Int. J. Appl. Math. Comput. Sci., 2023, Vol. 33, No. 1, 151–162
DOI: 10.34768/amcs-2023-0012

HOSPITALIZATION PATIENT FORECASTING BASED ON MULTI–TASK DEEP LEARNING

MIN ZHOU [a], XIAOXIAO HUANG [a,*], HAIPENG LIU [b], DINGCHANG ZHENG [b]

[a]The First Affiliated Hospital
Zhejiang University School of Medicine
No.79 Qingchun Rd., 310003, Hangzhou, China
e-mail: {minzhou,huangxiaoxiao}@zju.edu.cn

[b]Research Centre for Intelligent Healthcare
Coventry University
Priory Street, CV1 5FB, Coventry, UK
e-mail: {haipeng.liu,dingchang.zheng}@coventry.ac.uk

Forecasting the number of hospitalization patients is important for hospital management. The number of hospitalization patients depends on three types of patients, namely admission patients, discharged patients, and inpatients. However, previous works focused on one type of patients rather than the three types of patients together. In this paper, we propose a multi-task forecasting model to forecast the three types of patients simultaneously. We integrate three neural network modules into a unified model for forecasting. Besides, we extract date features of admission and discharged patient flows to improve forecasting accuracy. The algorithm is trained and evaluated on a real-world data set of a one-year daily observation of patient numbers in a hospital. We compare the performance of our model with eight baselines over two real-word data sets. The experimental results show that our approach outperforms other baseline algorithms significantly.

Keywords: hospitalization patients, forecasting, neural network, multitask learning.

1. Introduction

In China, due to the large amount of population and rapid growth of elderly people, there is an increasing gap between supply and demand of the relatively scarce medical resources (Luo *et al*., 2017). Many patients cannot get treatment immediately due to the shortage of healthcare resources, especially the inpatient beds. To alleviate the shortage of healthcare resources, it is important to achieve the reasonable allocation of inpatient beds based on the accurate estimation of the hospital occupancy. Accurate forecasting of inpatient occupancy can improve the allocation of healthcare resources in many aspects including nurse staffing (Ledersnaider and Channon, 1998) and patients receiving (Mackay, 2001).

However, building an accurate forecasting system for hospital occupancy is difficult. Hospital occupancy depends dynamically on the flows of admission and discharged patients. These patient flows fluctuate from week to week. Therefore, to achieve accurate forecasting of hospital occupancy, we should solve two sub-tasks of forecasting the admission patients and the discharged patients.

There are some previous works that focus on patient flow forecasting, which can be divided into three groups, including statistical methods, machine learning methods and neural network-based methods.

1. *Statistical methods.* There are two main statistical methods, namely autoregression integrated moving average (ARIMA) and exponential smoothing (ES). The ARIMA can perfectly model the linear patterns of a time series with minimum computational efforts, so it is widely used by researchers for various types of forecasting tasks (Box *et al*., 2015). Sun *et al*. (2009) adopted the ARIMA to forecasting daily attendances at an emergency department. Li *et al*. (2015) adopted the ARIMA to forecast monthly outpatient visits. Hadavandi *et al*. (2012)

*Corresponding author

developed a hybrid ARIMA model for outpatient visits forecasting. What is more, Bergs *et al*. (2014) used the automated exponential smoothing approach to predict monthly visits.

2. *Machine learning methods.* Machine learning (ML) methods have been proposed in the academic literature as alternatives to statistical ones for time-series forecasting (Makridakis *et al*., 2018). By treating time series forecasting problems as regression problems, many machine learning models are applied to find an appropriated mapping between input features and prediction values. Wang and Gu (2014) proposed to use support vector regression to forecast the daily diarrheal outpatient visits in Shanghai.

