Deep generative model with domain adversarial training for predicting arterial blood pressure waveform from photoplethysmogram signal

Keke Qin [a],*, Wu Huang [b], Tao Zhang [a]

[a] *Chengdu Techman Software Co.,Ltd, Chengdu, SC, 610000, China*
[b] *School of Computer Science, Sichuan University, Chengdu, SC, 610000, China*

ARTICLE INFO

ABSTRACT

Background and Motivations: Continuous blood pressure (BP) monitoring is of critical importance to health state tracking and disease prevention. However, current mainstream BP measurement approaches are cuff-based, which is inconvenient and limit its usage scenarios. Predicting arterial blood pressure (ABP) could provide richer information than isolated BP values. The individual differences among data may hinder training.
Methods: A novel continuous, non-invasive and cuff-less approach is presented for generating ABP waveform using only raw photoplethysmogram (PPG) signal, from a signal conversion perspective, where a convolution-based deep autoencoder (DAE) model is developed. To overcome individual differences, Multi-domain adversarial training is merged with DAE (abbr. RDAE) to learn cross-domain features, and partial data is further used to calibrate (optional) the general model.
Results: The mean absolute error (MAE) of uncalibrated RDAE reached 7.945, 4.114 and 3.834 mmHg in systolic BP (SBP), diastolic BP (DBP) and mean BP (MBP) prediction. After using 80 s data for calibration, the MAE of RDAE reduced to 5.424, 3.144 and 2.885 mmHg accordingly.
Conclusion: Owning to the high-quality converted ABP segments, the resulting estimated BP is accurate. According to the BHS standard, RDAE achieved Grade C, Grade A and Grade A for SBP, DBP and MBP prediction, and the calibrated RDAE achieved Grade B, Grade A, Grade A accordingly.
Significance: Both domain adversarial training and calibration improve the performance in varying degrees. RDAE is competitive to other mainstream regression-based deep learning methods, while with fewer model parameters, and to other representative machine learning methods, while no need of complicated feature engineering.

1. Introduction

As an important vital signal, blood pressure (BP) is a critical predictor to determine the cardiovascular health of patients [1]. Furthermore, long-term blood pressure monitoring can dynamically reflect the changes in the individual's health status, which is of great significance for the warning and prevention of potential diseases [2,3].

Continuous BP monitoring based on wearable sensors has increasingly broad development prospects [4]. Especially, Photoplethysmography (PPG) has achieved a wide range of applications [5,6] because of its cheapness and convenience, e.g. in blood pressure measurement. Concretely, with PPG sensor, PPG signal could be measured at wrist or finger, which is much convenient than the collection of Electrocardiogram (ECG) signal where multiple electrodes are required to attach at wrist and chest. In fact, electronic equipments with PPG sensor embedded, e.g. the smart bracelet, are becoming more and more popular, leading a healthy lifestyle.

The PPG and arterial BP (ABP) signals are intrinsically related, which is the potential hypothesis of related studies that predicting BP with PPG signal. Actually, the PPG and ABP signals have certain similarities in geometry [7]. From a physiological perspective, blood pressure is highly-related to the flow rate of blood [8]. Concretely, when blood vessels begin to contract, blood flows faster and impel higher pressure. On the contrary, when blood vessels gradually relax, blood flows slower and bring less pressure. Further, PPG signal could be disassembled into AC part and DC part, and the AC part is associated with the pulse rate of arterial blood [6].

Currently, there have extensive approaches to measure blood pressure by using PPG signal, ECG signal or both [1,2,8–22]. These methods

---

follow similar circuit: extract features implicitly (for deep learning-based methods) or explicitly (for traditional machine learning methods) from preprocessed signals and then solving prediction model by regressing predicted value to target value, the predictive model could be explicitly defined (e.g. PTT method [9] and its variants [2,10], as well as data-driven machine learning methods [11–14]) or implicitly defined (e.g. DL-based methods [1,8,15–22]). While, differently from the current mainstream methods that directly predicting blood pressure value based on physiological signals, such as PPG and ECG signal etc., based on the truth that there is a high similarity between the PPG and ABP morphologies revealed by several studies [7,23], we try to visualize blood pressure waveform with raw PPG signal only, which is actually a sequence-to-sequence conversion question. The advantage is that, first, in addition to the obtained blood pressure value, the blood pressure change curve is also visualized. Second, no troublesome feature extraction and feature selection/transformation is needed. Besides, if ABP waveform could be generated from the counterpart PPG signal, then this approach has potential application value—as an alternative to invasive ABP [7], it is more safe without the risk of infection.

To realize this target, we resort to generative model. Specifically, Autoencoder (AE) [24] model, a classical approach in unsupervised learning, is employed because of its simplicity and naturally sequence to sequence architecture with high symmetry. As for AE architecture, AE could be designed based on fully-connected module, convolution operation [25] or recurrent neural network (RNN) module and its variants [26]. Although RNN module is a common choice for handling time series data [18,27,28]. Whereas, RNN module propagate information from one timestep to the next by keeping it in their hidden state, which limits their capacity to perform complex computation on input stream [29]. In this study, taking into account the periodicity and pattern repetition properties of physiological signals (e.g. PPG and ABP signal), we build the proposed approach based on 1D convolution.

Another noteworthy question in this area is individual differences [17]. Specifically, different subjects exhibit PPG signals with different shape [8,14,30], and have different health state, resulting in differently blood pressure level and change pattern. Note that this phenomenon is more severe in inpatients, especially in intensive care unit patients. Individual differences propose new challenges to generalizability and robustness of prediction model. On the one hand, a model should keep enough tolerance to input signals with different shape and quality. Whereas, much of studies in this area heavily depends on feature engineering. Because extraction of features relies on positioning feature points, the data used in practice is elaborately processed and screened [8,15]. Therefore, further verification of the performance of these methods is needed in real scene. On the other hand, for a new out-of-domain individual, the model should be capable of quickly adapted using minor data of the new individuals. In the presented work, we regard each individual's record as a domain, and the thought of domain adversarial training [31] is absorbed to overcome the above obstacle. Specifically, in addition to the convolution-based deep autoencoder, an auxiliary domain classifier (connected to the latent code vector of AE) is maintained and is used to train the encoder part of AE adversarial to make the learned latent code vector informative (for the PPG-to-ABP signal conversion task) as well as general among different individuals (i.e. the so-called domain-invariant). The domain adversarial training essentially plays the role of 'regularizer' to impose the latent code vector more general across different subjects. Therefore, the proposed approach is called regularized deep autoencoder (RDAE). Fig. 1 presents a top-level description of RDAE for blood pressure measurement.

The main contributions of this study are summarized as follows:

- Different from the currently mainstream regression-based approaches that directly predict BP values, we study ABP waveform prediction from a signal conversion perspective, one advantage of which is that not only BP value but also ABP waveform are reported;

- A convolution-based deep autoencoder model is developed for converting PPG signal to ABP waveform. The results show that the proposed model is competitive compared with other state-of-the art deep learning methods. More importantly, the number of parameters of the model is far less than other methods. In addition, we presented RDAE with different configurations, providing a flexible model selection for inference in practice, by trading off between the accuracy and model size (complexity);

- Taking into account the huge individual differences, domain adversarial training is embedded into DAE to train a more general model. Then, minor data of test subject (only 80 s) is used to fine-tune the model, which significantly improves the prediction accuracy.

The rest of this study is organized as follows: related work is reviewed in Section 2. In Section 3, we describe the proposed approach—RDAE. Section 4 is experimental protocol, which is followed by results & analysis in Section 5. Further interpretation and limitation of the method are given in Sections 6, 7, respectively. Last, we conclude this study in Section 8.

2. Related work review

We mainly review related progress from the follow three categories, and a comprehensive comparison of related studies is summarized into a Table A.1 (served as supplementary material).

2.1. Traditional ML for BP prediction

As for traditional machine learning (ML) methods [32] for BP prediction, apart from prediction algorithm, feature engineering is an extremely important step [8,15]. Therefore, we review related work from these two aspects.

Feature engineering: Moreno et al. [11] firstly extracts the features comprehensively from PPG that required for build BP prediction model. Kachuee et al. [14] systematically investigate two types of features—physiological parameters and whole-based features that are required for BP prediction. Xing et al. [33] use amplitude and phase information of PPG signal based on FFT for beat-to-beat BP prediction. Thambiraj et al. [2,34] propose a new feature—Womersley number based on PPG signal. Fujita et al. [35] defines level-crossing features (LCFs) based on derivatives of PPG for BP prediction. Bose S et al. [36] firstly model the feature extraction process as dictionary learning and the sparse representation of raw PPG signal is used for beat-to-beat BP prediction.

Prediction algorithm: Pulse transit time (PTT) method and relevant improved methods [2,9,10] are very popular, these methods are effective when evaluated on healthy individuals and outpatients. Whereas otherwise when evaluated on inpatients, especially intensive care unit (ICU) patients. As patients suffer from various symptoms, such as bleeding, and affected by drugs, and PTT method is an ideal model, and the accuracy of which is influenced by several factors. Therefore, frequent calibration is needed to ensure accuracy [37]. In contrast, ML algorithms are suitable because of its powerful ability in learning the complex mapping relation between input and output. Multiple-layer perceptron (MLP) is widely used in related work [33,38,39]. In addition, classical regression approaches, such as support vector regression (SVR) [13,34], random forest (RF) [11,14,30,34], decision tree [40] etc., has been used.

To a large extent, these researches rely on the location of waveform feature points to extract features, urging the necessity of high-quality waveform [8,15]. Therefore, an elaborate signal preprocessing and screening in need to ensure the quality of data, in addition to the complicated feature extraction and feature selection/transformation process [11]. Inherited the ability of deep learning to learn useful features automatically [41], our study is differentiated from the above work lies in that raw PPG signal is directly fed with minimal preprocessing (Ref Section 4.1), and does not require complicated feature engineering.
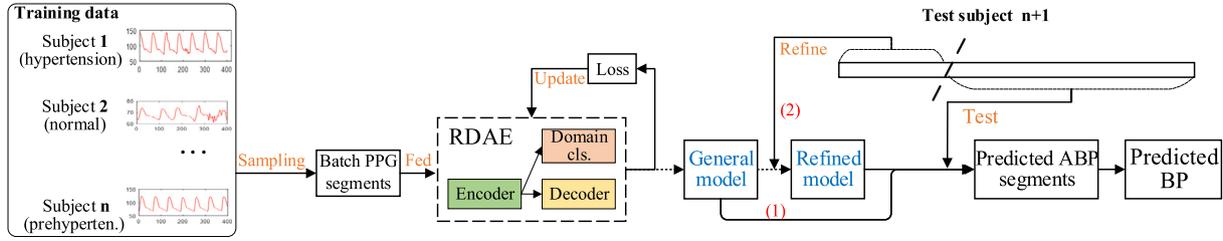
Fig. 1. The top-level diagram of blood pressure prediction based on RDAE. RDAE denotes the proposed regularized deep autoencoder model. Branch (1) indicates the general model after backpropagation-based training is directly evaluated on the test individuals; Branch (2) indicates the general model is firstly calibrated/refined using partial data of the test individual and then is evaluated on the remaining data.

2.2. Deep learning-based BP prediction

Deep learning, as a branch of ML, has attracted much attention in biomedical signal area because of its powerful ability in learning useful features automatically.

CNN based: Baek et al. [16] proposed a fully convolutional network by stacking the proposed Extraction–Concentration block for the time domain and frequency domain input, respectively. Slapnicar et al. [8] proposed a complex spectro-temporal deep network, where both PPG derivatives and frequency domain information of PPG were utilized. Eom et al. [19] proposed a model by stacking a VGG-style convolution model and Bi-GRU layer, where self-attention mechanism is utilized to quantify the importance of feature vectors along temporal direction. Schlesinger et al. [20] proposed CNN-based Siamese network for BP prediction.

RNN based: Tanveer et al. [21] proposed a model by stacking ANN and LSTM layer, and both the raw PPG and ECG signals were directly fed as input. Su et al. [18] proposed a model named DeepRNN by stacking multiple LSTM layer, and handcrafted features were fed as input. Fan et al. [22] proposed an multitask neural network, where PSO algorithm is used to search the optimal BP task weights. Zhang et al. [17] proposed a LSTM-based multitask regression network, and the thought of domain adversarial training was used to extract more general features between source and target domains, under domain adaption perspective. While, in our work, the multi-domain adversarial training is used to learn domain-invariant features among multi-domains that help converting PPG signal to ABP waveform.

BPs prediction is naturally modeled as a multi-task regression question in deep learning-based approaches. Therefore, differently from the traditional ML-based approaches where an independent predictive model is trained for each task (SBP, DBP or MBP prediction), only one model is needed here. Whereas, this inevitably introduced another question—the different loss scales among SBP, DBP and MBP will hinder the training of the model, which is seldom studied at present [22]. Although this study is based on deep learning, our goal is to predict the ABP waveform (predict BP becomes appendage to this task), instead of predict the BP values, thus avoiding the above problem. Besides, the above research usually use multiple different signals [16,18,19,21] or multi-channel signals [17] as input, while the proposed method use only PPG signal as input.

