^{sp}$ would be broken during the time-series generation. Therefore, we only focus on extracting the missingness patterns from the domain-shared time series ($\boldsymbol{X}_s^{co}$ and $\boldsymbol{X}_t^{co}$).

In each domain d, the missingness patterns of $\boldsymbol{X}_d^{co}$ should reflect: 1) data occurrence and 2) data co-occurrence among neighborhoods. To represent the data occurrence, a mask of missing data $\boldsymbol{X}_d^{ms}$ is extracted from $\boldsymbol{X}_d^{co}$. For each element in $\boldsymbol{X}_d^{co}$, $\boldsymbol{X}_{d,(p)}^{ms} = 0$ if the value of $\boldsymbol{X}_{d,(p)}^{co}$ is missing; otherwise, $\boldsymbol{X}_{d,(p)}^{ms} = 1$. The letter p represents the index of elements in $\boldsymbol{X}_d^{co}$. For data co-occurrence, n_p co-occurrence convolution kernels are designed to represent the correlation of data occurrence, where n_p is the number of domain-shared laboratory parameters. These co-occurrence convolution kernels are applied on $\boldsymbol{X}_d^{ms}$ to extract the missingness patterns $\boldsymbol{X}_d^{mp}$.

Denote the p_ith co-occurrence convolution kernel as $\mathcal{K}^{p_i} \in \mathbb{R}^{n_p \times l_n}$, where $l_n = 2k_t + 1$ is the length of the neighborhoods with k_t-nearest neighbors along the time axis. We first consider the correlation of data occurrence along the time axis. The center of $\mathcal{K}^{p_i}$ is set as 1, indicating the occurrence of the data itself. The occurrence of the k_i-nearest neighbor is less relevant when k_i becomes larger. Thus, we set $\mathcal{K}^{p_i}_{p_i,k_c \pm k_i} = 2^{-(k_i)}$, where $k_c = k_t + 1$ is the index of the center. Besides, the co-occurrence of different laboratory parameters should also be considered. Since the correlation between different laboratory parameters is lower than that of the same laboratory parameters, we define $\mathcal{K}^{p_i}_{p_j,:} = (1/2)\mathcal{K}^{p_i}_{p_i,:}$, for $p_j \neq p_i$. An example of co-occurrence convolution kernel is illustrated in Fig. 5. Take the sum of n_p co-occurrence convolution kernels, the missingness pattern is expressed as $\boldsymbol{X}_d^{mp} = \sum_{p_i}^{n_p} \mathcal{K}^{p_i} \otimes \boldsymbol{X}_d^{ms}$. A CNN network is then employed on $\boldsymbol{X}_d^{mp}$ to extract features of missingness patterns for label prediction.

Combining the networks of LSTM and CNN, the feature extractors in the source and target domains can be represented as $M_s = \{M_s^{\text{lstm}}, M_s^{cnn}\}$ and $M_t = \{M_t^{\text{lstm}}, M_t^{cnn}\}$, respectively. The extracted features of the generated target time series in (2), (4), and (6) can be updated as $M_s(\hat{\boldsymbol{X}}_t) = [M_s^{\text{lstm}}(\hat{\boldsymbol{X}}_t); M_s^{cnn}(\boldsymbol{X}_s^{mp})]$, where $\hat{\boldsymbol{X}}_t = \{\boldsymbol{X}_s^{co}; M_{\text{Seq}}(\boldsymbol{X}_s^{sp})\}$. Similarly, the extracted features of the real target time series are represented as $M_t(\boldsymbol{X}_t) = [M_t^{\text{lstm}}(\boldsymbol{X}_t); M_t^{cnn}(\boldsymbol{X}_t^{mp})]$.

IV. Experiments

The proposed method is evaluated by the task of mortality prediction on two medical datasets, namely, the PWH dataset and the MICMIC-III [11] dataset. Results are reported and analyzed in Section IV-C. We also conduct experiments of heterogeneous medical time-series adaptation, in which the time series of additional unseen laboratory parameters are given in the target domain. Results of the heterogeneous domain adaptation will be discussed in Section IV-D. In order to analyze the stability of the proposed method among different sample populations, statistical experiments are performed and reported in Section IV-E. Ablation studies are given in Section IV-F to show the effectiveness of each component in the proposed method. In Section IV-G, we visualize the feature distribution of the generated target time series to intuitively display the performance of the time-series generation network.

A. Dataset

The *PWH dataset* includes the laboratory tests of 6237 patients collected from the Prince of Wales Hospital in Hong Kong over a decade. For each patient, the average values of a six-month window are calculated to form up the time series with a length of 20 for mortality prediction. Seven parameters of laboratory tests are contained in the PWH dataset, namely, Cr, Hb, Hct, PLT, PT, Urea, and WCC. In this article, we mainly study the laboratory parameter bias caused by the difference in medical devices. Thus, experiments are performed across relevant laboratory parameters. Within these seven laboratory parameters, both Cr and Urea are excretion products that can reflect the liver function. In addition, both PLT and PT are the indicators of the clotting condition/tendency in blood. Therefore, PLT↔PT and Cr↔Urea are selected as two pairs of replacement laboratory parameters in the source and target domains in our experiments. The test results of the other laboratory parameters (Hb, Hct, and WCC) are used as the domain-shared time series in the source and target domains. For short, the domain-shared time series are denoted as $\boldsymbol{C}_P$ in Tables IV and VIII.

The *MIMIC-III dataset* [11] is a publicly available database that records data of patients staying within the intensive care units (ICU) at Beth Israel Deaconess Medical Center. We study the data of adults in ten days, which are less affected by deviations in the sampling process. 38 549 samples are selected from the MIMIC-III dataset [11]. For each sample, the average values in 1/5 day are computed, and time series with 50 time points are used as the input data. In the MIMIC-III dataset [11], results of 374 laboratory parameters are recorded, and the 20 most common parameters are selected in our

Authorized licensed use limited to: University of Southern California. Downloaded on October 11,2023 at 02:46:45 UTC from IEEE Xplore. Restrictions apply.

experiments. Among these records, mean corpuscular volume (MCV) and red cell distribution width (RDW) are the indices related to the size of red blood cells (RBCs). Furthermore, both Hematocrit (Hem) and RBCs measure the amount of RBCs in the blood. Thus, we form up four pairs of datasets by replacing MCV and RDW with each other and replacing Hem and RBC with each other. Time series of the other laboratory parameters are regarded as the domain-shared time series. In Tables V and IX, $\boldsymbol{C}_M$ is used as the abbreviation of domain-shared time series in the MIMIC-III dataset [11].