3. *Neural network-based methods.* In recent years, artificial neural networks (ANNs) attract more and more attention in various fields (Bingi and Prusty, 2021; Kowal *et al*., 2021). ANNs have been adopted in many difficult tasks, like machine translation (Cheng, 2019), image recognition (Chen *et al*., 2019) and speech recognition (Nassif *et al*., 2019) etc., and obtained satisfying results due to their intrinsic property that they can approximate any nonlinear function (Cross *et al*., 1995). There are also some works adopting ANNs for forecasting problems. Menke *et al*. (2014) adopted ANNs to predict emergency department (ED) volume. Jiang *et al*. (2018) adopted an ANN to model the flow of patients under different severities. Hou *et al*. (2021) proposed a time-varying mechanism to solve forecasting problems with imbalanced data.

However, these methods were based on single-task strategies, with no reference to the relationship among admission patients, discharged patients and inpatients. To overcome the limitations of the single-task strategy, a multi-task strategy has been proposed to train a model by learning multiple tasks which are different but correlated (Zhang and Yang, 2021). A trained model based on the multi-task strategy is able to capture the latent relationship between these tasks and to achieve better performance in forecasting a process with different flows. The multi-task strategy has been widely used in many fields including natural language processing (Sanh *et al*., 2019), compute vision (Dai *et al*., 2016), and urban computing (Zhang *et al*., 2019).

In order to forecast the number of admission patients, discharged patients and inpatients simultaneously, in this paper, we propose an end-to-end neural network based forecasting model with the multi-task strategy, named HPNet. We construct three groups of features, namely date features, closeness features and periodic features as the inputs of HPNet. To learn the latent influence of timestamps, we propose a timestamp fusion strategy that will embed date features into low-dimensional vectors. The HPNet consists of three key components: an encoder of historical data, an embedding layer of date features, and three forecasting modules. Moreover, we design a constraint loss to model the relationship among the three types of patients and bound the forecasting results. Finally, we conduct extensive experiments based on two real-world data sets. The experimental results show that our approach outperforms other baseline algorithms significantly.

The contributions of this paper can be concluded as follows:

- A novel neural network-based forecasting model is proposed to forecast the number of admission patients, discharged patients and inpatient simultaneously.
- The forecasting timestamp fusion strategy and a constraint loss are proposed to improve the forecasting accuracy.
- We conduct extensive experiments based on two real-world data sets and compare the performance of our model with six baselines. The experimental results demonstrate that HPNet outperforms the state-of-the-art solution.

The rest of the paper is organized as follows. Section 2 outlines the preliminaries, including some important concepts and our problem definition. Section 3 presents the dataset we used and some insights about this data set. Section 4 describes our solution in detail. Section 5 presents the experimental results of our model on real-world data sets. Finally, we conclude our paper in Section 6.

2. Preliminaries

2.1. Three types of patients. There are three types of patients in our problem setting, they are *admission patients*, *discharged patients* and *inpatients*. Their detailed descriptions are as follows:

- *Admission patients:* For a certain day t, admission patients refer to the new patients who are admitted into the hospital on this day. The number of admission patients on the day t is denoted by n_a^t.
- *Discharged patients:* In contrast to admission patients, discharged patients refer to the patients who leave the hospital every day. The number of discharged patients on the day t is denoted by n_d^t.
- *Inpatients:* Inpatients represent all patients who are in the hospital at the end of the day. Apparently, the number of inpatients is influenced by admission patients and discharged patients.

The number of the three types of patients can be formulated as

$$n_p^t = n_p^{t-1} + n_a^t - n_d^t, \tag{1}$$

where n_p, n_a, n_d represent the numbers of inpatients, admission patients and discharged patients, respectively. Here n_p^t and n_p^{t-1} represent the numbers of inpatients on the day t and day $t-1$, respectively.

2.2. Hospitalization prediction problem. Generally speaking, hospitalization prediction is a time series forecasting problem, which aims to predict the number of inpatients at time $T+1$ given the historical observations until time T. But the hospitalization in our paper contains three perspectives, which are admission patients, discharged patients and inpatients, as defined above. Our task in this paper is to predict the number of all these patients in the future.