2.3. ABP waveform prediction

Currently, as far as we know, there are few studies that directly predict BP waveform. Ibtehaz et al. [15] firstly applies UNet [42], a classic network in the field of medical imaging, in conjunction with the concept of deep supervision, for generating ABP waveform from PPG signal. After that, Athaya et al. [43] did almost the same work as Ibtehaz et al. [15] based on UNet. Sadrawi et al. [44] applies two networks—LNet and UNet for building ABP waveform predictive model, respectively. However, none of the above mentioned studies
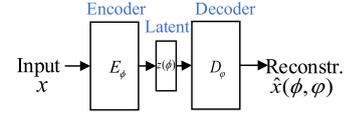
consider individual differences among data derived from different participants, we alleviated this problem by incorporating domain adversarial training. Besides, studies [15,43,44] directly use classical networks in other domains for their own study and no model complexity is analyzed. In contrast, by quantitatively analyzing the impact of several key parameters of the model on performance and model complexity (i.e #Param), several configuration schemes are given, which provide a flexible choice for model configuration in resource-constrained environment. Last, we presented two versions of the proposed method—RDAE and RDAE (with calibration).

3. Regularized deep autoencoder (RDAE)

In this section, we firstly give a formal description of signal conversion-based BP (waveform) prediction. Then, we briefly reviewed AutoEncoder, which is followed by the proposed RDAE. Last, we remarked the currently used evaluation procedure in this area.

3.1. Problem definition

Segment PPG signal and the synchronized ABP signal every T seconds, assume fs the sampling frequency. For ith sample (x_i, y_i), $x_i \in R^{1 \times d}$, $y_i \in R^{1 \times d}$, $d = T \cdot fs$, x_i denotes raw PPG segment, y_i denotes the counterpart raw ABP segment. Denotes $D = (X, Y)$ the dataset, $X \in R^{N \times d}$, $Y \in R^{N \times d}$, where N denotes the number of samples, d the length of a sample. We model BP prediction as a signal conversion question. Firstly, we need to find a mapping $f: x \rightarrow y$, f could be formulated as $f^* = \arg\min_f E_{(x,y) \sim p_{data}} l(f(x), y)$, where p_{data} denotes the prior joint distribution of X x Y, which is unknown. $l(,)$ denotes the loss function. After f^* is established, given test input x, predicted SBP, DBP and MBP could be computed as follows,

$$\hat{p}_{SBP} = \max f^*(x),$$
$$\hat{p}_{DBP} = \min f^*(x), \quad\quad (1)$$
$$\hat{p}_{MBP} = 2/3 \cdot \hat{p}_{DBP} + 1/3 \cdot \hat{p}_{SBP}$$

Since BP prediction is modeled as a signal conversion question, naturally, sequence to sequence model is selected. In the following, we firstly review autoencoder (AE) before introducing the proposed regularized convolution-based deep autoencoder (RDAE).

3.2. Autoencoder

The concept of 'autoencoder' (AE) was originally proposed by Hinton et al. [24], and autoencoder refers to a multilayer neural network

Fig. 2. A basic diagram of autoencoder comprised of encoder and decoder.
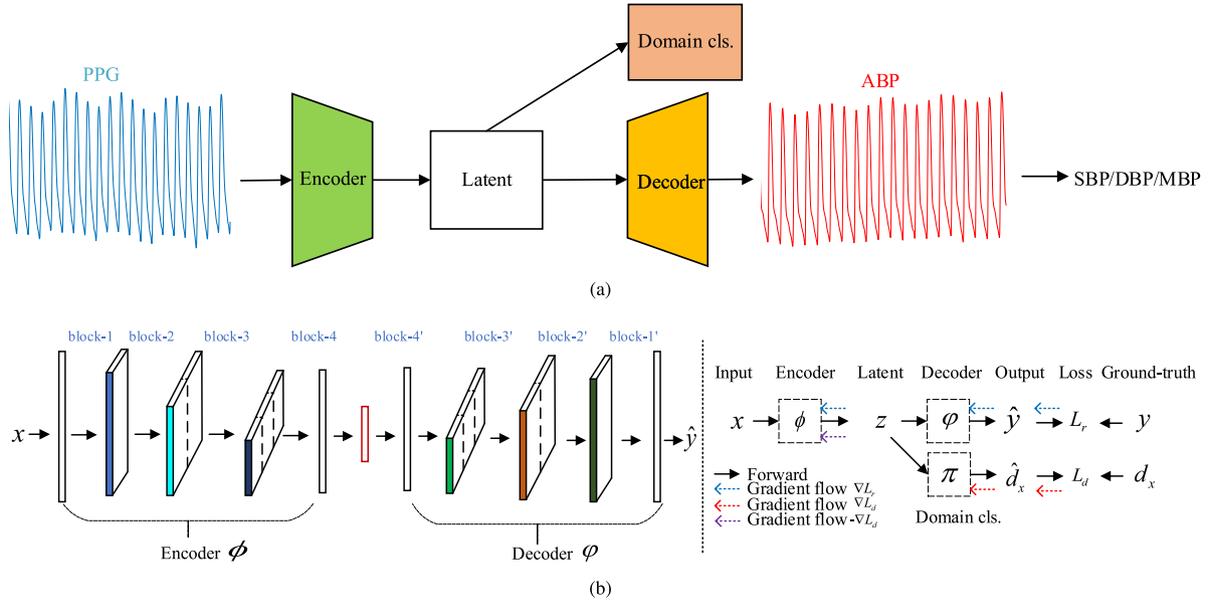
Fig. 3. Deep autoencoder with domain adversarial training (RDAE). (a) RDAE receives raw PPG signal as input, and generate the counterpart ABP waveform, RDAE is composed of three modules: encoder, decoder and domain classifier; (b) Left: the hierarchical network structure of RDAE, where encoder is composed by stacking 4 blocks (block-1, block-2, block-3, block-4), decoder is composed by stacking 4 blocks (block-4', block-3', block-2', block-1'). The domain classifier module is ignored; Right: Gradient flow in RDAE update, signal conversion loss—L_r is minimized w.r.t encoder and decoder during the backpropagation-based training. domain cls. module is updated by minimizing domain cls. loss—L_d, on the other hand, encoder is trained adversarially by maximizing domain cls. loss—L_d to learn domain-invariant features among multi-domains. x indicates PPG segment, $\hat{y}$ indicates predicted ABP segment.

with a low-dimensional central layer that was trained by reconstructing its input. As Fig. 2 shows, the network is comprised of two parts: Encoder and Decoder, which is parameterized with ϕ and φ, respectively. The Encoder—E_ϕ maps the input—x to a compact latent vector—$z(\phi)$, and the Decoder—D_φ tries to regenerate the input—$\hat{x}$ based on the latent vector—$z(\phi)$. The network is trained by minimizing reconstruction loss—$L_r(x,\hat{x})$, L_r could be mean square error (MSE) or mean absolute error (MAE). As an unsupervised learning framework, AE has been applied to various task, such as image reconstruction, clustering and machine translation etc.

3.3. RDAE

From representation learning aspect, the idea of AE is to learn a low dimensional, compact latent representation that adequately reconstruct the original input to a certain extent [45]. Here, we developed deep autoencoder for signal conversion-based BP (waveform) prediction, a supervised learning task. Concretely, ABP waveform is directly predicted, with PPG signal as input. Therefore, the objective is to learn a low dimensional, compact representation that could convert PPG signal to ABP waveform.

We developed convolution-based AE for the PPG-to-ABP conversion task. Furthermore, the pooling operation in the building block 'convolution+pooling' of classical CNN network [25,42,46] was replaced with stride convolution [47]. The architecture of RDAE is as Fig. 3 presents, Encoder and Decoder are comprised of three convolution blocks, respectively, and are connected by a 'bottleneck' multilayer perceptron. In practice, domain classifier is comprised of three fully-connected layer. The configuration details of RDAE is summarized in Table 1.

To make the learned latent representation more robust to different subjects (domains), the thought of domain adversarial neural network [31] inspired us. Concretely, apart from AE, an auxiliary domain classifier was connected to the latent representation and is trained to differentiate the belonging record of the latent representation of each raw PPG signal. On the other hand, in addition to minimizing the signal conversion error, the encoder part of AE is also updated adversarial to

weaken the distinguishing ability of the domain classifier. Therefore, after convergence, the Encoder could generate the latent representation as informative as possible (for the PPG-to-ABP signal conversion task), as well as domain-invariant. Thus, a smaller amount of data of test individuals is needed to fine-tune the general model.

Formally, assume E_ϕ the encoder parameterized by ϕ, D_φ the decoder parameterized by φ, D_π the domain classifier parameterized by π, the optimization objective of the proposed RDAE is as follows:

$$l(\phi,\varphi,\pi)=E_{(x,y)\sim p_{data}}L_r(D_\varphi(E_\phi(x)),y)-\lambda \cdot L_d(D_\pi(E_\phi(x)),d_x) \quad (2)$$

where the first term denotes the signal conversion loss, and L_r is computed with MAE, p_{data} denotes the unknown joint distribution of X x Y. The second term denotes the classification loss w.r.t the domain classifier, and L_d is computed with softmax loss, d_x denotes the one-hot encoding of the ground-truth domain label of sample—x. Hyperparameter λ trade off the weight of the two terms. Then, autoencoder and domain classifier are updated by optimizing the following two questions alternatively:

$$\hat{\phi}, \hat{\varphi} = \arg\min_{\phi,\varphi} l(\phi, \varphi, \hat{\pi}), \quad (3)$$

$$\hat{\pi} = \arg\max_{\pi} l(\hat{\phi}, \hat{\varphi}, \pi), \quad (4)$$

Gradient descent is used to solve (3) and (4), substitute Eq. (2) into (3) and (4), it is easily derived,

$$\pi \leftarrow \pi - \alpha\lambda \cdot \nabla_\pi L_d, \quad (5)$$

$$\varphi \leftarrow \varphi - \alpha \cdot \nabla_\varphi L_r, \quad (6)$$

$$\phi \leftarrow \phi - \alpha \cdot \nabla_\phi (L_r - \lambda \cdot L_d) \quad (7)$$

It is clearly parameters π and φ are updated regularly (i.e. along the negative gradient direction w.r.t loss). While, for parameter ϕ, it is updated along the gradient direction w.r.t the domain classifier loss L_d and along the negative gradient direction w.r.t the converted loss L_r, which implies the target: keep the latent representation (i.e. the output of Encoder) informative as well as domain-invariant.

Algorithm 1 RDAE

Input: D: data, $\{(x^i, y^i)\}$; E_ϕ: encoder; D_φ: decoder; D_π: domain classifier; B: batchsize; K: number of inner iterations; $maxEpoch$: maximum number of epoches; α: learning rate; λ: weight factor

Output: optimal model parameterized with φ and ϕ

1: $t = 0$;
2: initialize parameters ϕ, φ and π;
3: **repeat**
4:    $t \leftarrow t + 1$;
5:    **for** each $k \in [1, K]$ **do**
6:       sample $\{(x^i, y^i)\}_{i=1}^B$ from dataset D;
7:       compute latent code $\{z^i\}_{i=1}^B$: $z^i = E_\phi(x^i)$;
8:       compute output of domain classifier $\{\hat{d}_{x^i}\}_{i=1}^B$: $\hat{d}_{x^i} = D_\pi(z^i)$;
9:       compute gradient w.r.t π: $g_\pi \leftarrow \nabla \pi \frac{1}{B}\sum_{i=1}^B L_d(\hat{d}_{x^i}, d_{x^i})$;
10:       update domain classifier: $\pi \leftarrow \pi - \alpha \cdot g_\pi$;
11:    **end for**
12:    sample $\{(x^i, y^i)\}_{i=1}^B$ from dataset D;
13:    compute latent code $\{z^i\}_{i=1}^B$: $z^i = E_\phi(x^i)$;
14:    compute output of domain classifier $\{\hat{d}_{x^i}\}_{i=1}^B$: $\hat{d}_{x^i} = D_\pi(z^i)$;
15:    compute output of decoder $\{\hat{y}^i\}_{i=1}^B$: $\hat{y}^i = D_\varphi(z^i)$;
16:    compute gradient w.r.t ϕ: $g_\phi \leftarrow \nabla\phi\frac{1}{B}\sum_{i=1}^B L_r(\hat{y}^i, y^i) - \lambda \cdot L_d(\hat{d}_{x^i}, d_{x^i})$;
17:    update encoder: $\phi \leftarrow \phi - \alpha \cdot g_\phi$;
18:    compute gradient w.r.t φ: $g_\varphi \leftarrow \nabla\varphi\frac{1}{B}\sum_{i=1}^B L_r(\hat{y}^i, y^i)$;
19:    update decoder: $\varphi \leftarrow \varphi - \alpha \cdot g_\varphi$;
20: **until** converged or $t > maxEpoch$

3.4. Calibration

Calibration [14,16] is an additional & optional step of the proposed RDAE. Although the established health-care standards does not allow calibration of test device during evaluation process [14,30]. However, in actual application, individual's BP related data is measured and recorded continuously, providing a possibility of calibrating the general model using individual's historical data, as Fig. 1 shows. Note that in transfer learning, calibration (also known as fine-tuning) is a technique to adapt the learning model in the source domain to the target domain [48]. In this study, to validate the calibration step, for each test record, the first quarter data (8 samples) of the record is used to calibrate the general model. After calibration, the refined model is tested using the remaining data of the record.