B. Experimental Settings

In the implementation of a time-series generation network, two-layer LSTMCells are employed in the encoder and decoder. The numbers of units in the LSTMCells are set as 50. The encoded features are the concatenation of the outputs of hidden states along the time axis. Thus, the dimension of the encoded features for each sample is $50 \times l_t$, where l_t is the time length of the source and target time series.

In the missingness-aware feature extraction network, both the source and target LSTM networks are implemented by two-layer LSTMCells with 50 hidden units. The source and target CNNs are the networks with two convolutional layers followed by a fully connected layer. For each domain, the outputs of LSTM and CNN are concatenated as the missingness-aware features for mortality prediction. Both the feature-level and time-series-level domain discriminators are implemented with three fully connected layers: two layers with 50 hidden units followed by the final discriminator output. To extract the missingness patterns, the number of nearest neighbors along the time axis (k_t) is set as 3. Thus, the size of co-occurrence convolution kernels is $n_p \times 7$, where n_p is the number of domain-shared laboratory parameters.

In our experiments, 50% of data are used as the source data and the rest are regarded as the target data. This half-split strategy is adopted because we want to eliminate the imbalance in the amount of data across domains so that the performance degradation across domains is entirely caused by the changes in laboratory parameters. Both the source and target data include the time series of the domain-shared laboratory parameters ($\boldsymbol{C}_P$ and $\boldsymbol{C}_M$ in the PWH and MIMIC-III datasets, respectively). For each pair of replaceable laboratory parameters, one is selected to be contained in the source time series. The time series of these laboratory parameters are replaced by the ones of their related laboratory parameters in the target data. The labels of the source samples are given for training, and the target labels are only available for evaluation. The prediction accuracy (ACC) and the area under the curve (AUC) in the target domain are reported as the evaluation measure. We also plot the ROC curves to show the performance of the proposed method.

C. Medical Time-Series Adaptation Across Different Laboratory Parameters

Experimental results of the proposed method on the PWH and MIMIC-III [11] datasets are reported in Tables II and III, respectively. In these tables, the results of the LSTM networks trained by the source data are listed as the baselines (denoted as LSTM_{Src}). Furthermore, we also select the time series of the target laboratory parameters from the source samples to train an LSTM network. This LSTM network is tested on the target domain, and the performance (denoted as LSTM_{Tgt}) is also reported in Tables II and III for reference. The results of LSTM_{Tgt} reflect the degree of performance drop caused by the distribution bias between the same laboratory parameters in the source and target domains. The difference in accuracies between LSTM_{Src} and LSTM_{Tgt} represents the effect of changing laboratory parameters across domains.

TABLE II
PERFORMANCE (ACC/AUC) OF CROSS-DOMAIN TIME-SERIES ADAPTATION IN THE PWH DATASET

Types of Lab Tests *(Domain-shared: $\boldsymbol{C}_P$)*		LSTM_{Src} *(baseline)*	Ours	LSTM_{Tgt} *(referred)*
Source	Target			
Cr	Urea	83.45/90.83	82.91/89.65	84.73/90.18
Urea	Cr	82.84/88.79	82.26/**89.60**	83.52/89.88
PLT	PT	77.58/85.28	**82.52/88.39**	84.57/90.58
PT	PLT	67.74/83.28	**81.82/88.06**	81.37/87.42
Cr,PLT	Urea,PT	73.89/84.81	**80.76/88.46**	83.07/90.16
Cr,PT	Urea,PLT	63.21/80.06	**83.90/88.41**	81.43/87.51
Urea,PLT	Cr,PT	77.84/82.12	**82.33/89.15**	82.71/89.82
Urea,PT	Cr,PLT	68.38/82.04	**81.01/87.84**	79.22/85.31
Average		74.37/84.65	**82.19/88.67**	81.89/88.39
Standard Deviation		7.38/3.62	7.45/0.69	1.84/1.88

TABLE III
PERFORMANCE (ACC/AUC) OF CROSS-DOMAIN TIME-SERIES ADAPTATION IN THE MIMIC-III DATASET [11]

Types of Lab Tests *(Domain-shared: $\boldsymbol{C}_M$)*		LSTM_{Src} *(baseline)*	Ours	LSTM_{Tgt} *(referred)*
Source	Target			
MCV,Hem	RDW,RBC	81.91/79.56	81.77/77.27	79.84/79.19
MCV,RBC	RDW,Hem	82.30/77.92	81.73/**79.50**	80.69/78.67
RDW,Hem	MCV,RBC	68.13/73.07	**82.42/78.62**	78.18/78.36
RDW,RBC	MCV,Hem	69.60/75.72	**82.05/79.79**	80.61/79.01
Average		75.49/76.57	**81.99/78.80**	79.52/78.67
Standard Deviation		7.67/2.81	0.32/1.13	1.17/0.37

As shown in Table II, compared to the baselines (LSTM_{Src}), the proposed method increases the average ACC and AUC in the PWH dataset by 7.82% and 4.04%, respectively. In addition, the performance deviation of our method across different parameter pairs is acceptable, getting a comparable/lower value of standard deviation on ACC/AUC, compared to the baseline. It is shown that the performance of our method in $\{\boldsymbol{C}_P, \text{Cr}\} \rightarrow \{\boldsymbol{C}_P, \text{Urea}\}$ is not as good as that in the other datasets. That is because results of LSTM_{Src} and LSTM_{Tgt} in $\{\boldsymbol{C}_P, \text{Cr}\} \rightarrow \{\boldsymbol{C}_P, \text{Urea}\}$ are comparable. Therefore, generating target-specific time series could not improve the prediction performance in the target domain.

Furthermore, it is interesting to see that our method even outperforms LSTM_{Tgt} in some pairs of datasets. This indicates that the proposed method not only reduces the domain gap between time series of different laboratory parameters but also reduces the domain shift between time series of the same laboratory parameters in the source and target domains. Consistent results can be found in Table III that records the accuracies of target-domain prediction in the MIMIC-III dataset [11]. The average ACC and AUC of the proposed method are 6.50% and

Authorized licensed use limited to: University of Southern California. Downloaded on October 11,2023 at 02:46:45 UTC from IEEE Xplore. Restrictions apply.