Specifically, assume that the historical hospitalization observations $\{X_t \mid t = t_1, \ldots, t_T\}$ were made with the corresponding timestamp t, where $X_t = \{n_a^t, n_d^t, n_p^t\}$ is a triplet representing the number of admission patient, discharged patients and inpatients on the day t. The goal of this paper is to collectively predict $X_{t_{T+1}}$ in the future.

3. Data description and analysis

In this section, we first describe a real-world hospitalization data set collected from a branch hospital of The First Affiliated Hospital of the Zhejiang University School of Medicine. Then we conduct empirical analyses to explore the characteristics of the hospitalization distribution.

3.1. Data description. We collect hospitalization data from 2015-01-01 to 2019-12-31. The data set contains daily numbers of admission patients, discharged patients and inpatients with time information. An example of such data is (2019-10-24,524,541,3378). It indicates that 524 patients were admitted to the hospital while 541 patients left the hospital on October 24th, 2019. At the end of this day, there were 3378 patients in the hospital.

Figure 1 displays the distributions and pairwise scatter plots of the three types of patients. As we can see from Fig. 1(a), the Pearson coefficient between the number of admission patients and the number of discharged patients is 0.94, which proves that the number of admission patients is positively correlated with the number of discharged patients. There is no obvious correlation between other types of patients (Figs. 1(b) and (c)).

3.2. Empirical analysis and feature extraction. In this section, we make an empirical analysis on hospitalization data and introduce what features we extract for prediction.

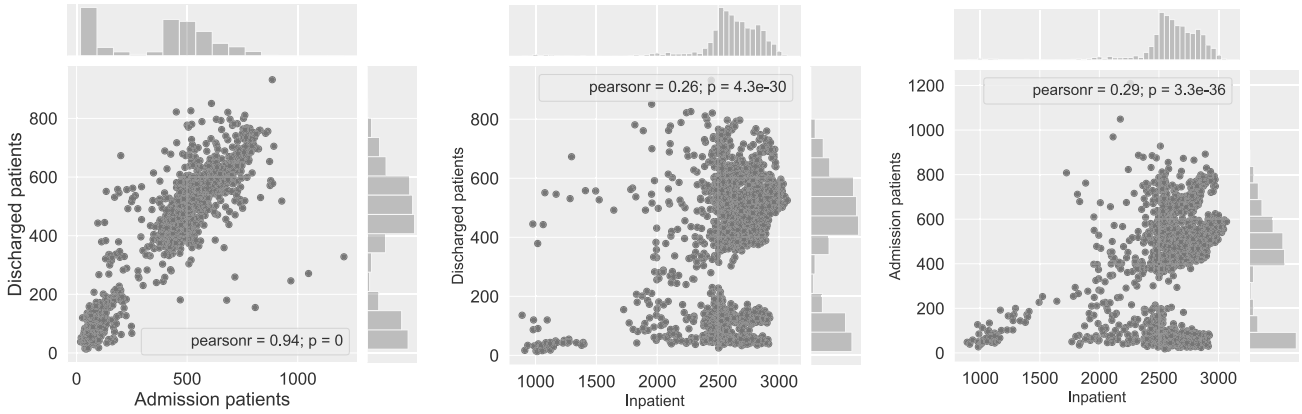
Date features. Figure 2(a) shows the box plot of admission patients and discharged patients on different weekdays. As can be seen, the numbers of admission patients and discharged patients show three different trends on different weekdays. Specifically, the numbers of admission patients and discharged patients are highest on Monday while their numbers on Saturday and Sunday are significantly low. The rest of the four days (from Tuesday to Thursday) show similar distributions. This phenomenon reflects the characteristic that admission patients and discharged patients are active on working days while their numbers are relatively stable on Saturday and Sunday. According to this observation, we can conclude that weekday is an important factor that should be taken into consideration when we predict hospitalization in the future.

Similarly, Fig. 2(b) displays the box plot of the numbers of admission patients and discharged patients on working days and holidays, respectively. We can see that the numbers of both admission patients and discharged patients on working days are much larger than that on holidays. Therefore, the time information indicating working or holidays is also important to predict hospitalization.