3.5. What is a good evaluation procedure?

Currently, there are three evaluation procedures: (1) Leave-one-out [17]: Each record/subject is used to test the model trained on the remaining records; (2) Splitting at record level [14,16,34]: Train, validation and test set are split at record level based on a certain proportion; (3) Splitting at sample level [15,18]: Train, validation and test set are split at the final sample set. Generally, (1) could make the most of data to train the model and is suitable when the number of records is small or not very large; (2) is suitable when the number of records is large, whereas, individual differences are significant, especially in ICU patients, (2) could not adequately ensure the three subsets are independent identity distribution (I.I.D., an acquiescent assumption for evaluation in ML) [50]; (3) is at risk of data leakage, based on the consideration that the physiological signal of an individual is highly regular and will not change significantly in a short time, random division will cause there is sample similar to training set in test set. Thereby, the seemingly 'good' result does not objectively reflect the generalization ability of model actually. Generally, both (2) and (3) will cause biased results. An intuitive comparison of these procedures is best viewed in Fig. 4(b).

A good evaluate procedure should ensure the split sets are I.I.D, as well as avoid data leakage. Taking the significant differences of

(a)

(2) Split at sample-level
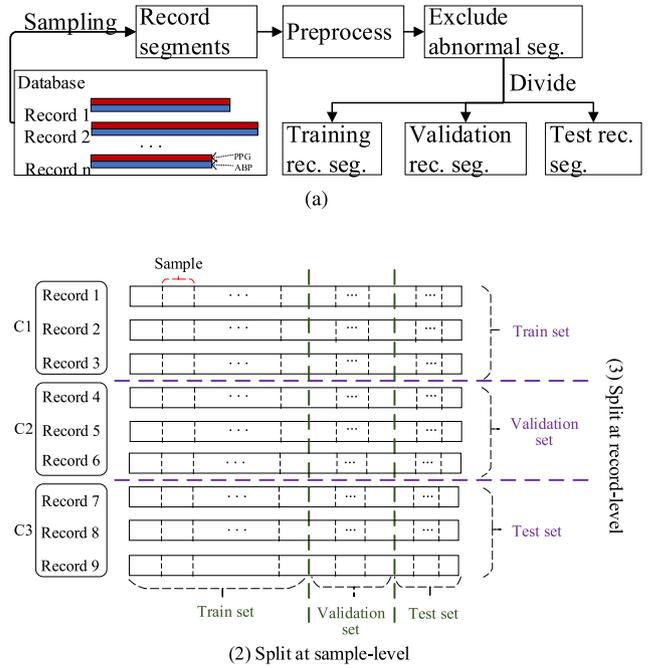
Fig. 4. Data preparation and evaluation procedure. (a) Data preparation process; (b) A demo visualizing the differences between evaluation procedures (2) and (3). Suppose there are nine records (records $1 \sim 3, 4 \sim 6, 7 \sim 9$ belong to class C1, C2 and C3, respectively). (2) will cause data of a record appears simultaneously in train., val. and test set, resulting in overestimation of the generalizability of model. (3) avoided this problem by splitting at record level, which may leads to another question—the BP distribution difference among train., val., and test set is too large (e.g. the final train, val. and test sets belong to C1, C2, C3, respectively). The proposed eval. procedure is similar to (3) and tackle the problem of which by splitting on each class of records instead of on the total records. (1) could be viewed as a special case of (3) that only a record contained in the test set in each evaluation.

blood pressure level between different individuals, we proposed a new division criterion for evaluation: Firstly, categorize records into three classes (Normal, Prehypertension, Hypertension) based on BP value. Secondly, divide records of each class into train, validation and test set based on a certain proportion (6:2:2). Finally, the train, validation and test set of each class are combined to form the final train, validation and test set, respectively. Here, what we want to emphasize is that different evaluation procedures lead to significant differences in the final results even for the same dataset and algorithms [8,16,51].

4. Experiment

4.1. Data prepare

The Cuff-less blood pressure estimation dataset [https://archive.ics.uci.edu/ml/datasets/Cuff-Less+Blood+Pressure+Estimation], which is originally derived from the MIMIC II database [52], is used. This version of data has been widely used in related studies [3,13,16,20,34,39], because of its ease of use. In UCI_BP, totally 12 000 subject records were hierarchically stored in four .mat files, where each record contains synchronized PPG, ECG and ABP signals with duration from 8 to 592 s. Fig. 4(a) presents the data preparation process. In this study, the record that with duration surpass 320 s were selected (3958 records reserved). Then, each record is divided into signal fragments (PPG signal and synchronized ABP signal) with length 10 s. Next, the final dataset is established, after several steps—*preprocess, exclude abnormal segments, divide* and *normalization* were executed, which is described as follows.
Preprocess
PPG fragment: Each PPG fragment is firstly preprocessed using 4-order bandpass filter with cut-off frequencies [0.5 Hz, 8 Hz], and then to

Table 1
Network configurations of RDAE.

Module	Block	Layer	Kernel[a]	Output[a]	Stride/Factor	Pad.	Activ.	Drop rate.
	—	Input	—	$(L,1)$	—	—	—	
	block-1	Conv1D	$(15, b)$	(L, b)	1/	same	ReLU	—
		Conv1D	$(15, b)$	$(L/5, b)$	5/	same	ReLU	—
	block-2	Conv1D	$(15, 2b)$	$(L/5, b)$	1/	same	ReLU	—
		Conv1D	$(15, 2b)$	$(L/25, 2b)$	5/	same	ReLU	—
		Dropout	—	—	—	—	—	0.1
	block-3	Conv1D	$(15, 4b)$	$(L/25, 4b)$	1/	same	ReLU	—
		Conv1D	$(15, 4b)$	$(L/50, 4b)$	2/	same	ReLU	—
		Dropout	—	—	—	—	—	0.1
Conv. AE	block-4	Flatten	—	L x 32/50	—	—	—	—
		Dropout	—	—	—	—	—	0.1
		Dense	—	32	—	—	ReLU	—
	—	Dense	—	*latent_dim*	—	—	None	—
	block-4'	Dense	—	32	—	—	ReLU	—
		Dense	—	$Lx32/50$	—	—	ReLU	—
		Dropout	—	—	—	—	—	0.1
		Reshape	—	$(L/50, 4b)$	—	—	—	—
	block-3'	Conv1D	$(15,4b)$	$(L/50, 4b)$	1/	same	ReLU	—
		UpSample	—	$(L/25, 4b)$	/2	—	—	—
		Dropout	—	—	—	—	—	0.1
	block-2'	Conv1D	$(15, 2b)$	$(L/25, 2b)$	1/	same	ReLU	—
		UpSample	—	$(L/5, 2b)$	/5	—	—	—
		Dropout	—	—	—	—	—	0.1
	block-1'	Conv1D	$(15, b)$	$(L/5, b)$	1/	same	ReLU	—
		UpSample	—	(L, b)	/5	—	—	—
		Conv1D	$(15, 1)$	$(L,1)$	1/	same	None	—
Domain cls.	—[b]	Dense	—	*latent_dim*	—	—	None	—
		BatchNorm	—	—	—	—	ReLU	—
		Dense	—	*latent_dim*	—	—	None	—
		BatchNorm	—	—	—	—	ReLU	—
		Dense	—	#records	—	—	None	—

[a](a, b), a denotes kernel size (in 'Kernel' column) or output dimension (in 'Output' column), and b denotes the number of channels.

[b]'—' denotes the corresponding attribute is not applicable to the layer, 'None' denotes linear activation.

Table 2
Data exclusion criteria. ABP segments satisfying one of the following conditions were excluded.

ABP signal	Features[a]						
	SQI [49]	SBP	DBP	len(SBP)	std(Δp_x)	std(ΔSBP)	ΔSBP
Abnormality condition	< 0	$\geq 180 \vee < 90$	$\geq 120 \vee < 60$	≤ 5	≥ 20	≥ 20	$\exists, > 10$

[a]len(SBP) denotes the number of cycles in a segment; std(Δp_x) denotes the standard deviation of all time intervals between adjacent cycles in a segment; std(ΔSBP) denotes the standard deviation of all peak values in a segment; ΔSBP denotes difference between any adjacent peak values in a segment.

correct the outliers, based on peak detection, the cycle-s with peak value out of the range $[\mu - 2\delta, \mu + 2\delta]$ is clipped proportionally to ensure $max(s) = \mu + 2\delta$ (when s contains crest), $min(s) = \mu - 2\delta$ (when s contains trough), where μ and δ denote mean and standard deviation of peak values or trough values in the fragment. Fig. 5(a) presents two cases.

ABP fragment: Each ABP fragment is firstly preprocessed with Savitzky–Golay filter [33], and then is split into multiple cycles based on peak detection. Last, SBP and DBP are computed as average value of minimum values and maximum values of these cycles, MBP is computed as MBP=(2 · DBP+SBP)/3. Fig. 5(b) presents two cases.

Exclude abnormal segments: Because MIMIC is a critical care medicine dataset, there are severe noise contained in the signal. For each PPG fragment and the synchronized ABP signal, if one condition in Table 2 is satisfied, it was excluded, and Fig. 5(c) presents several excluded signal fragments. It is worth mentioning that the exclusion criterion is mainly for ABP signal, because we want to reserve as much variations among PPG segments as possible, in an attempt to improve the generalizability of model.

Divide: After abnormal segments were excluded, there are 1227 records reserved that within each contains up to 32 samples. We firstly divide

Table 3
Categorization rules. Records are categorized into three classes [51] based on the range of BP values.

Sym.	Category	Condition	#Record
C1	Normal	SBP < 90 or DBP < 60 or (SBP $\in$ [90,120) and DBP $\in$ [60,80))	317
C2	Prehypertension	SBP $\in$ [120,140) and DBP $\in$ [80,90)	613
C3	Hypertension	SBP $\geq$ 140 or DBP $\geq$ 90	297

these records into three categories according Table 3 [51]. Then, train, validation and test records were divided from each category according 6:2:2. Last, after combine subsets from the three categories, the final dataset is acquired, with #train, #validation and #test records 739, 244, 244, respectively. We declare that all samples of each record only appear in train set or validation set or test set. Each record contains 32 samples, summing up totally 39 264 samples, and Fig. 6 presents the distribution of BP values of these samples. The sampling frequency is 125 Hz, each sample contains a preprocessed raw PPG signal with 625 points, the corresponding raw ABP signal and BP values (include SBP, DBP and MBP).
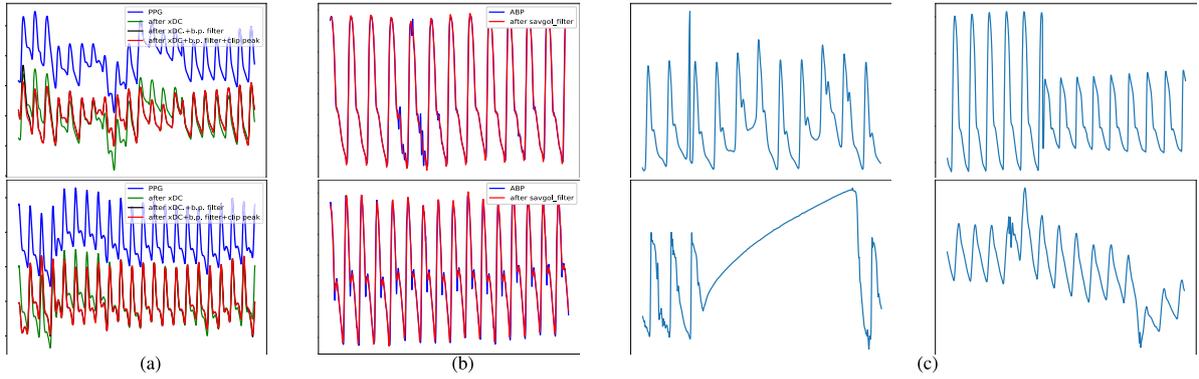
Fig. 5. Preprocess results and excluded abnormal segments. (a) two cases of PPG segments; (b) two cases of ABP segments; (c) four cases of excluded abnormal ABP segments.