TABLE IV
RESULT (ACC/AUC) COMPARISON OF CROSS-DOMAIN TIME-SERIES ADAPTATION IN THE PWH DATASET (COLORS RED AND BLUE REPRESENT THE BEST AND THE SECOND BEST RESULTS FOR EACH PAIR OF DATASETS, RESPECTIVELY)

Types of Lab Tests (*Domain-shared:* $\boldsymbol{C}_P$)		D-Coral[41]	ADDA[46]	TRL[43]	MCD[48]	Gen2Ad[50]	T-DANN[56]	SWD[49]	GA[59]	Ours
Source	Target									
Cr	Urea	78.51/88.78	**84.22**/**90.56**	77.90/86.68	76.88/84.42	80.92/87.83	81.62/88.95	79.67/83.59	**84.12**/**90.59**	82.91/89.65
Urea	Cr	71.33/78.77	**83.32**/89.00	81.53/88.03	75.85/80.99	81.85/86.90	82.55/88.12	72.67/77.75	**83.10**/**89.12**	82.26/**89.60**
PLT	PT	67.93/79.09	79.47/85.20	79.35/80.20	73.70/76.79	82.04/87.79	**82.68**/84.46	64.40/69.17	79.47/**86.19**	**82.52**/**88.39**
PT	PLT	**77.39**/84.51	75.63/83.52	71.55/81.40	64.24/77.58	65.39/80.93	75.69/**84.58**	65.17/73.66	74.57/84.13	**81.82**/**88.06**
Cr,PLT	Urea,PT	55.84/64.83	78.06/85.81	77.04/84.80	64.82/78.26	77.74/81.84	77.07/85.58	71.74/77.15	**78.64**/**86.59**	**80.76**/**88.46**
Cr,PT	Urea,PLT	56.00/53.39	69.28/81.09	71.94/78.05	55.26/76.87	70.59/**83.33**	**73.96**/81.47	54.91/77.26	72.42/81.16	**83.90**/**88.41**
Urea,PLT	Cr,PT	63.34/53.48	79.38/82.58	76.75/77.76	75.24/81.52	**79.86**/**87.16**	77.58/82.43	70.91/76.27	78.06/83.82	**82.33**/**89.15**
Urea,PT	Cr,PLT	73.09/78.24	73.25/81.02	72.39/78.79	63.92/71.55	72.23/**82.90**	74.47/82.50	69.02/72.33	**74.57**/81.40	**81.01**/**87.84**
Average		67.93/72.64	77.83/84.85	76.06/81.96	68.74/78.50	76.33/84.84	**78.20**/84.76	68.56/75.90	78.12/**85.38**	**82.19**/**88.67**
Standard Deviation		8.86/13.68	5.00/3.52	3.71/4.03	7.79/3.88	6.19/2.87	3.60/2.70	7.28/4.30	4.15/3.40	7.45/0.69

TABLE V
RESULT (ACC/AUC) COMPARISON OF CROSS-DOMAIN TIME-SERIES ADAPTATION IN THE MIMIC-III DATASET [11] (COLORS RED AND BLUE REPRESENT THE BEST AND THE SECOND BEST RESULTS FOR EACH PAIR OF DATASETS, RESPECTIVELY)

Types of Lab Tests (*Domain-shared:* $\boldsymbol{C}_M$)		D-Coral[41]	ADDA[46]	TRL[43]	MCD[48]	Gen2Ad[50]	T-DANN[56]	SWD[49]	GA[59]	Ours
Source	Target									
MCV,Hem	RDW,RBC	82.36/69.68	82.22/78.67	80.48/72.44	79.14/66.89	80.95/74.80	81.53/**78.93**	**82.50**/70.85	**82.41**/**79.84**	81.77/77.27
MCV,RBC	RDW,Hem	80.01/63.02	82.96/76.86	**83.61**/75.30	66.16/69.79	78.43/73.51	80.60/76.82	58.14/66.29	**83.02**/**77.79**	81.73/**79.50**
RDW,Hem	MCV,RBC	**79.95**/63.15	79.45/74.15	75.95/**75.53**	36.83/71.28	79.84/73.89	78.98/75.51	45.82/69.08	77.83/72.12	**82.42**/**78.62**
RDW,RBC	MCV,Hem	80.48/60.88	**80.94**/75.74	76.61/68.63	45.97/73.10	72.13/**78.78**	79.02/75.47	42.96/73.67	72.40/75.47	**82.05**/**79.79**
Average		80.70/64.18	**81.39**/76.38	79.16/72.98	57.02/70.27	77.84/75.25	80.03/**76.68**	57.36/69.97	78.92/76.31	**81.99**/**78.80**
Standard Deviation		1.32/3.81	1.54/1.90	3.56/3.22	19.17/2.63	3.94/2.42	1.25/1.62	18.01/3.10	4.92/3.31	0.32/1.13

2.23% higher than those of the baselines (LSTM_{Src}), respectively. Our method also obtains low standard deviation across different settings of laboratory parameters, which shows the effectiveness of our method under different conditions. Similar to $\{\boldsymbol{C}_P, \text{Cr}\} \rightarrow \{\boldsymbol{C}_P, \text{Urea}\}$ in the PWH dataset, the results of LSTM_{Src} are better than those of LSTM_{Tgt} in $\{\boldsymbol{C}_M, \text{MCV, Hem}\} \rightarrow \{\boldsymbol{C}_M, \text{RDW, RBC}\}$. Therefore, the performance of our method is not so good for $\{\boldsymbol{C}_M, \text{MCV, Hem}\} \rightarrow \{\boldsymbol{C}_M, \text{RDW, RBC}\}$.

We also compare our results with eight state-of-the-art UDA methods: 1) correlation alignment for deep-domain adaptation (D-Coral) [41]; 2) adversarial discriminate-domain adaptation (ADDA) [46]; 3) transferable representation learning (TRL) [43]; 4) maximum classifier discrepancy (MCD) [48]; 5) generate to adapt (Gen2Ad) [50]; 6) LSTM-based domain-adversarial neural network (T-DANN) [56]; 7) sliced Wasserstein discrepancy (SWD) [49]; and 8) graph alignment (GA) [59]. For fairness, the deep-learning networks of all these eight compared methods are implemented by the LSTM networks with the same architecture of LSTM_{Src}. The comparison results in the PWH and MIMIC-III [11] datasets are reported in Tables IV and V, respectively.

In Table IV, the proposed method achieves higher average ACC and AUC than all the other domain adaptation methods. We can find that other domain adaptation methods only slightly improve the performance in the target domain. The ACC and AUC in the target domain may even decrease after the adaptation by some existing domain adaptation methods. This observation supports the statement in Section I that the direct alignment between time series of different laboratory parameters may be ineffective. It is also shown that our method outperforms other UDA methods in most experimental settings, with comparable and lower standard deviation values on ACC and AUC, respectively.

For the results in the MIMIC-III dataset [11], it is shown in Table V that the ACC of the proposed method is not greatly improved, compared to other domain adaptation methods. But the proposed method obtains higher AUC than other domain adaptation methods, while other domain adaptation methods do not obtain any improvement in AUC. In the MIMIC-III dataset [11], data are imbalanced. Nearly 80% of samples in the MIMIC-III dataset [11] are labeled as alive. Even if all samples are predicted to survive, the accuracy of mortality prediction (ACC) can reach 80%. Therefore, the results of AUC are more representative to the performance of different domain adaptation methods in the MIMIC-III dataset [11]. Moreover, comparing the results in the last row in Table V, we can find that our method obtains outstanding (low) standard deviation values among all the UDA methods. This indicates that our method is more stable and effective than the other methods in the MIMIC-III dataset [11].