All the above observations indicate the temporal trend of hospitalization distributions. Hence, we extract the two types of date features based on the given time information: weekday feature f_w and holiday feature f_h. The values of f_w are $\{1, 2, 3, 4, 5, 6, 7\}$ indicating the weekday and $f_h \in \{0, 1\}$ indicating whether or not it is a holiday.

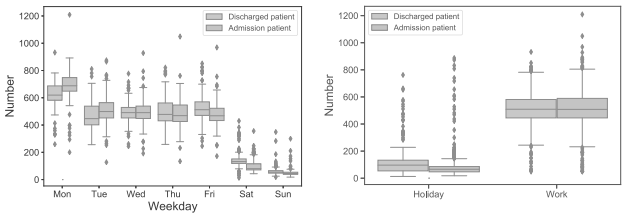
Closeness features. Figure 3 summarizes the Pearson correlation of admission patients and discharged patients. As can be seen, the numbers of admission patients from Monday to Friday are strongly correlated with one another, while they are weakly correlated with the numbers of admission patients on Saturday and Sunday. On the contrary, the numbers of admission patients on Saturday and Sunday are strongly correlated with each other, while they are weakly correlated with the number of admission patients from Monday to Friday. Besides, discharged patients show a similar correlation. This indicates that when we predict hospitalization on different weekdays, historical data should contribute different importance. For example, if the forecasting day is from Monday to Friday, we should pay more attention to the historical hospitalization of working days.

Then we conduct the Ljung–Box test for admission patients and discharged patients to study the existence of autocorrelation. The p-values are less than 0.05 ($3.5 \times$



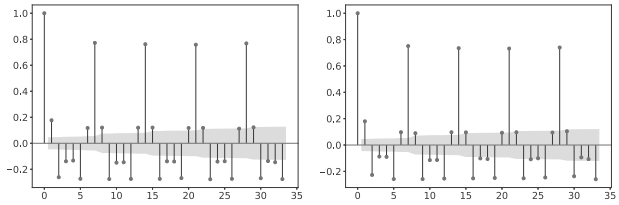
(a) discharged patients vs. admission patients (b) discharged patients vs. inpatients (c) admission patients vs. inpatients

Fig. 1. Pairwise scatter plots of three types of patients.



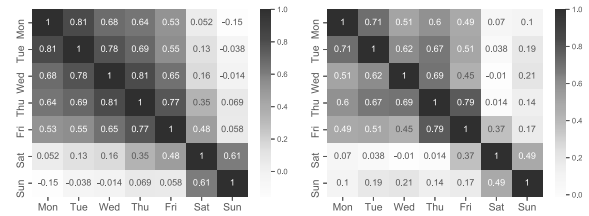
(a) weekday distribution (b) holiday distribution

Fig. 2. Box plots.



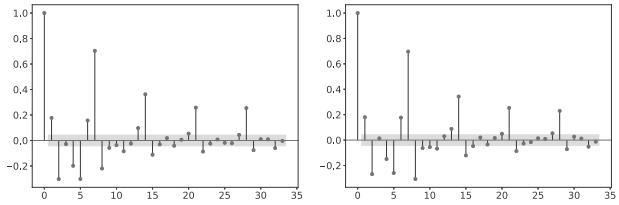
(a) auto-correlation of admission patients (b) auto-correlation of discharged patients

Fig. 4. Auto-correlation.



(a) correlation of admission patients (b) correlation of discharged patients

Fig. 3. Pearson correlation.



(a) partial auto-correlation of admission patients (b) partial auto-correlation of discharged patients

Fig. 5. Partial auto-correlation.