Normalization: To remove the differences of PPG signals among different individuals, the raw signal has to be normalized [17,22,33,43]. Z-Score normalizer is used here. Concretely, first, fitting a normalizer on PPG segments of the training records, denote the fitted normalizer as $(\mu_{train}^{PPG}, \delta_{train}^{2,PPG})$. Then, apply the normalizer to each PPG segments of training, validation and test records, respectively, i.e.

$$x_{norm} = (x - \mu_{train}^{PPG})/\delta_{train}^{PPG} \qquad (8)$$

where x denotes any PPG segment, and x_{norm} the corresponding normalized segment. ABP signal is normalized in a similar way. Additionally, in the inference/test stage, the output of the model—$\hat{y}$ has to be reprojected to the original space through reversed normalization as follows,

$$\hat{y} = \hat{y} \cdot \delta_{train}^{ABP} + \mu_{train}^{ABP} \qquad (9)$$

where $(\mu_{train}^{ABP}, \delta_{train}^{2,ABP})$ denotes the normalizer fitted on ABP segments of the training records.

4.2. Implementation details

Weights of all model layers were initialized based on 'glorot-uniform' initializer, *latent_dim* is set to 5 by default. To improve efficiency, K (ref Algorithm 1) is set to 1. Batchsize B equals 128, and each batch is comprised of samples of 4 records. Hyperparameter—λ is set to 0.01 in default. Adam optimizer is used to solve model, learning rate is initially set to 0.0005 and is decayed by 0.01 every 550 batches. The maximum training epochs is set to 100. Additionally, for the calibration step, considering that there are only 8 samples, stochastic gradient descent is used to optimize the general model, and the training epochs is set to 20. All methods were implemented based on Python, and DL methods were implemented based on tensorflow framework with version 2.1.0 and all experiments were executed on NVIDIA RTX2080Ti with 11 GiB memory. The code will be released on Github soon.

4.3. Evaluation metrics

To evaluate the performance of the proposed method, three classical metrics [10,14,16], i.e. mean absolute error (MAE), mean error (ME) and standard deviation (SD), were used, which is formulated as below,

$$MAE = \frac{1}{N} \sum_{i=1}^{N} |p_i - \hat{p}_i|, \qquad (10)$$

$$ME = \frac{1}{N} \sum_{i=1}^{N} p_i - \hat{p}_i, \qquad (11)$$

$$SD = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (p_i - \hat{p}_i)^2} \qquad (12)$$

where N denotes the number of test segments (samples), p_i and $\hat{p}_i$ denote the ground-truth BP value and predicted BP value, respectively.

4.4. Evaluation procedure

The dataset is divided into training records, validation records, and test records according to the method described in Section 4.1. The model is trained iteratively on the training records until up to the maximum #epochs, and the performance of the model is dynamically monitored and evaluated on the validation set after each epoch. Then, the model with the lowest MAE on validation set is selected as the final prediction model. Last, the predictive model is evaluated on the test records, and experiments were repeated ten times by generating ten different random splitting of records to get reliable results.

5. Results

5.1. Predicted waveform analysis

Fig. 7 presents the conversion results of RDAE on five waveforms with different characteristics before (Fig. 7(a)) and after (Fig. 7(b)) calibration. The first picture (pic.) represents the case with clear dicrotic notch. The second pic. represents the case with-no clear dicrotic notch. The third pic. represents the case with Hypotension. The fourth pic. represents the case with obviously abnormal cycle exists. The fifth pic. represents the case with severe fluctuation in blood pressure. It can be seen that the converted ABP seg. (i.e. ABP*pred*.) fitted the ground-truth ABP seg. (i.e. ABP$_{gt}$) well with cycle consistency in each heartbeat, especially at the position of beginning, main crest and dicrotic notch peak in each cycle. Furthermore, the conversion result of RDAE shows a certain degree of robustness on abnormal seg. as the last two columns in Fig. 7 shows, which, we argue, is owned to the following two points: (*i*) in the data preparation stage, the exclusion criterion of abnormal seg. is mainly for ABP seg. Therefore, the variations among PPG segments were preserved as much as possible; (*ii*) in the training process, in addition to minimizing the conversion error, RDAE attempts to minimize the distinction among latent representations of different domains. Last, after additional calibration step to RDAE, the quality of converted ABP seg. has a certain degree of improvement. On the other hand, there is a significant difference between the predicted and ground-truth waveforms both in the case of hypotension and the case with severe BP fluctuation, and the possible cause includes: (*i*) it is possible that some PPG segments of two different individuals are very similar, but the corresponding ABP waveforms are very different [8]. Then, if one record is included in the training set and another record is included in the test set, the predictive result will be very poor; (*ii*) there are few wave pairs with certain characteristic (such as large fluctuation of blood pressure in a short time) in the training set, which cannot be fully used for training well.
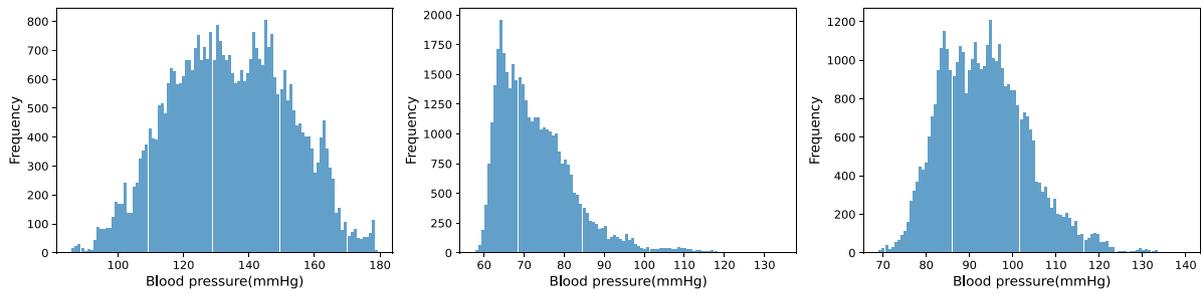
Fig. 6. Distribution of blood pressure. Left: SBP; Middle: DBP; Right: MBP.
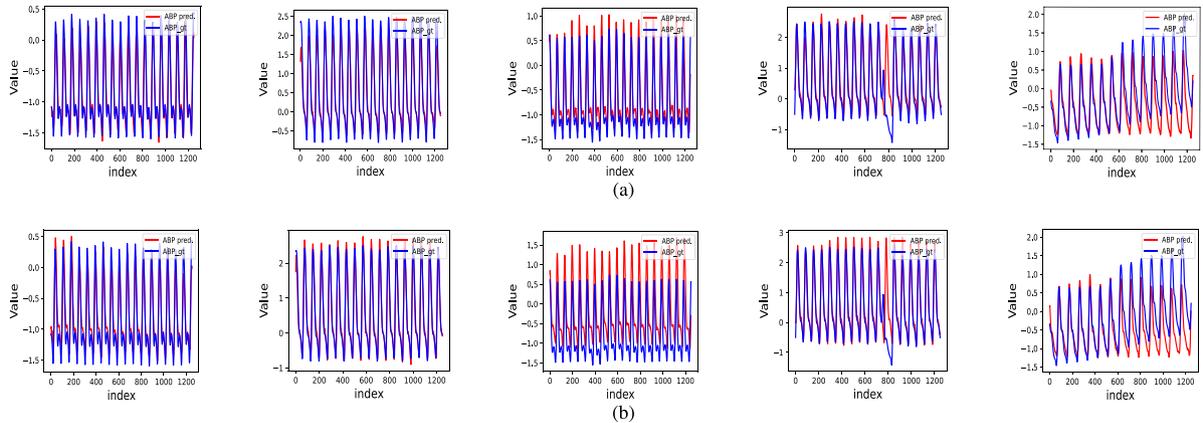
Fig. 7. Comparison results between predicted ABP waveform (on the test set) and ground-truth ABP waveform based on the model—RDAE. (a) with calibration; (b) no calibration. Each row contains 5 cases (from left to right): (*i*) with clear dicrotic notch, (*ii*) with-no clear dicrotic notch, (*iii*) Hypotension, (*iv*) with obviously abnormal cycle, (*v*) with greatly fluctuated blood pressure.

5.2. Ablation study

To validate the effectiveness of each component of RDAE, we designed the following five approaches:

- DAE (pool): denotes all the stride-convolution in DAE is replaced with conventional pooling operation.
- DAE: denotes RDAE with-no domain adversarial training.
- RDAE: denotes the proposed methods withno calibration.
- DAE (with calibration): denotes DAE with additional calibration.
- RDAE (with calibration): denotes RDAE with additional calibration (i.e. partial data of target domain is used to retrain the pretrained (general) model.

Fig. 8 presents the loss & performance (MAE) variation in the training process. In Fig. 8(a), it can be seen that the train loss, validation loss and test loss of RDAE drops quickly in the first 20 epochs and then gradually converge until it stabilizes after 100-th epochs. In addition, the variation of test loss (i.e. MAE) w.r.t the three blood pressure prediction tasks (i.e. SBP, DBP and MBP) in the training process were visualized in Fig. 8(b). Since SBP and DBP correspond to the peak and valley values of ABP waveform, respectively, the quality of converted ABP seg. determines the accuracy of blood pressure prediction to a certain extent. As Fig. 8(b) shows, although not directly regressing blood pressure as training targets, owning to the gradually improved quality of converted ABP seg., the metric—MAE w.r.t the three tasks drops consistently, of which the trend was similar to that of loss in Fig. 7(a) in the training process.

Table 4 presents the numerical comparison results of the mentioned five approaches, which use exactly the same configuration (the additional parameter—λ in RDAE is set to 0.01) and are trained and tested using the identical train set and test set, respectively. It is clearly that the prediction performance (i.e. MAE, the smaller, the better) improves steadily from top to down. Especially, the comparison of train loss

and test loss among RDAE, DAE, DAE (pool) is presented in Fig. 8(c), it can be seen that the conversion loss of DAE drops quickly than that of DAE (pool), the conversion loss of RDAE drops slightly quick than DAE, although not particularly obvious due to large loss range. In particular, the conversion loss is only a term in the loss function of RDAE. Therefore, we draw conclusions: *i*) utilization of the fully-convolution block, instead of the regular "convolution+pooling" block, in building DAE improved the performance (DAE vs. DAE (pool)); (*ii*) inclusion of domain adversarial training helps RDAE to get better generalization ability than DAE (RDAE vs. DAE), and allows the general model better adapt to new individuals when using the same number of target samples for fine-tuning (RDAE (with calibration) vs. DAE (with calibration)). Especially note that the inclusion of domain adversarial training strategy makes the standard deviation of the three metrics (i.e. MAE, ME, STD) get smaller, which proven the effect of domain adversarial training in improving model's robustness; (*iii*) additional calibration (i.e. personalization [16]) step further improves the performance of RDAE, especially in SBP prediction (RDAE (with calibration) vs. RDAE).