To intuitively demonstrate the performance of the proposed method, the ROC curves in the PWH and MIMIC-III [11] datasets are plotted in Figs. 6 and 7, respectively. It is shown that the ROC curves of the proposed method are located above the curves of other domain adaptation methods for most pairs of datasets in Figs. 6 and 7. This result reflects the good performance of the proposed method.

D. Heterogeneous Medical Time-Series Adaptation

In practical applications, the number of laboratory parameters can be different in the training and test datasets. To show the performance of our method under this scenario, we conduct

Authorized licensed use limited to: University of Southern California. Downloaded on October 11,2023 at 02:46:45 UTC from IEEE Xplore. Restrictions apply.

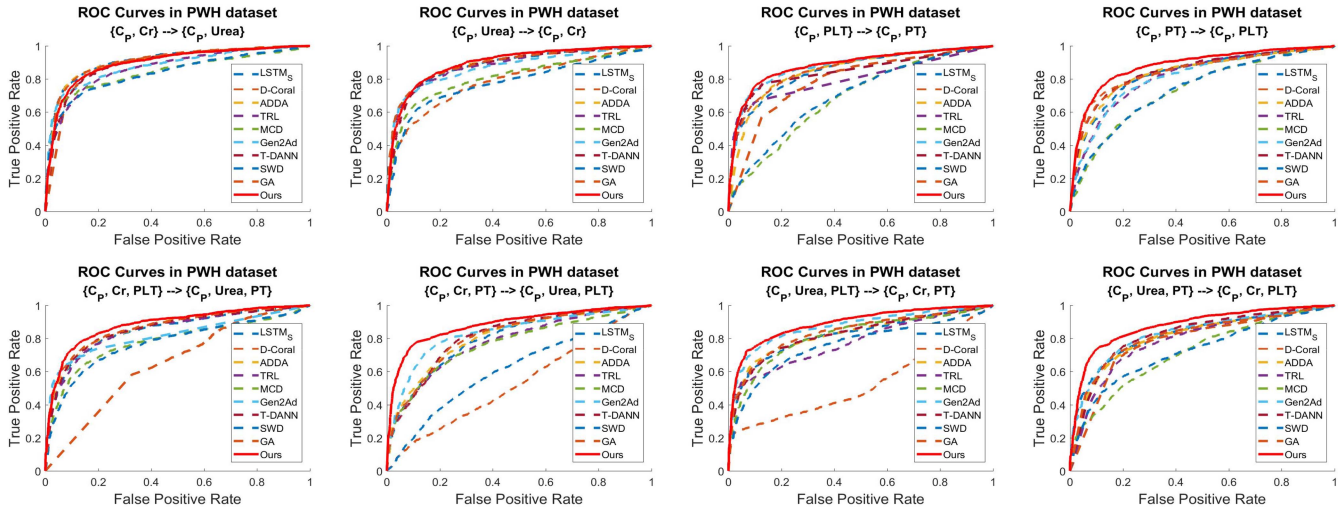

Fig. 6. ROC curves in the PWH dataset.

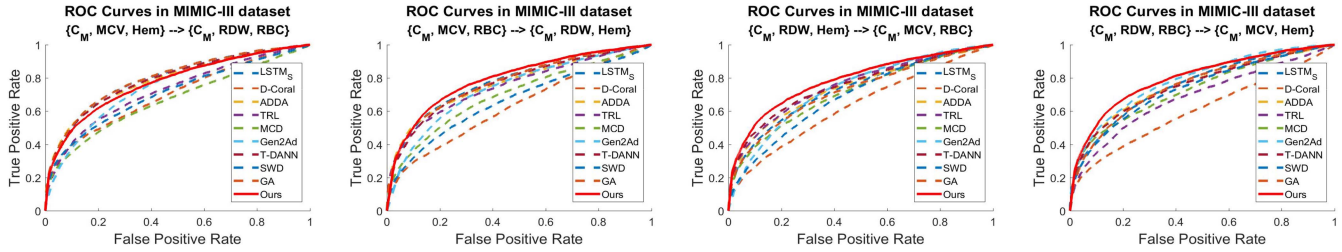

Fig. 7. ROC curves in the MIMIC-III dataset [11].

TABLE VI
PERFORMANCE (ACC/AUC) OF HETEROGENEOUS TIME-SERIES ADAPTATION IN THE PWH DATASET

Types of Blood Tests		LSTM$_{Src}$	Ours
Source	Target		
$\boldsymbol{A_P}$	$\boldsymbol{A_P}$	84.35/90.40	—
$\boldsymbol{A_P}$ - {Urea}	$\boldsymbol{A_P}$	83.52/89.88	**85.28/90.88**
$\boldsymbol{A_P}$ - {Cr}		84.73/**90.18**	**85.44/90.18**
$\boldsymbol{A_P}$ - {PT}		81.37/87.42	**82.87/88.09**
$\boldsymbol{A_P}$ - {PLT}		84.57/90.58	**84.77/91.22**
$\boldsymbol{A_P}$ - {Urea, PT}		79.22/85.31	**81.75/87.08**
$\boldsymbol{A_P}$ - {Urea PLT}		82.71/89.82	**83.96/90.53**
$\boldsymbol{A_P}$ - {Cr, PT}		81.43/87.51	**83.16/88.19**
$\boldsymbol{A_P}$ - {Cr, PLT}		83.07/90.16	**85.21/90.75**
$\boldsymbol{A_P}$ - {Cr, Urea, PLT, PT}		77.04/84.30	**82.49/87.40**
Average		81.96/88.35	**83.88/89.37**
Standard Deviation		2.53/2.33	1.98/1.91

experiments of heterogeneous medical time-series adaptation, in which the target samples include time series of additional unseen laboratory parameters. Our experiments are conducted on the PWH dataset. The time series of all the laboratory parameters (denoted as $\boldsymbol{A_P}$) in the PWH dataset are selected from the target data as the target domain. The source domain consists of source data that excludes time series of one or more laboratory parameters. Nine source domains are formed. They are $\boldsymbol{A_P}$ – {*Urea*}, $\boldsymbol{A_P}$ – {Cr}, $\boldsymbol{A_P}$ – {PT}, $\boldsymbol{A_P}$ – {PLT}, $\boldsymbol{A_P}$ – {Urea, PT}, $\boldsymbol{A_P}$ – {Urea, PLT}, $\boldsymbol{A_P}$ – {Cr, PT}, $\boldsymbol{A_P}$ – {Cr, PLT}, and $\boldsymbol{A_P}$ – {Cr, Urea, PLT, PT}. The time series of additional laboratory parameters in the target domain are the target-specific time series. As no domain-specific time series exists in the source domain, we generate target-specific time series using domain-shared time series in the source domain.