5.3. Comparison with the AAMI standard

Table 5 presents the comparison results of RDAE with the AAMI standard [53]. The test device satisfying the AAMI standard if its precision must not differ from the mercury standard by a mean error of ≤ 5 mmHg or a standard deviation of ≤ 8 mmHg. According to the AAMI standard, almost all of the proposed RDAE with or without calibration satisfy the AAMI standard except the SBP prediction of RDAE, on which the metric—ME ≤ 5 mmHg, while the metric—STD slightly bigger than 8 mmHg.
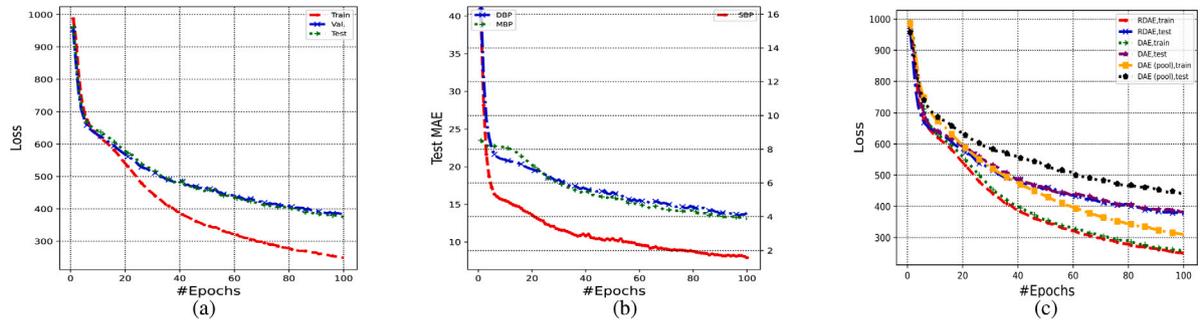
Fig. 8. Loss/Performance curve variation in the training process, with the maximum #Epochs is set to 100, *b* equals 32. (a) train/val./test loss of RDAE; (b) test performance (i.e. MAE) of RDAE w.r.t the SBP, DBP and MBP prediction; (c) train/test loss comparisons among RDAE, DAE and DAE (pool).
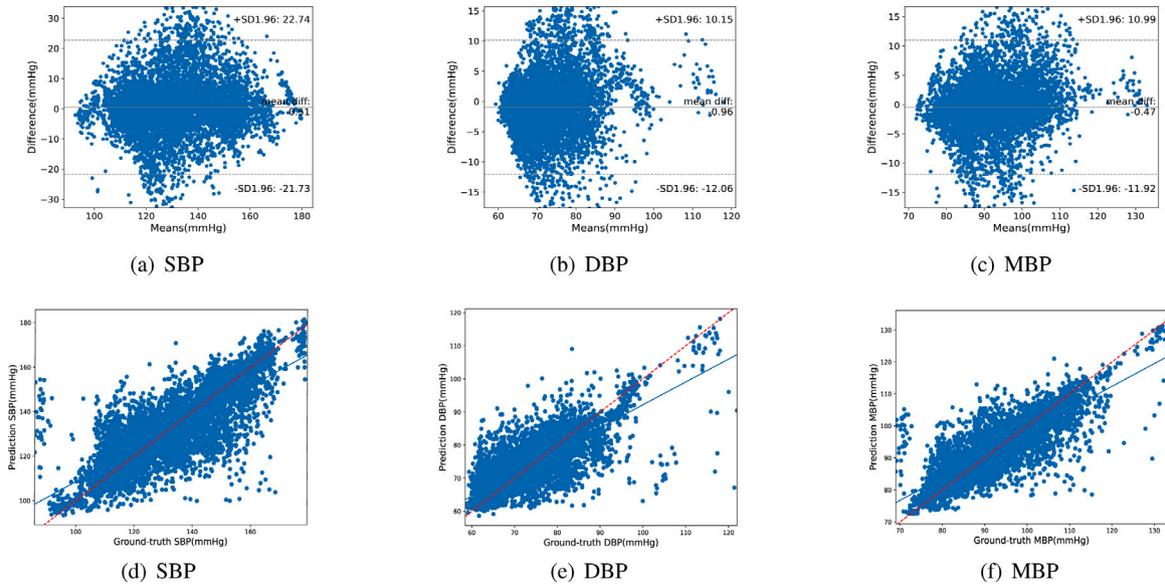
Fig. 9. Correlation and Bland–Altman plots of the proposed method—RDAE (no calibration) on SBP, DBP and MBP predictions. (a) ~ (c) Bland–Altman plots on SBP, DBP and MBP prediction, respectively; (d) ~ (f) Correlation plots on SBP, DBP and MBP prediction, respectively. The blue solid line and the red dotted line indicate the fitted line and the reference line, respectively.
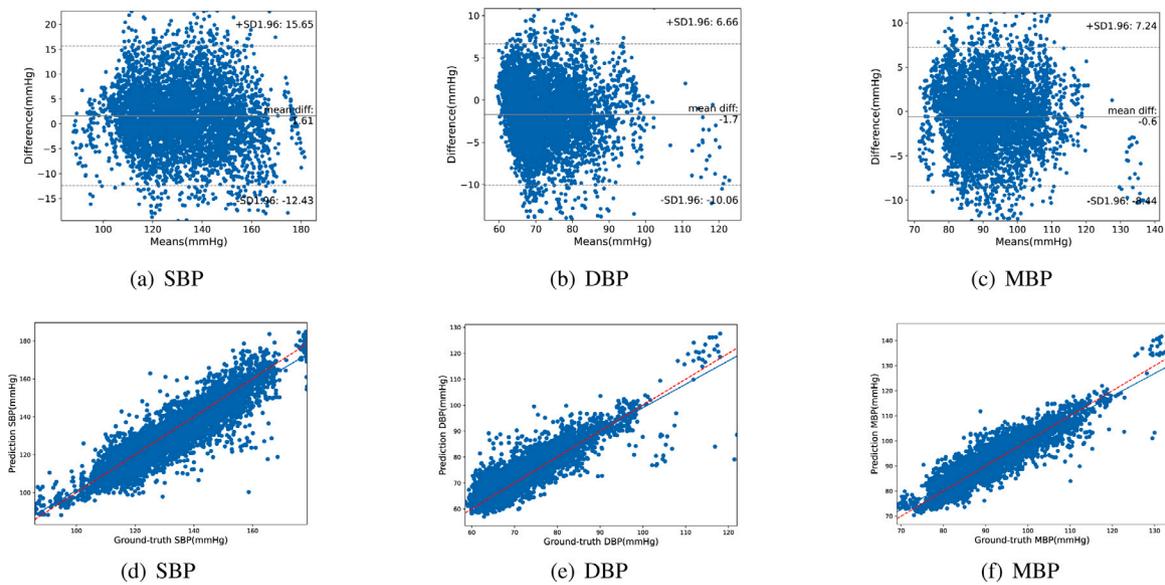
Fig. 10. Correlation and Bland–Altman plots of the proposed method—RDAE (with calibration) on SBP, DBP and MBP predictions. (a) ~ (c) Bland–Altman plots on SBP, DBP and MBP prediction, respectively; (d) ~ (f) Correlation plots on SBP, DBP and MBP prediction, respectively. The blue solid line and the red dotted line indicate the fitted line and the reference line, respectively.

Table 4

Results of the proposed methods—RDAE and RDAE (with calibration). The result is presented as the average value, with standard errors in parentheses.

Method	Task	Metrics (unit: mmHg)		
		MAE	ME	STD
DAE (pool)	SBP	9.568(0.801)	2.090(2.469)	12.278(0.871)
	DBP	4.865(0.234)	−0.581(0.807)	6.477(0.342)
	MBP	4.699(0.170)	0.309(1.150)	6.204(0.218)
DAE	SBP	8.082(0.538)	1.024(1.398)	10.559(0.682)
	DBP	4.179(0.485)	−0.322(0.858)	5.569(0.574)
	MBP	3.903(0.315)	0.126(0.795)	5.210(0.390)
RDAE	SBP	7.945(0.389)	1.447(0.877)	10.372(0.532)
	DBP	4.114(0.237)	−0.417(0.641)	5.504(0.396)
	MBP	3.834(0.242)	0.204(0.436)	5.130(0.349)
DAE (with calibration)	SBP	5.734(0.265)	1.686(0.431)	7.009(0.335)
	DBP	3.175(0.255)	−1.328(0.256)	3.787(0.272)
	MBP	2.979(0.147)	−0.323(0.257)	3.522(0.168)
RDAE (with calibration)	SBP	5.424(0.164)	1.648(0.319)	6.640(0.190)
	DBP	3.144(0.162)	−1.280(0.208)	3.740(0.172)
	MBP	2.885(0.147)	−0.304(0.181)	3.412(0.173)

Table 5

Evaluation of RDAE and RDAE (with calibration) with the AAMI standard.

Method/ Standard	Task	Metrics (unit: mmHg)		#Subject (test)	Pass
		ME	STD		
RDAE	SBP	1.447(0.877)	10.372(0.532)		No
	DBP	−0.417(0.641)	5.504(0.396)	244	Yes
	MBP	0.204(0.436)	5.130(0.349)		Yes
RDAE (with calibration)	SBP	1.648(0.319)	6.640(0.190)		Yes
	DBP	−1.280(0.208)	3.740(0.172)	224	Yes
	MBP	−0.304(0.181)	3.412(0.173)		Yes
AAMI	—	≤ 5	≤ 8	≥ 85	—

5.4. Comparison with the BHS standard

Table 6 presents the comparison results of RDAE with BHS standard [54]. The test device achieves Grade A, Grade B or Grade C if the corresponding condition is satisfied. The criterion of a device full filling the BHS standard is that it must achieve at least Grade B for SBP and DBP prediction. According to the BHS grading criterion, the no calibration version of RDAE achieves Grade C, Grade A and Grade A in the SBP, DBP and MBP prediction task, respectively. After calibration, the calibration version of RDAE achieves Grade B, Grade A and Grade A in the SBP, DBP and MBP prediction task, respectively. In conclusion, the proposed RDAE satisfies the BHS standard except the no calibration version of RDAE in SBP prediction.

To visualize the fitness of predicted BP values to ground-truth BP values, Fig. 9 presents the Bland–Altman plot and Correlation plot of uncalibrated RDAE in SBP, DBP and MBP prediction. As Fig. 9(a)~(c) shows, the Bland–Altman plot shows that most of the estimated points for SBP, DBP and MBP prediction were within 0.51 ± 22.74,

0.96 ± 12.06, −0.47 ± 11.92 limits. Therefore, BP estimated by the uncalibrated RDAE has the ability of approximating ground-truth BP. After calibration, the limits further narrowed to 1.61 ± 15.65, −1.7 ± 10.06, −0.6 ± 8.44 for SBP, DBP and MBP prediction, respectively, as Fig. 10(a)~(c) shows. As for Correlation plot, Fig. 9(d)~(f) clearly indicates that there is a strong linear relationship between the estimated BP and the ground-truth BP for the three prediction tasks. While, the fitted line (blue solid line) has a certain angle difference to the reference line $y = x + b$ (red dotted line). Concretely, for all the three tasks, the BP of the samples with relatively low blood pressure was overestimated, and the BP of the samples with relatively high blood pressure was underestimated. After calibration, this linear relationship is closer to the reference line. As Fig. 10(d)~(f) shows, for all the three tasks, the angle between the fitted line and the reference line is significantly reduced, which means the estimated error is further reduced.

5.5. Analysis of latent representation

To further analyze the effect of domain adversarial training on the latent representation, we try to visualize the latent representation of training set and test set based on RDAE and DAE, respectively. Fig. 11 presents the result. It can be seen that the latent representation of train set and test set based on RDAE seems more diverse and disorder, and with more irregular boundaries than that based on DAE. Because of the large number of domains (records), it is hard for the eyes to distinguish this subtle difference. Therefore, the optimization objective of clustering [55] is employed to quantify this difference. Concretely, the domain label of each latent representation is served as its belonging cluster index, then the objective is computed as follows,

$$dist = \frac{1}{N} \sum_{i=1}^{N_d} \sum_{x \in d_i} \| x - \mu_{d_i} \|^2 \tag{13}$$

where $N = \sum_{i=1}^{N_d} |d_i|$, $\mu_{d_i} = \frac{1}{|d_i|} \sum_{x \in d_i} x, N_d$, denotes the number of domains (records), d_i denotes the set composed of the t-SNE [56] mapping of the latent representations in ith domain. As expected, on the train set, $dist_{RDAE} = 1270.058 > dist_{DAE} = 1161.077$, on the test set, $dist_{RDAE} = 1331.232 > dist_{DAE} = 1020.656$, which reflects that RDAE has the ability of confusing the latent represents of different domains to a certain extent, in other words, makes the learned latent representations more general among different domains.

5.6. Effect of hyperparameter

Hyperparameter λ controls the weight of domain classifier loss, to investigate its effect on the performance (MAE) of RDAE, RDAE is tested with different λ values—0.001, 0.01, 0.1, 1, 5, 10, respectively. All experiments are tested under the same configuration (except λ value) and use the identical training, validation and test sets. As the value of λ keeps increasing, the performance is gradually improved. Whereas, there are increasing possibility of cases that the loss does not converge, in the training process, resulting in a higher mean MAE with large standard deviation, as Fig. 12 shows. By trading off stability and performance, λ is set to 0.01 in all experiment.

Table 6

Evaluation of RDAE and RDAE (with calibration) with the BHS standard.

Method/ Standard	Task	Proportion of the subjects with MAE satisfying:			Grade
		≤5 mmHg	≤10 mmHg	≤15 mmHg	
RDAE	SBP	46.3%(1.9%)	72.1%(2.0%)	85.2%(1.4%)	C
	DBP	73.2%(1.8%)	91.9%(0.9%)	97.0%(0.8%)	A
	MBP	76.0%(1.8%)	92.3%(1.0%)	96.9%(0.7%)	A
RDAE (with calibration)	SBP	58.5%(1.7%)	85.6%(0.8%)	95.0%(0.5%)	B
	DBP	81.5%(1.2%)	96.4%(0.7%)	99.0%(0.3%)	A
	MBP	83.6%(1.5%)	97.5%(0.8%)	99.6%(0.2%)	A
BHS	—	60%	85%	95%	Grade A
		50%	75%	90%	Grade B
		40%	65%	85%	Grade C
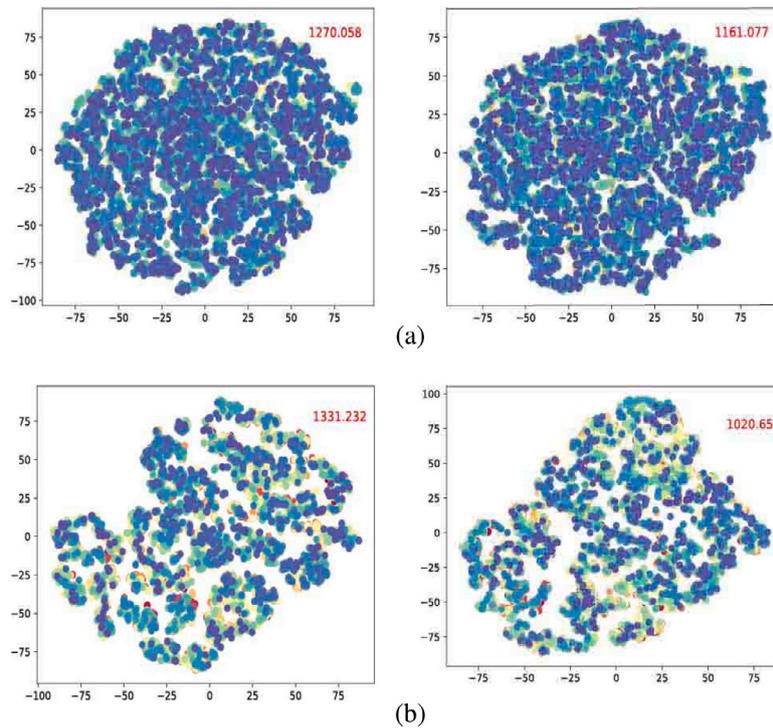
Fig. 11. Comparison of latent representations based on t-SNE [56]. (a) Latent representations of RDAE (left) vs. DAE (right) on training set; (b) Latent representations of RDAE (left) vs. DAE (right) on test set.
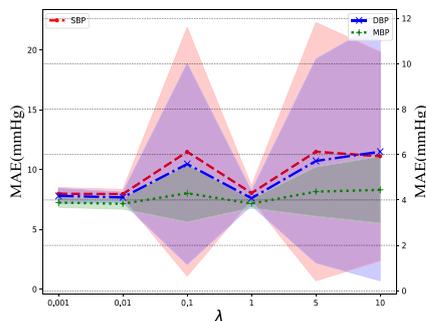
Fig. 12. Effect of hyperparameters—λ to performance (MAE).

Table 7
Results of RDAE with different configuration of b.

Method	Config.(b)	MAE (unit: mmHg)			#Param.
		SBP	DBP	MBP	
RDAE	8	10.877(1.068)	5.645(0.353)	5.324(0.391)	0.42M
	16	8.844(0.439)	4.669(0.392)	4.347(0.316)	1.63M
	32(default)	7.945(0.389)	4.114(0.237)	3.834(0.242)	6.49M
	64	7.700(0.537)	3.930(0.376)	3.707(0.218)	25.91M

5.7. RDAE with different configurations

In the designed convolution-based RDAE, the number of convolution kernels and the dimensions of latent code representation (i.e. the width of the 'bottleneck' layer of MLP) are two key parameters that affect the performance. We experimentally verified the effect of these parameters on model's accuracy and complexity, which provide a flexible choices for practical model configuration on resource-limited hardware [57].

Table 7 presents the comparison results of RDAE with different base (b) of the number of convolution kernels. With the increase of b, the

Table 8
Results of RDAE with different configuration of $latent\_dim$.

Method	Config. ($latent\_dim$)	MAE (unit: mmHg)			#Param.
		SBP	DBP	MBP	
RDAE	3	10.050(0.619)	5.166(0.367)	4.683(0.283)	6.488M
	5(default)	7.945(0.389)	4.114(0.237)	3.834(0.242)	6.490M
	7	7.087(0.438)	3.881(0.336)	3.593(0.246)	6.492M
	9	6.664(0.413)	3.662(0.338)	3.484(0.262)	6.494M
	11	6.479(0.561)	3.512(0.226)	3.394(0.222)	6.496M
	13	6.483(0.397)	3.509(0.287)	3.400(0.237)	6.498M

performance of RDAE to predict SBP, DBP and MBP improved gradually, while, the number of parameters required increases exponentially. Especially, when b surpasses 32, the improvement in performance compared to the increased storage overhead is almost negligible. By trading off the performance and model size (complexity), b is set to 32 in all the experiments.

Table 8 presents the comparison results of RDAE with different $latent\_dim$ values. It revealed that with the increase of $latent\_dim$, the performance is gradually improved with very little increase in the number of parameters. $latent\_dim$ is set to 5 in all the experiments. In conclusion, the effect of $latent\_dim$ to performance is more significant than that of b to performance, our suggestion is: parameter b should not be too large (32 seems the best trade-off) when resource is limited, parameters b and $latent\_dim$ could be adjusted larger in resource sufficient platform.

5.8. Comparison with other systems

Table 9 presents the comparison results of the proposed RDAE (including calibrated version and uncalibrated version) with other methods. Since a single metric—ME cannot evaluate the performance properly, and metric—STD is similar to MAE to a certain extent except is more sensitive to abnormal values. Therefore, metric—MAE is used to evaluate the BP prediction performance of these approaches. In

Table 9

Comparison of RDAE and RDAE (with calibration) with other systems.

Split criterion	Cal.	Method	Dataset	#Record used	#Sample	Signals	Input	MAE (mmHg)			#Param.
								SBP	DBP	MBP	
split at record level	No	RDAE (proposed)	MIMIC II	1227	39264	PPG, ABP	Raw signal	7.945	4.114	3.834	6.490M
		Baek [16]	MIMIC II	1912	1912	PPG,ECG,ABP	Raw signal	9.30	5.12	–	17.596M
		Kachuee [14]	MIMIC II	3663	3663	PPG,ECG,ABP	Features	11.17	5.35	5.92	–
		Thambiraj [34]	MIMIC II	3801	3801	PPG,ECG,ABP	Features	9.00	5.48	3.2	–
		Eom [19]	Private	15	Unknown	PPG,ECG,BCG	Raw signal	9.70	5.79	–	10.648M
		Slapnicar [8]	MIMIC III	510	700h	PPG,ECG,ABP	Raw signal	15.41	12.38	–	3.523M
	Yes	RDAE (proposed)	MIMIC II	1227	39264	PPG, ABP	Raw signal	5.424	3.144	2.885	6.490M
		Baek [16]	MIMIC II	1912	1912	PPG,ECG,ABP	Raw signal	5.32	3.38	–	17.596M
		Kachuee [14]	MIMIC II	3663	3663	PPG,ECG,ABP	Features	8.21	4.31	–	–
		Zhang [17][a]	Private	11	Unknown	PPG,ABP	Raw signal	6.79[a]	4.48[a]	–	0.045M
		Eom [19]	Private	15	Unknown	PPG, ECG, BCG	Raw signal	4.06	3.33	–	10.648M
		Slapnicar [8]	MIMIC III	510	700h	PPG,ECG,ABP	Raw signal	9.43	6.88	–	3.523M
split at sample level[b]	—	Mousavi [30]	MIMIC II	1323	1323	PPG,ECG,ABP	Whole based feature	3.97	2.43	2.61	–
		Su [18]	Private, healthy records	84	10 min /record	PPG,ECG,ABP	Features	3.73	2.43	–	2.790M
		Ibtehaz [15]	MIMIC II	unknown	127260	PPG, ABP	Raw signal	5.73	3.45	2.31	39.967M
		Sharifi [3]	MIMIC II	3663	Unknown	PPG,ECG,ABP	Raw signal	7.83[c]	4.86[c]	3.63[c]	–

[a]Denotes Root mean square error (RMSE).

[b]Note that there is a risk of data leakage when splitting train/val./test sets at sample level, resulting in abnormally high prediction accuracy, because data of one record will appears simultaneous in train set and test set.

[c]Note that the author uses a different method of training set and test set partition from all other studies, [3] was categorized into *'split at sample level'* because data of a record always appeared simultaneously in training set and test set.

addition, the split criterion and whether or not has calibration are key important factors that relates to the performance [8,16]. Specifically, *split at record level* will result in exceptionally high accuracy, because this splitting criterion causes highly similar samples to appear in both the training set and test set, although there are few studies have clearly pointed out this problem. For fairness, these approaches (with different versions) are first categorized and then are compared.

From Table 9, the performance of the approaches with *split at record level* seems inferior to the approaches with *split at sample level*, whereas, this does not objectively reflect the generalization ability of a model, because the latter is at the risk of data leakage in the evaluation process. For the uncalibrated version, all approaches perform relatively poorly in predicting SBP. Nevertheless, the proposed method—RDAE can still reach 7.945 mmHg MAE in SBP prediction, which is obviously better than other approaches. After calibration, the mean absolute error of RDAE in SBP, DBP and MBP prediction improve 2.521 mmHg, 0.970 mmHg, 0.949 mmHg, respectively. Although the result of Baek et al. [16] method is slightly better than our calibrated RDAE in SBP prediction, however, they use half of test samples to calibrate the model, while we use only a quarter of test samples (8 samples) to calibrate the general model. Specifically, as for DL methods, the model size of RDAE is slightly larger than Slapnicar et al. [8] method, and far less than Baek et al. [16] method, Eom et al. [19] method and Ibtehaz et al. [15] method. In addition, in general, for the studies [17–19] based on self-collected data from healthy individuals or outpatients, of which the result is better than that based on intensive care patients, a simple prediction algorithm [37], or DL method with small size [17] is enough. In conclusion, both the uncalibrated version and calibrated version of RDAE is competitive to other methods in BP prediction accuracy and model size, although the accurate BP prediction is only a by-product of high-quality converted ABP signal.

5.9. Discussion on system performance

Although the proposed method has achieved competitive results compared to other studies that using the same data source, the performance of the method, as revealed in Sections 5.3 and 5.4 , is still some distance (in SBP prediction) from the highest level of the BHS & AAMI standards. Note that the dataset used is collected from ICU patients, where the health status of the participants were more complex/diversified—suffer from adverse events (bleeding, in surgery,

etc.), resulted in more fluctuations in physiological state. Therefore, the data itself is more challenging than that of healthy individuals and outpatients. Concretely, (1) PPG and ABP waveforms of an individual corrupted by noise more severe and fluctuated more during the data collection period; (2) the differences among different individuals are more diverse and significant; Term (1) means the unknown relation between PPG and ABP signals is more complex. Term (2) means the general trained model has to be calibrated/fine-tuned to adapt to the test individual. As revealed by experiments, firstly, multi-domain adversarial training helps improving the predictive performance through learning domain-invariant features among multiple training domains/individuals. Secondly, after calibrating the general model using 80 s data of a test individual, the performance is improved significantly.

6. Interpretation of the method for PPG signal to ABP waveform conversion

Since the PPG and ABP signals have the same incentive source—heart [7], when blood flows from the brachial artery to the digital artery, ABP and PPG is measured from the brachial artery and digital artery, respectively [7,43]. Therefore, there exist a strong interconnection between the obtained PPG and ABP signal. Martínez et al. [7] analyzed the similarities in time and frequency domains between PPG and ABP signals, and the experimental results reveal that there has a high morphology correlation between PPG and ABP waveforms for normotensive, prehypertensive and hypertensive individuals. Hubner et al. [23] clinically investigated the potential of PPG in detecting a spontaneous pulse from the finger and other sensory organs, and the analysis based on a qualitative visual description of similarities between the frequency content of PPG and ABP waveform indicate that, during normal, chest compressions and pauses states, PPG resemble to ABP, and PPG signal at finger can indicate pulse presence at the moment the heart resumes beating. Furthermly, Tusman et al. [58] experimentally verified the feasibility of categorizing individual's BP based on the contour of the PPG signal, and the accuracy reached 98.4% and 97.8% for diagnosing hypotension and hypertension, respectively. Dash et al. [59] investigated the mutual conversion between PPG waveform and ABP waveform. Based on the linear transfer function (LTF) technique, they firstly fitted a model with ABP as input and PPG as output, and then fitted an inverse model with PPG as input and