The results of heterogeneous adaptation are reported in Table VI. Time series of $\boldsymbol{A_P}$ are extracted from the source data to train a referred model. The result of this model is reported in the first row in Table VI for reference. In general, the prediction performance decreases with the decrease in the number of laboratory parameters used for training. We also test the performance of the source model in the target time series that exclude the target-specific laboratory parameters. The results of this experiment are denoted as LSTM$_{Scr}$ and shown in Table VI for comparison. It can be found that generating the target-specific time series using the proposed method can increase both ACC and AUC in the target domain for each pair of datasets. In some pairs of datasets, the results of the proposed method are even as good as the results of $\boldsymbol{A_P} \rightarrow \boldsymbol{A_P}$, which shows the effectiveness of the proposed method.

E. Statistical Analysis

This section presents the statistical experiments, which test the stability of the proposed method for different sample populations. The MIMIC-III dataset [11] is selected for experiments. For each pair of laboratory parameters, a cross-domain experiment is conducted five times. In each trial, half of the samples in the MIMIC-III dataset are randomly selected as the source domain while the remaining samples are the target domain. We test the performance of the source model and the proposed method in the target domain for each trail. The mean

Authorized licensed use limited to: University of Southern California. Downloaded on October 11,2023 at 02:46:45 UTC from IEEE Xplore. Restrictions apply.

TABLE VII
STABILITY ANALYSIS (ACC/AUC) OF CROSS-DOMAIN TIME-SERIES ADAPTATION IN THE MIMIC-III DATASET [11]

Types of Lab Tests *(Domain-shared: $\boldsymbol{C}_M$)*		LSTM$_{Src}$ *(baseline)*	Ours
Source	Target		
MCV,Hem	RDW,RBC	81.81±1.39 / 79.07±0.96	81.79±0.37 / 78.63±1.06
MCV,RBC	RDW,Hem	82.05±0.29 / 79.36±1.08	82.25±0.38 / 79.83±0.35
RDW,Hem	MCV,RBC	69.03±1.17 / 74.08±0.98	82.19±0.16 / 79.15±0.51
RDW,RBC	MCV,Hem	69.85±1.47 / 75.67±0.51	82.23±0.49 / 79.27±0.46

TABLE VIII
PERFORMANCE (ACC/AUC) OF ABLATION STUDIES IN THE PWH DATASET

Types of Blood Tests		LSTM$_{Src}$	$\boldsymbol{Q}_1$ = M-Patterns	$\boldsymbol{Q}_1$ + Feat-Align	$\boldsymbol{Q}_1$ + Gen	$\boldsymbol{Q}_1$ + Feat-Align + Gen
Source	Target					
$\boldsymbol{C}_P$, Cr	$\boldsymbol{C}_P$, Urea	**83.45/90.83**	82.04/90.35	82.62/89.85	83.26/89.53	82.91/89.65
$\boldsymbol{C}_P$, Urea	$\boldsymbol{C}_P$, Cr	**82.84**/88.79	85.70/91.17	81.21/89.55	82.23/89.46	82.26/**89.60**
$\boldsymbol{C}_P$, PLT	$\boldsymbol{C}_P$, PT	77.58/85.28	80.60/86.51	82.36/88.32	**83.03**/88.21	82.52/**88.39**
$\boldsymbol{C}_P$, PT	$\boldsymbol{C}_P$, PLT	67.74/83.28	64.05/79.86	73.16/79.52	80.98/87.12	**81.82/88.06**
$\boldsymbol{C}_P$, Cr, PLT	$\boldsymbol{C}_P$, Urea, PT	73.89/84.81	78.29/81.58	79.96/83.04	**82.10**/88.33	80.76/**88.46**
$\boldsymbol{C}_P$, Cr, PT	$\boldsymbol{C}_P$, Urea, PLT	63.21/80.06	61.83/76.41	71.58/78.54	83.89/88.35	**83.90/88.41**
$\boldsymbol{C}_P$, Urea, PLT	$\boldsymbol{C}_P$, Cr, PT	77.84/82.12	80.08/86.83	82.10/88.51	**82.39**/89.12	82.33/**89.15**
$\boldsymbol{C}_P$, Urea, PT	$\boldsymbol{C}_P$, Cr, PLT	68.38/82.04	67.58/77.50	69.31/76.84	**81.30**/87.80	81.01/**87.84**
Average		74.37/84.65	75.02/83.78	77.79/84.27	**82.40**/88.49	82.19/**88.67**

TABLE IX
PERFORMANCE (ACC/AUC) OF ABLATION STUDIES IN THE MIMIC-III DATASET [11]

Types of Blood Tests		LSTM$_{Src}$	$\boldsymbol{Q}_1$ = M-Patterns	$\boldsymbol{Q}_1$ + Feat-Align	$\boldsymbol{Q}_1$ + Gen	$\boldsymbol{Q}_1$ + Feat-Align + Gen
Source	Target					
$\boldsymbol{C}_M$, MCV, Hem	$\boldsymbol{C}_M$, RDW, RBC	81.91/79.56	81.47/**80.23**	**83.06**/80.07	79.28/77.49	81.77/77.27
$\boldsymbol{C}_M$, MCV, RBC	$\boldsymbol{C}_M$, RDW, Hem	**82.30**/77.92	81.12/78.25	81.84/77.92	80.48/79.05	81.73/**79.50**
$\boldsymbol{C}_M$, RDW, Hem	$\boldsymbol{C}_M$, MCV, RBC	68.13/73.07	69.73/75.74	81.52/78.00	80.71/**78.78**	**82.42**/78.62
$\boldsymbol{C}_M$, RDW, RBC	$\boldsymbol{C}_M$, MCV, Hem	69.60/75.72	71.58/76.98	81.33/76.69	80.10/79.54	**82.05/79.79**
Average		75.49/76.57	75.98/77.80	81.94/78.17	80.14/78.72	**81.99/78.80**

and standard deviation of multiple test results are calculated and listed in Table VII. It is shown that the standard deviations of ACC and AUC obtained by the source model are lower than 1.5 and 1.1, respectively. This result reflects that the source model is less sensitive to the change of training/test samples. Compared to the source model, the performance deviation of our method is relatively low, with values of standard deviations less than 1.1 on both ACC and AUC. The standard deviation on some laboratory parameter pairs (e.g., {$\boldsymbol{C}_M$, RDW, Hem} $\longrightarrow$ {$\boldsymbol{C}_M$, MCV, RBC}) is even lower than that of the baseline (LSTM$_{\text{Src}}$). This indicates that the performance of our method is stable among different sample populations.