PPG segment
(a)
conv1d
conv1d_1
conv1d_2
conv1d_3
conv1d_4
conv1d_5
dense
dense_1
dense_2
dense_3
conv1d_6
conv1d_7
conv1d_8
conv1d_9
(b)
conv1d
conv1d_1
conv1d_2
conv1d_3
conv1d_4
conv1d_5
dense
dense_1
dense_2
dense_3
conv1d_6
up_sampling1d
conv1d_7
up_sampling1d_1
conv1d_8
up_sampling1d_2
conv1d_9
(c)
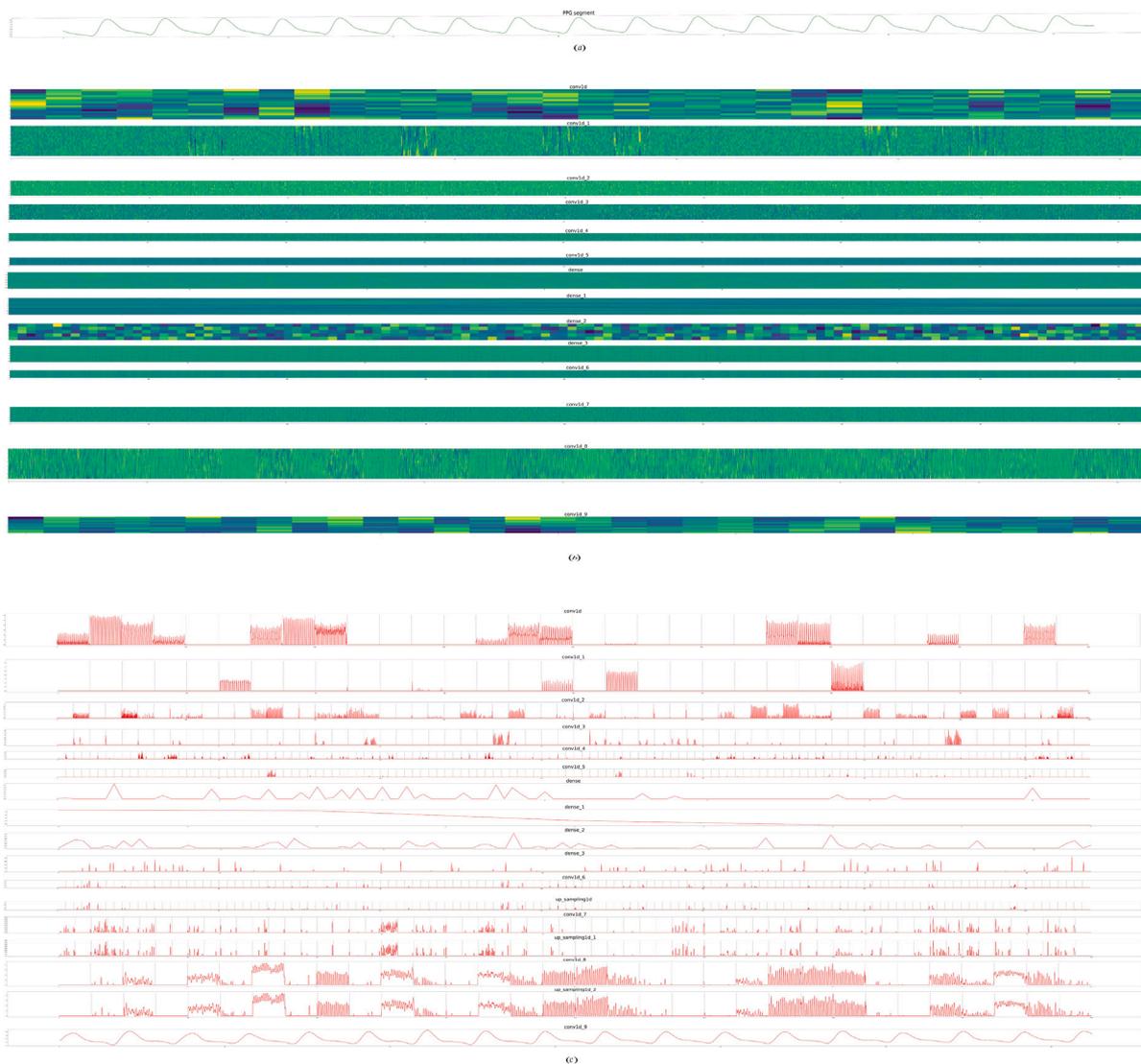
Fig. 13. Visualization of the learned model for converting PPG signal to ABP waveform. (a) raw PPG segment; (b) learned weights of each model layer. There are totally 14 layers presented (including 6 conv. layers in encoder, 4 dense layers, 4 conv. layers in decoder). For convolution layer, the convolution kernels (i.e. weights) were flattened along the output channel dimension, and adjacent channels were separated by red dashed lines; (c) filters response of each layer of the model. The are totally 17 layer responses (including 6 conv. layers response in encoder, 4 dense layers response, 4 conv. layers response in decoder, 3 up-sampling layers response in decoder). The response corresponding the convolution layer is flattened along the channel dimension, and adjacent channels were separated by blue dashed lines.

ABP as output. Experimental results on ten individuals from MIMIC II indicated that the average estimation accuracy reaches 84.4% and 84.7%, respectively.

From ML perspective, training predictive model using either ABP waveform or BP value as a reference does not make much difference in the sense that both of them are actually supervised learning tasks. The development of ML, especially in deep learning, has achieved great success in almost all aspects and in all kinds of scenarios due to its strong ability of feature learning, expression and modeling of complex relationships [41]. The proposed model could map any given signal (e.g. PPG signal) to another signal (e.g. ABP signal), but to make the conversion meaningful, the signals should be correlated in some way. As mentioned above, PPG signal and the counterpart ABP signal are highly correlated in both morphological and physiological sense, and this pave the way for the proposed method that convert PPG signal to ABP waveform (also known as generate ABP waveform using PPG signal in related studies [44]).

Visualization skill is a common tool in interpreting deep learning model [24,25]. To further understand and explore the proposed model that converting PPG signal to ABP waveform, as Fig. 13 presents, we visualized the learned model weights (Fig. 13(b)) and visualized the filters response of each model layer (Fig. 13(c)) using a PPG segment (Fig. 13(a)) as input. As far as we know, this is the first visual interpretation of the model for BP (waveform) prediction among all deep learning methods [1,8,15–22,33,38,39,43,44] in this field. Inherited from the highly symmetrical model structure comprised of encoder and decoder, as Fig. 13(b) presents the learned weights of each layer of Encoder is visually mirror symmetric with that of Decoder. Differently from the hand-crafted features with explicit significance, the features automatically learned by the deep model is highly abstract and with hierarchical structure. We then fed the learned model with a PPG segment, and the response of each model layer is presented. For a convolution layer with multiple kernels, each kernel, as a feature extractor, to learned useful representation/pattern, as Fig. 13(c) shows, the response of each layer is sparse. For the encoder part, with the increase of layer, the response is very like waveform segments at the beginning, and gradually becomes more and more abstract. For the decoder part, with the increase of layer, each layer decode the response from the upper layer into more concrete response, and the response of the last layer (i.e.the layer named ‘conv1d_9’ with single convolution kernel) of Decoder is the predicted ABP waveform.

7. Limitations

This study has several limitations. *As for the data used*, on the one hand, the data comes from ICU patients whose average age and blood pressure are generally higher than the general population [16,52]. In other words, the data is biased. Therefore, retraining the model on more unbiased dataset and further evaluation of the method on more general population is meaningful. On the other hand, although the total number of records in the final dataset is large enough, limited by the data (each record with duration from 8 ∼ 592 s), there are only 32 samples reserved in each record, and it may be not enough for adequate learning of a domain. *Methodologically*, the proposed method essentially utilizes the powerful abilities of deep learning in feature learning and complex relationships modeling. At the same time, it inherits the shortcomings of deep learning—it is a black box model and lacks interpretability [45]. Note that we first attempt to interpret the model through visualization in this area. In addition, we used one-hot encoding as the ground-truth domain label when computing domain classifier loss, of which the default assumption is that different domains are independent of each other, and the similarity among different domains is ignored. In fact, there are different degrees of similarity between different domains, although there is currently no method to quantify the waveform differences between different individuals. Therefore, we cannot further analyze the impact of individual differences on training and on the generalizability of model. *Clinically*, the sensitivity of the predictive model to a sudden rise or down of BP is an key & challenging issue to be further explored. Currently, as far as we know, almost all studies (including our work) research BP (waveform) prediction in a static environment (different samples in a record is independent, and there is no consideration of their temporal correlation). From a long-term perspective, the blood pressure of an individual (especially critically ill patient) changes dynamically, this change can be roughly divided into two types: (1) changes in blood pressure patterns, also known as concept drift [60]; (2) A transient increase or decrease in blood pressure caused by an event or adverse reaction, also known as abnormal. We believe that in order to achieve long-term accurate blood pressure (waveform) prediction, the temporal correlation of individual signals must be considered, and there must be some mechanism to automatically detect sudden changes (rising or falling) in blood pressure patterns.

8. Conclusions

In this study, we proposed a cuffless, continuous approach—RDAE for visualizing ABP waveform with only single-point PPG signal as input. Concretely, we developed convolution-based deep autoencoder, of which an inherent end-to-end mode with high symmetry, coupled with domain adversarial training strategy, which regularizes the latent representation to be domain-invariant, to visualize ABP waveform. Results show that not only a good resulting ABP waveform is obtained, but also a high blood pressure prediction accuracy is achieved. Ablation study reveals that, first, the fully-convolution block in building Encoder and Decoder of AE helps RDAE to capture features effectively. Second, domain adversarial training makes the loss converge faster in the training phase and allows the general model better adapt to new individuals when using the same number of target samples for fine-tuning.

Specifically, we highlight this study with several insights and traits: (*i*) the fully-convolution block is more appropriate than the traditional 'convolution+pooling' block in processing physiological signal data; (*ii*) domain adversarial training forces AE to learn a more general, low-dimensional, cross domain latent representation, which overcome the problem of individual differences that impede the training to a certain extent, and improved model generalization ability; (*iii*) the results of RDAE with different configurations were presented, providing a flexible selection of model in practice.

It is worth mentioning that the domain classifier module in RDAE is only needed in training phase. In practice, the trained model (RDAE) could be optimized to exclude the unnecessary part (e.g. domain classifier module) through tools (e.g. tensorflow lite converter [https://tensorflow.google.cn/lite/convert/index]) before it is used for inference on mobile devices (e.g. smart phone) or embedded devices. In other words, RDAE will not increase storage overhead and bring inference latency. Actually, all one needs is a device with PPG sensor embedded [61].

In future work, we plan to do further research from the following aspects:

(1) Three minutes is a proper time interval for in-clinic calibration of blood pressure system. In practice, a reasonable longer duration than 80 s for calibration may further improve the precision [17].
(2) Actually, a proper loss function with domain similarities considered may further promote training [48]. In future work, we consider modeling domain similarities by encoding it into domain label.
(3) Further improve the model sensitivity to sudden rise or up in BP by considering temporal correlation between samples and employing concept drift detection [62] in a streaming environment.

CRediT authorship contribution statement

Keke Qin: Conceptualization, Methodology, Software, Writing – original draft. **Wu Huang:** Conceptualization, Writing – review & editing, Supervision. **Tao Zhang:** Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.bspc.2021.102972. Supplementary material contains a Table A.1 comprehensively summarizes the comparison of related studies.

References

[1] S. Lee, J.H. Chang, Deep belief networks ensemble for blood pressure estimation, IEEE Access PP (2017) 1, http://dx.doi.org/10.1109/ACCESS.2017.2701800.
[2] G. Thambiraj, U. Gandhi, V. Devanand, M. Umapathy, Noninvasive cuffless blood pressure estimation using pulse transit time, Womersley number, and photoplethysmogram intensity ratio, Physiol. Meas. 40 (2019) http://dx.doi.org/10.1088/1361-6579/ab1f17.
[3] I. Sharifi, S. Goudarzi, M.B. Khodabakhshi, A novel dynamical approach in continuous cuffless blood pressure estimation based on ECG and PPG signals, Artif. Intell. Med. 97 (2018) http://dx.doi.org/10.1016/j.artmed.2018.12.005.
[4] T. Arakawa, Recent research and developing trends of wearable sensors for detecting blood pressure, Sensors 18 (2018) 2772, http://dx.doi.org/10.3390/s18092772.
[5] Q. Zhu, X. Tian, C.W. Wong, M. Wu, ECG reconstruction via PPG: A pilot study, in: 2019 IEEE EMBS International Conference on Biomedical and Health Informatics, BHI, IEEE, 2019, pp. 1–4, http://dx.doi.org/10.1109/BHI.2019.8834612.
[6] J. Allen, Photoplethysmography and its application in clinical physiological measurement, Physiol. Meas. 28 (2007) R1–39, http://dx.doi.org/10.1088/0967-3334/28/3/R01.
[7] G. Martínez, N. Howard, D. Abbott, K. Lim, R. Ward, M. Elgendi, Can photoplethysmography replace arterial blood pressure in the assessment of blood pressure? J. Clin. Med. 7 (2018) 316, http://dx.doi.org/10.3390/jcm7100316.
[8] G. Slapnicar, N. Mlakar, M. Lustrek, Blood pressure estimation from Photoplethysmogram using a spectro-temporal deep neural network, Sensors 19 (2019) 3420, http://dx.doi.org/10.3390/s19153420.
[9] L. Peter, N. Noury, M. Cerny, A review of methods for non-invasive and continuous blood pressure monitoring: Pulse transit time method is promising? IRBM 35 (2014) http://dx.doi.org/10.1016/j.irbm.2014.07.002.

[10] A. Esmaili, M. Kachuee, M. Shabany, Nonlinear cuffless blood pressure estimation of healthy subjects using pulse transit time and arrival time, IEEE Trans. Instrum. Meas. PP (2017) 1–10, http://dx.doi.org/10.1109/TIM.2017.2745081.

[11] E. Monte-Moreno, Non-invasive estimate of blood glucose and blood pressure from a photoplethysmograph by means of machine learning techniques, Artif. Intell. Med. 53 (2011) 127–138, http://dx.doi.org/10.1016/j.artmed.2011.05.001.

[12] S. Ahmad, S. Chen, K. Soueidan, I. Batkin, M. Bolic, H. Dajani, V. Groza, Electrocardiogram-assisted blood pressure estimation, IEEE. Trans. Biomed. Eng. 59 (2012) 608–618, http://dx.doi.org/10.1109/TBME.2011.2180019.