F. Ablation Studies

To further analyze the contribution of each component in the proposed method, we perform ablation studies by gradually adding different components of the proposed method. The results of ablation studies in the PWH and MIMIC-III [11] datasets are presented in Tables VIII and IX, respectively. In Tables VIII and IX, "M-Patterns" denotes the component of missingness-aware feature extraction network, "Gen" means time-series generation without feature-space alignment, and "Feat-Align" refers the alignment achieved by the feature-level adversarial network.

As shown in Table VIII, the performance is not improved by merely adding missingness patterns. That is because time series of different laboratory tests have different missingness patterns. The missingness patterns of one laboratory parameter may not help to classify the time series of the other laboratory parameters. However, we have verified that missingness patterns could improve the prediction of the same laboratory parameters. In specific, ACC/AUC of $\boldsymbol{C}_P \rightarrow \boldsymbol{C}_P$ is increased from 77.04/84.30 to 80.44/86.33 by adding missingness patterns for mortality prediction. It is shown in Table VIII, the improvement in the PWH dataset is mainly contributed by the component of "Gen" that refers to the proposed time-series generation network with similarity distillation. Only a minor improvement can be obtained by adding the component of "Feat-Align" to $\boldsymbol{Q}_1$. As data are imbalanced in the MIMIC-III dataset [11], we focus on analyzing the AUC values in Table IX. We can see that missingness patterns slightly improve the prediction performance in the MIMIC-III dataset [11]. The accuracies are further increased by adding the component of Gen or Feat-Align. Better results are achieved by applying both components of Gen and Feat-Align.

G. Visualization of Feature Distribution

In this section, we visualize the feature distribution of the generated target time series using t-SNE [60] to intuitively show the effect of the proposed method. The dataset of {$\boldsymbol{C}_P$, Cr, PT} $\rightarrow$ {$\boldsymbol{C}_P$, Urea, PLT} is selected for visualization. The features of the generated and the real target time series are plotted in Fig. 8(a). We also extract features of the real source and target time series using LSTM$_{Scr}$. These features are plotted in Fig. 8(b) for comparison.

Authorized licensed use limited to: University of Southern California. Downloaded on October 11,2023 at 02:46:45 UTC from IEEE Xplore. Restrictions apply.

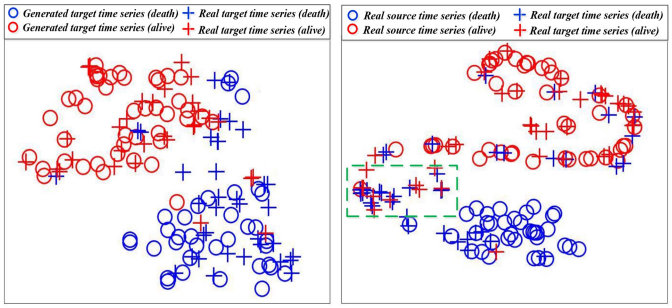

Fig. 8. Feature distribution of the generated and the real medical time series. (a) Proposed method. (b) LSTM_{Src}.

We can find that in Fig. 8(b), features of some real target time series (samples in the green dotted rectangle) are located away from the features of any source time series. The predicting mortality for these samples is difficult as their features are different from both positive (death) and negative (alive) samples in the source domain. In contrast, in Fig. 8(a), the features of each real target sample are more likely to be located near the features of certain generated target time series. Therefore, label information can be more easily propagated to the real target time series via the generated time series. This indicates that the bias between different laboratory parameters is reduced in the feature space by generating target time series using the proposed method.

V. Conclusion

This article proposes a medical time series adaptation method to address the problem of laboratory parameter bias between time series in the training and test datasets. A time-series generation network with similarity distillation is developed to reduce the domain gap caused by the difference of laboratory parameters across the source and target domains. We also proposed a missingness-aware feature extraction network to extract missingness patterns from medical time series as auxiliary features for cross-domain medical applications. The experimental results show that the proposed method achieves good performance across replaceable laboratory parameters in two medical time-series datasets.

In this article, we mainly study the problem of laboratory parameter bias that is caused by the change of test equipment across the training and test datasets. The series substitution occurs between relevant laboratory parameters. The prior medical knowledge of laboratory tests may restrict the popularization of our model in practical scenarios. Thus, analyzing the underlying relations between different laboratory parameters is one of our future studies. The relation analysis can then be introduced in the proposed method to automatically match replaceable laboratory parameters in different datasets, thus expanding the scope of application. Moreover, the time lengths of time series are assumed to be the same in the source and target domains, which limits the application range of the proposed model. In the future, we will try to relax the constraint of time length in the generator so that the source model can be adapted to time series with different lengths. We will also try to develop an adaptive model for medical time series with multiple patterns, which improves the model flexibility in clinical application.

Acknowledgment

The authors would like to thank the colleagues from the Department of Computer Science, Hong Kong Baptist University and the researchers from the Machine Intelligence Laboratory, the University of Tokyo for their precious comments and suggestions.

References

[1] K. Häyrinen, K. Saranto, and P. Nykänen, "Definition, structure, content, use and impacts of electronic health records: A review of the research literature," *Int. J. Med. Informat.*, vol. 77, no. 5, pp. 291–304, 2008.

[2] P. Yadav, M. Steinbach, V. Kumar, and G. Simon, "Mining electronic health records (EHRs): A survey," *ACM Comput. Surveys*, vol. 50, no. 6, p. 85, 2018.

[3] I. M. Baytas, C. Xiao, X. Zhang, F. Wang, A. K. Jain, and J. Zhou, "Patient subtyping via time-aware LSTM networks," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Disc. Data Min.*, 2017, pp. 65–74.

[4] Z. Che, S. Purushotham, K. Cho, D. Sontag, and Y. Liu, "Recurrent neural networks for multivariate time series with missing values," *Sci. Rep.*, vol. 8, no. 1, p. 6085, 2018.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[7] M. Ye, J. Shen, X. Zhang, P. C. Yuen, and S.-F. Chang, "Augmentation invariant and instance spreading feature for softmax embedding," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access.

[8] J. Boursier *et al.*, "Diagnostic accuracy and prognostic significance of blood fibrosis tests and liver stiffness measurement by fibroscan in non-alcoholic fatty liver disease," *J. Hepatol.*, vol. 65, no. 3, pp. 570–578, 2016.

[9] B. Shickel, P. J. Tighe, A. Bihorac, and P. Rashidi, "Deep EHR: A survey of recent advances in deep learning techniques for electronic health record (EHR) analysis," *IEEE J. Biomed. Health Informat.*, vol. 22, no. 5, pp. 1589–1604, Sep. 2018.

[10] Q. Tan *et al.*, "UA-CRNN: Uncertainty-aware convolutional recurrent neural network for mortality risk prediction," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manag.*, 2019, pp. 109–118.

[11] A. E. Johnson *et al.*, "Mimic-III, a freely accessible critical care database," *Sci. Data*, vol. 3, May 2016, Art. no. 160035.

[12] R. Miotto, L. Li, B. A. Kidd, and J. T. Dudley, "Deep patient: An unsupervised representation to predict the future of patients from the electronic health records," *Sci. Rep.*, vol. 6, May 2016, Art. no. 26094.

[13] E. Choi, M. T. Bahadori, A. Schuetz, W. F. Stewart, and J. Sun, "Doctor AI: Predicting clinical events via recurrent neural networks," in *Proc. Mach. Learn. Healthcare Conf.*, 2016, pp. 301–318.

[14] A. Gharehbaghi and M. Lindén, "A deep machine learning method for classifying cyclic time series of biological signals using time-growing neural network," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 9, pp. 4102–4115, Sep. 2018.

[15] R. Gopalan, R. Li, and R. Chellappa, "Domain adaptation for object recognition: An unsupervised approach," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 999–1006.

[16] M. Jing, J. Zhao, J. Li, L. Zhu, Y. Yang, and H. T. Shen, "Adaptive component embedding for domain adaptation," *IEEE Trans. Cybern.*, early access, Mar. 6, 2020, doi: 10.1109/TCYB.2020.2974106.

[17] L. Li, Z. Wan, and H. He, "Dual alignment for partial domain adaptation," *IEEE Trans. Cybern.*, early access, Apr. 29, 2020, doi: 10.1109/TCYB.2020.2983337.

[18] Y. Yang and J. Eisenstein, "Unsupervised multi-domain adaptation with feature embeddings," in *Proc. Conf. North Amer. Assoc. Comput. Linguist. Human Lang. Technol.*, 2015, pp. 672–682.

[19] Y. Ziser and R. Reichart, "Pivot based language modeling for improved neural domain adaptation," in *Proc. Conf. North Amer. Assoc. Comput. Linguist. Human Lang. Technol.*, vol. 1, 2018, pp. 1241–1251.

Authorized licensed use limited to: University of Southern California. Downloaded on October 11,2023 at 02:46:45 UTC from IEEE Xplore. Restrictions apply.

[20] S. Purushotham, W. Carvalho, T. Nilanon, and Y. Liu, "Variational recurrent adversarial deep domain adaptation," in *Proc. Int. Conf. Learn. Represent.*, 2016, p. 6.
[21] K. Yan, L. Kou, and D. Zhang, "Learning domain-invariant subspace using domain features and independence maximization," *IEEE Trans. Cybern.*, vol. 48, no. 1, pp. 288–299, Jan. 2018.
[22] Q. Tan *et al.*, "Data-GRU: Dual-attention time-aware gated recurrent unit for irregular multivariate time series," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, 2020, pp. 930–937.
[23] H.-G. Kim, G.-J. Jang, H.-J. Choi, M. Kim, Y.-W. Kim, and J. Choi, "Recurrent neural networks with missing information imputation for medical examination data prediction," in *Proc. IEEE Int. Conf. Big Data Smart Comput.*, 2017, pp. 317–323.
[24] J. Grabocka, N. Schilling, M. Wistuba, and L. Schmidt-Thieme, "Learning time-series ShapeLets," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Disc. Data Min.*, 2014, pp. 392–401.
[25] X. Wang *et al.*, "RPM: Representative pattern mining for efficient time series classification." in *Proc. EDBT*, 2016, pp. 185–196.
[26] C. Bock, T. Gumbsch, M. Moor, B. Rieck, D. Roqueiro, and K. Borgwardt, "Association mapping in biomedical time series via statistically significant shapelet mining," *Bioinformatics*, vol. 34, no. 13, pp. i438–i446, 2018.
[27] A. G. Maletzke, C. A. Ferrero, C. M. Tibes, E. A. Cherman, and W. Zalewski, "Medical time series classification using global and local feature extraction strategies," *J. Health Informat.*, vol. 9, no. 3, pp. 73–80, 2017.
[28] R. J. Kate, "Using dynamic time warping distances as features for improved time series classification," *Data Min. Knowl. Disc.*, vol. 30, no. 2, pp. 283–312, 2016.
[29] Z. C. Lipton, D. C. Kale, C. Elkan, and R. Wetzel, "Learning to diagnose with LSTM recurrent neural networks," in *Proc. Int. Conf. Learn. Represent.*, 2016, pp. 1–6.
[30] T. Pham, T. Tran, D. Phung, and S. Venkatesh, "Predicting healthcare trajectories from medical records: A deep learning approach," *J. Biomed. Informat.*, vol. 69, pp. 218–229, Mar. 2017.
[31] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[32] Y. Wang, D. J. Miller, K. Poskanzer, Y. Wang, L. Tian, and G. Yu, "Graphical time warping for joint alignment of multiple curves," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3648–3656.
[33] M. Uzair and A. Mian, "Blind domain adaptation with augmented extreme learning machine features," *IEEE Trans. Cybern.*, vol. 47, no. 3, pp. 651–660, Mar. 2017.
[34] J. Li, K. Lu, Z. Huang, L. Zhu, and H. T. Shen, "Transfer independently together: A generalized framework for domain adaptation," *IEEE Trans. Cybern.*, vol. 49, no. 6, pp. 2144–2155, Jun. 2019.
[35] H. Daumé, III, A. Kumar, and A. Saha, "Frustratingly easy semi-supervised domain adaptation," in *Proc. Workshop Domain Adapt. Nat. Lang. Process.*, 2010, pp. 53–59.
[36] J. Donahue, J. Hoffman, E. Rodner, K. Saenko, and T. Darrell, "Semi-supervised domain adaptation with instance constraints," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 668–675.
[37] T. Yao, Y. Pan, C.-W. Ngo, H. Li, and T. Mei, "Semi-supervised domain adaptation with subspace learning for visual recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 2142–2150.
[38] B. Gong, K. Grauman, and F. Sha, "Connecting the dots with landmarks: Discriminatively learning domain-invariant features for unsupervised domain adaptation," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 222–230.
[39] R. Aljundi, R. Emonet, D. Muselet, and M. Sebban, "Landmarks-based kernelized subspace alignment for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 56–63.
[40] B. Fernando, A. Habrard, M. Sebban, and T. Tuytelaars, "Unsupervised visual domain adaptation using subspace alignment," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 2960–2967.
[41] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 443–450.
[42] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 97–105.
[43] M. Long, Y. Cao, Z. Cao, J. Wang, and M. I. Jordan, "Transferable representation learning with deep adaptation networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 2, pp. 3071–3085, Dec. 2019.
[44] A. Gretton, K. M. Borgwardt, M. Rasch, B. Schölkopf, and A. J. Smola, "A kernel method for the two-sample-problem," in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, pp. 513–520.
[45] I. J. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.
[46] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, vol. 1, 2017, p. 4.
[47] C. Chen *et al.*, "Progressive feature alignment for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 627–636.
[48] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada, "Maximum classifier discrepancy for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3723–3732.
[49] C.-Y. Lee, T. Batra, M. H. Baig, and D. Ulbricht, "Sliced Wasserstein discrepancy for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 10285–10295.
[50] S. Sankaranarayanan, Y. Balaji, C. D. Castillo, and R. Chellappa, "Generate to adapt: Aligning domains using generative adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8503–8512.
[51] R. Volpi, P. Morerio, S. Savarese, and V. Murino, "Adversarial feature augmentation for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5495–5504.
[52] F. Lv, J. Zhu, G. Yang, and L. Duan, "TarGAN: Generating target data with class labels for unsupervised domain adaptation," *Knowl. Based Syst.*, vol. 172, pp. 123–129, May 2019.
[53] T. Alves, A. Laender, A. Veloso, and N. Ziviani, "Dynamic prediction of ICU mortality risk using domain adaptation," in *Proc. IEEE Int. Conf. Big Data*, 2018, pp. 1328–1336.
[54] P. Gupta, P. Malhotra, J. Narwariya, L. Vig, and G. Shroff, "Transfer learning for clinical time series analysis using deep neural networks," 2019. [Online]. Available: arXiv:1904.00655.
[55] J. Chung, K. Kastner, L. Dinh, K. Goel, A. C. Courville, and Y. Bengio, "A recurrent latent variable model for sequential data," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 2980–2988.
[56] P. R. D. O. da Costa, A. Akcay, Y. Zhang, and U. Kaymak, "Remaining useful lifetime prediction via deep domain adaptation," 2019. [Online]. Available: arXiv:1907.07480.
[57] S. Zheng, K. Ristovski, A. Farahat, and C. Gupta, "Long short-term memory network for remaining useful life estimation," in *Proc. IEEE Int. Conf. Prognostics Health Manag. (ICPHM)*, 2017, pp. 88–95.
[58] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 3104–3112.
[59] B. Yang and P. C. Yuen, "Cross-domain visual representations via unsupervised graph alignment," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 5613–5620.
[60] L. v. d. Maaten and G. Hinton, "Visualizing data using T-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.