[13] M. Kachuee, M.M. Kiani, H. Mohammadzade, M. Shabany, Cuff-less high-accuracy calibration-free blood pressure estimation using pulse transit time, in: 2015 IEEE International Symposium on Circuits and Systems, ISCAS, IEEE, 2015, pp. 1006–1009, http://dx.doi.org/10.1109/ISCAS.2015.7168806.

[14] M. Kachuee, M.M. Kiani, H. Mohammadzade, M. Shabany, Cuff-less blood pressure estimation algorithms for continuous health-care monitoring, IEEE. Trans. Biomed. Eng. 64 (2016) 1, http://dx.doi.org/10.1109/TBME.2016.2580904.

[15] N. Ibtehaz, M.S. Rahman, PPG2ABP: Translating photoplethysmogram (PPG) signals to arterial blood pressure (ABP) waveforms using fully convolutional neural networks, 2020, arXiv:arXiv:2005.01669.

[16] S. Baek, J. Jang, S. Yoon, End-to-End blood pressure prediction via fully convolutional networks, IEEE Access 7 (2019) 1, http://dx.doi.org/10.1109/ACCESS.2019.2960844.

[17] L.D. Zhang, N.C. Hurley, B. Ibrahim, E. Spatz, H.M. Krumholz, R. Jafari, M.J. Bobak, Developing personalized models of blood pressure estimation from wearable sensors data using minimally-trained domain adversarial neural networks, in: Machine Learning for Healthcare Conference, PMLR, 2020, pp. 97–120.

[18] P. Su, X.R. Ding, Y.T. Zhang, J. Liu, F. Miao, N. Zhao, Long-term blood pressure prediction with deep recurrent neural networks, in: 2018 IEEE EMBS International Conference on Biomedical and Health Informatics, BHI, IEEE, 2018, pp. 323–328, http://dx.doi.org/10.1109/BHI.2018.8333434.

[19] H. Eom, D. Lee, S. Han, Y. Hariyani, Y. Lim, I. Sohn, K. Park, C. Park, End-to-End deep learning architecture for continuous blood pressure estimation using attention mechanism, Sensors 20 (2020) 2338, http://dx.doi.org/10.3390/s20082338.

[20] O. Schlesinger, N. Vigderhouse, Y. Moshe, D. Eytan, Estimation and tracking of blood pressure using routinely acquired photoplethysmographic signals and deep neural networks, Crit. Care Explor. 2 (2020) e0095, http://dx.doi.org/10.1097/CCE.0000000000000095.

[21] S. Tanveer, M. Hasan, Cuffless blood pressure estimation from electrocardiogram and photoplethysmogram using waveform based ANN-LSTM network, Biomed. Signal Process. Control 51 (2019) 382–392, http://dx.doi.org/10.1016/j.bspc.2019.02.028.

[22] X.M. Fan, H.L. Wang, F. Xu, Y. Zhao, K.L. Tsui, Homecare-oriented intelligent long-term monitoring of blood pressure using Electrocardiogram signals, IEEE Trans. Ind. Inform. PP (2019) 1, http://dx.doi.org/10.1109/TII.2019.2962546.

[23] H. P, W. RWCGR, M. J, W. C, S. F, A series of case studies on detection of spontaneous pulse by photoplethysmography in cardiopulmonary resuscitation, AM. J. Emerg. Med. 38 (2019) http://dx.doi.org/10.1016/j.ajem.2019.05.044.

[24] G. Hinton, R. Salakhutdinov, Reducing the dimensionality of data with neural networks, Science 313 (2006) 504–507, http://dx.doi.org/10.1126/science.1127647.

[25] A. Krizhevsky, I. Sutskever, G. Hinton, ImageNet classification with deep convolutional neural networks, in: NeurIPS, Vol. 25, 2012, http://dx.doi.org/10.1145/3065386.

[26] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (1997) 1735–1780, http://dx.doi.org/10.1162/neco.1997.9.8.1735.

[27] H. Harutyunyan, H. Khachatrian, D. Kale, A. Galstyan, Multitask learning and benchmarking with clinical time series data, Sci. Data 6 (2019) http://dx.doi.org/10.1038/s41597-019-0103-9.

[28] S.Y. Shih, F.K. Sun, H.Y. Lee, Temporal pattern attention for multivariate time series forecasting, Mach. Learn. 108 (2019) http://dx.doi.org/10.1007/s10994-019-05815-0.

[29] N. Mishra, M. Rohaninejad, X. Chen, P. Abbeel, A simple neural attentive meta-learner, 2017, arXiv:arXiv:1707.03141.

[30] S.S. Mousavi, M. Firouzmand, M. Charmi, M. Hemmati, M. Moghadam, Y. Ghorbani, Blood pressure estimation from appropriate and inappropriate PPG signals using a whole-based method, Biomed. Signal Process. Control 47 (2019) 196–206, http://dx.doi.org/10.1016/j.bspc.2018.08.022.

[31] Y. Gani, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, V. Lempitsky, Domain-adversarial training of neural networks, J. Mach. Learn. Res. 17 (59) (2016) 1–35.

[32] C. El-Hajj, P. Kyriacou, A review of machine learning techniques in photoplethysmography for the non-invasive cuff-less measurement of blood pressure, Biomed. Signal Process. Control 58 (2020) 101870, http://dx.doi.org/10.1016/j.bspc.2020.101870.

[33] X.M. Xing, M.S. Sun, Optical blood pressure estimation with photoplethysmography and FFT-based neural networks, Biomed. Opt. Express 7 (2016) 3007–3020, http://dx.doi.org/10.1364/BOE.7.003007.

[34] G. Thambiraj, U. Gandhi, U. Mangalanathan, J.M.J. Valanarasu, M. Anand, Investigation on the effect of Womersley number, ECG and PPG features for cuff less blood pressure estimation using machine learning, Biomed. Signal Process. Control 60 (2020) 101942, http://dx.doi.org/10.1016/j.bspc.2020.101942.

[35] D. Fujita, A. Suzuki, K. Ryu, PPG-based systolic blood pressure estimation method using PLS and level-crossing feature, Appl. Sci. 9 (2019) 304, http://dx.doi.org/10.3390/app9020304.

[36] S. Bose, N. Sree, K. Arumugam, Sparse representation of photoplethysmogram using K-SVD for cuffless estimation of arterial blood pressure, in: 2017 4th International Conference on Advanced Computing and Communication Systems, ICACCS, 2017, pp. 1–5, http://dx.doi.org/10.1109/ICACCS.2017.8014669.

[37] F. Miao, Z.D. Liu, J.K. Liu, B. Wen, Y. Li, Multi-sensor fusion approach for cuff-less blood pressure measurement, IEEE J. Biomed. Health PP (2019) 1, http://dx.doi.org/10.1109/JBHI.2019.2901724.

[38] M. Forouzanfar, H. Dajani, V. Groza, M. Bolic, S. Rajan, Feature-based neural network approach for oscillometric blood pressure estimation, IEEE Trans. Instrum. Meas. 60 (2011) 2786–2796, http://dx.doi.org/10.1109/TIM.2011.2123210.

[39] Y.C. Hsu, Y.H. Li, C.C. Chang, L.N. Harfiya, Generalized deep neural network model for cuffless blood pressure estimation with photoplethysmogram signal only, Sensors 20 (2020) http://dx.doi.org/10.3390/s20195668.

[40] B. Zhang, J.D. Ren, Y.Q. Cheng, B. Wang, Z.Y. Wei, Health data driven on continuous blood pressure prediction based on gradient boosting decision tree algorithm, IEEE Access PP (2019) 1, http://dx.doi.org/10.1109/ACCESS.2019.2902217.

[41] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436–444, http://dx.doi.org/10.1038/nature14539.

[42] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Vol. 9351, Springer, 2015, pp. 234–241, http://dx.doi.org/10.1007/978-3-319-24574-4_28.

[43] A. T., C. S., An estimation method of continuous non-invasive arterial blood pressure waveform using photoplethysmography: A U-Net architecture-based approach, Sensors 21 (2021) 1867, http://dx.doi.org/10.3390/s21051867.

[44] M. Sadrawi, Y.-T. Lin, C.-H. Lin, B. Mathunjwa, S.-Z. Fan, M. Abbod, J.-S. Shieh, Genetic deep convolutional autoencoder applied for generative continuous arterial blood pressure via Photoplethysmography, Sensors 20 (2020) 3829, http://dx.doi.org/10.3390/s20143829.

[45] Y. Yu, K. Chan, C. You, C. Song, Y. Ma, Learning diverse and discriminative representations via the principle of maximal coding rate reduction, in: NeurIPS, Vol. 33, 2020.

[46] J. Oh, J. Wang, J. Wiens, Learning to exploit invariances in clinical time-series data using sequence transformer networks, in: Machine Learning for Healthcare Conference, PMLR, 2018, pp. 332–347.

[47] J. Springenberg, A. Dosovitskiy, T. Brox, M. Riedmiller, Striving for simplicity: The all convolutional net, 2014, arXiv:arXiv:1412.6806.

[48] J.P. Li, S. Qiu, Y.Y. Shen, C.L. Liu, H.G. He, Multisource transfer learning for cross-subject EEG emotion recognition, IEEE Trans. Cybern. PP (2019) 1–13, http://dx.doi.org/10.1109/TCYB.2019.2904052.

[49] Y.B. Liang, M. Elgendi, Z.C. Chen, R. Ward, An optimal filter for short photoplethysmogram signals, Sci. Data 5 (2018) 180076, http://dx.doi.org/10.1038/sdata.2018.76.

[50] Z.H. Zhou, Machine Learning, Tsinghua University Press, 2016.

[51] M. Simjanoska, S. Kochev, J. Tanevski, A.M. Bogdanova, G. Papa, T. Eftimov, Multi-level information fusion for learning a blood pressure predictive model using sensor data, Inform. Fusion 58 (2019) http://dx.doi.org/10.1016/j.inffus.2019.12.008.

[52] M. Saeed, M. Villarroel, A.T. Reisner, G. Clifford, L.W. Lehman, G. Moody, T. Heldt, T.H. Kyaw, B. Moody, R.G. Mark, Multiparameter intelligent monitoring in intensive care II (MIMIC-II): A public-access intensive care unit database, Crit. Care Med. 39 (5) (2011) 952.

[53] E. OBrien, B. Waeber, G. Parati, J. Staessen, M. Myers, Blood pressure measuring devices: Recommendations of the European society of hypertension, BMJ (Clin. Res. Ed.) 322 (2001) 531–536.

[54] E. OBrien, J. Petrie, W. Littler, M. Swiet, P. Padfield, K. O'Malley, M. Jamieson, D. Altman, M. Bland, N. Atkins, The British Hypertension Society protocol for the evaluation of automated and semi-automated blood pressure measuring devices with special reference to ambulatory systems, J. Hypertens. 8 (1990) 607–619, http://dx.doi.org/10.1097/00004872-199007000-00004.

[55] J.A. Hartigan, M.A. Wong, A K-means clustering algorithm: Algorithm AS 136, 28, 1979, pp. 100–108, http://dx.doi.org/10.2307/2346830,

[56] L. Maaten, G.F. Hinton, Viualizing data using t-SNE, J. Mach. Learn. Res. 9 (2008) 2579–2605.

[57] X. Zhang, X. Zhou, M. Lin, J. Sun, ShuffleNet: An extremely efficient convolutional neural network for mobile devices, 2018, pp. 6848–6856, http://dx.doi.org/10.1109/CVPR.2018.00716.

[58] G. Tusman, C. Acosta, S. Pulletz, S. Bohm, A. Scandurra, J. martinez arca, M. Madorno, F. Suarez-Sipmann, Photoplethysmographic characterization of vascular tone mediated changes in arterial pressure: An observational study, J Clin. Monit. Comput. 33 (2019) http://dx.doi.org/10.1007/s10877-018-0235-z.

[59] A. Dash, N. Ghosh, A. Patra, A. Dutta Choudhury, Estimation of arterial blood pressure waveform from photoplethysmogram signal using linear transfer function approach, in: 2020 42nd Annual International Conference of the IEEE Engineering in Medicine Biology Society, Vol. 2020, EMBC, 2020, pp. 2691–2694, http://dx.doi.org/10.1109/EMBC44109.2020.9175696.

[60] P.-H. Chiang, S. Dey, Offline and online learning techniques for personalized blood pressure prediction and health behavior recommendations, IEEE Access PP (2019) 1, http://dx.doi.org/10.1109/ACCESS.2019.2939218.

[61] A. Chandrasekhar, C.S. Kim, M. Naji, K. Natarajan, J.O. Hahn, R. Mukkamala, Smartphone-based blood pressure monitoring via the oscillometric finger-pressing method, Sci. Transl. Med. 10 (2018) http://dx.doi.org/10.1126/scitranslmed. aap8674.

[62] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, H. Bouchachia, A survey on concept drift adaptation, ACM Comput. Surv. 46 (2014) http://dx.doi.org/10. 1145/2523813.