Baoyao Yang received the B.Sc. degree in computer science and technology from the South China University of Technology, Guangzhou, China, in 2014, and the Ph.D. degree in computer science from Hong Kong Baptist University, Hong Kong, in 2018.

Since graduation, she has been working as a Postdoctoral Research Fellow with the Department of Computer Science, Hong Kong Baptist University. In 2019, she spent a two-month sabbatical leave to work as a Visiting Researcher with the Machine Intelligence Laboratory, Department of Information Science and Technology, University of Tokyo, Tokyo, Japan. Her current research interests include transfer learning, cross-modal adaptation, and machine learning in medical applications.

Mang Ye received the B.S. and M.S. degrees in electronic information from Wuhan University, Wuhan, China, in 2013 and 2016, respectively, and the Ph.D. degree in computer science from the Department of Computer Science, Hong Kong Baptist University, Hong Kong, in 2019.

He is currently a Research Scientist with the Inception Institute of Artificial Intelligence, Abu Dhabi, UAE. His research interests focus on multimedia retrieval, computer vision, and pattern recognition.

Authorized licensed use limited to: University of Southern California. Downloaded on October 11,2023 at 02:46:45 UTC from IEEE Xplore. Restrictions apply.

Qingxiong Tan received the B.S. degree from the China University of Mining and Technology, Xuzhou, China, in 2014, and the M.S. degree from the Huazhong University of Science and Technology, Wuhan, China, in 2017. He is currently pursuing the Ph.D. degree with the Department of Computer Science, Hong Kong Baptist University, Hong Kong.

His research interests focus on machine learning and data mining, including deep learning, medical data mining, and time-series prediction.

Pong C. Yuen (Senior Member, IEEE) received the B.Sc. degree in electronic engineering (with First Class Hons.) from the City Polytechnic of Hong Kong, Hong Kong, in 1989, and the Ph.D. degree in electrical and electronic engineering from the University of Hong Kong, Hong Kong, in 1993.

He joined the Hong Kong Baptist University, Hong Kong, in 1993, where he served as the Head of the Department of Computer Science from 2011 to 2017. He is currently a Chair Professor with the Department of Computer Science and an Associate Dean of science faculty with Hong Kong Baptist University. In 1998, he spent a six-month sabbatical leave with the University of Maryland Institute for Advanced Computer Studies, University of Maryland at College Park, College Park, MD, USA. From June 2005 to January 2006, he was a Visiting Professor with GRAVIR Laboratory (Graphics, Vision and Robotics), INRIA Rhone Alpes, Montbonnot-Saint-Martin, France. From July 2017 to January 2018, he was a Visiting Faculty with the ETH Zurich, Zürich, Switzerland. He was the Director of Croucher Advanced Study Institute on Biometric Authentication in 2004 and the Croucher ASI on Biometric Security and Privacy in 2007. He has been serving as the Director of IAPR/IEEE Winter School on Biometrics since 2017. He has been actively involving in many international conferences and professional community. His current research interests include video surveillance, human face recognition, and biometric security and privacy.

Dr. Yuen was a recipient of the University Fellowship to Visit the University of Sydney in 1996 and the Outstanding Editorial Board Service Award in 2018. He received the first-prize and second-prize Natural Science Awards from the Guangdong Province and the Ministry of Education, China, respectively. He was the Track Co-Chair of the International Conference on Pattern Recognition in 2006, the Program Co-Chair of the IEEE Fifth International Conference on Biometrics: Theory, Applications and Systems in 2012, the IEEE International Conference on Identity, Security and Behavior Analysis in 2016, the International Conference on Pattern Recognition and Artificial Intelligence in 2018 and 2020, and the International Workshop on Information, Forensics and Security in 2018. He served as an Associate Editor for the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY from 2014 to 2018. He has also served as the Vice President (Technical Activities) of the IEEE Biometrics Council from 2017 to 2019, and an Associate Editor/Senior Editor of *SPIE Journal of Electronic Imaging* from 2012 to 2019. He is currently the Editorial Board Member of *Pattern Recognition*. He is a Fellow of IAPR.

Authorized licensed use limited to: University of Southern California. Downloaded on October 11,2023 at 02:46:45 UTC from IEEE Xplore. Restrictions apply.