10^{-14} and 1.4×10^{-14}). Therefore, we can conclude that the data are not independently distributed and they exhibit serial correlation. To quantitively determine how much historical data should be utilized for forecasting, we conduct autocorrelation and partial autocorrelation analyses for admission patients and discharged patients. Figures 4 and 5 display the ACF and PACF of admission patients and discharged patients, respectively. As can be seen, both the series of admission patients and discharged patients show strong autocorrelation with the time lag of 7. Therefore, in our experiments, we use a time window whose size equals 7 to select historical hospitalization data for prediction. Formally, given the prediction time $T+1$, we extract closeness features $f_c = \{X_{t_{T-6}}, X_{t_{T-5}}, \ldots, X_{t_T}\}$

Periodic features. According to Figs. 4 and 5, the auto-correlation coefficients get maximal when the time lag is a multiple of 7, which means there are obvious periodic patterns for admission patients and discharged patients. Therefore, to provide more information and improve forecasting accuracy, we extract historical hospitalization observations from the previous four weeks

at seven-day intervals. Formally, given the prediction time $T+1$, we extract periodic features $f_p = \{X_{t_{T-6}}, X_{t_{T-13}}, X_{t_{T-20}}, X_{t_{T-27}}\}$.

4. Methodology

Figure 6 presents the framework of HPNet. The main idea of the HPNet is the following. Given the historical data and the target forecasting timestamp, the HPNet will output the numbers of admission patients, discharged patients, and inpatients simultaneously. The HPNet consists of three key components: an encoder of historical data, an embedding layer of date features, and three forecasting modules. The encoder takes the partial historical data as the input and derives a high-dimensional latent context vector $\mathbf{c}$. By exploring the hospitalization data, we find that admission patients and discharged patients show a strong correlation with weekdays and holidays. Therefore, we propose the embedding layer to transfer the time information into an embedding vector to improve the forecasting accuracy. The context vector $\mathbf{c}$ and the embedding vector $\mathbf{e}$ are then fed into three forecasting modules simultaneously to forecast the three types of patients, respectively. Furthermore, we proposed a constraint loss to keep the relationship between the three types of patients (expressed as Eqn.(1)).

4.1. Input encoding. First, we project historical data X into a high-dimensional latent context $\mathbf{c}$ by an encoder. To capture different temporal dependencies, Zhang *et al.* (2019) proposed a deep spatiotemporal residual network that selects different key frames along the time. Inspired by this solution, we select recent, periodical data to forecast the patient flows for timestamp t, denoted by X_t^{close} and X_t^{period}, respectively. Recent data represent the numbers of patients during the latest l_c days while periodical data represent the numbers of patients on the same day of weeks in historical data. They are defined as follows:

$$X_t^{\text{close}} = \{X_{t-l_c'}, X_{t-l_c+1}, \ldots, X_{t-1}\},$$
$$X_t^{\text{period}} = \{X_{t-l_p\times 7}, X_{t-(l_p-1)\times 7}, \ldots, X_{t-7}\},$$

where l_c and l_p are the lengths of these two parts of sequences. According to the exploration results of Section 3.2, in our experiments we set $l_c = 7$ and $l_p = 4$.

The encoder consists of three linear layers, followed by a rectified linear unit (ReLu) (Nair and Hinton, 2010) activation function after each layer. Given the input X, the encoder can be formulated as follows:

$$Z_1 = \sigma\left(\mathbf{W}_1 X + \mathbf{b_1}\right), \tag{2}$$
$$Z_2 = \sigma\left(\mathbf{W}_2 Z_1 + \mathbf{b}_2\right), \tag{3}$$
$$\mathbf{c} = \sigma\left(\mathbf{W}_3 Z_2 + \mathbf{b}_3\right), \tag{4}$$

where $\mathbf{W_1} \in \mathbb{R}^{I\times H}, \mathbf{W}_2 \in \mathbb{R}^{H\times 2H}, \mathbf{W}_3 \in \mathbb{R}^{2H\times H}, \mathbf{b}_1 \in \mathbb{R}^H, \mathbf{b}_2 \in \mathbb{R}^{2H}$, and $\mathbf{b}_3 \in \mathbb{R}^H$ are learnable parameters, which can be updated by the gradient decent algorithm. I represents the dimension of inputs while H represents the number of hidden units in the encoder. Finally, the output $\mathbf{c}$ of the encoder is an H-dimensional vector based on the input data. The output $\mathbf{c}$ can be regarded as a deep latent vector containing characteristics of the input sequence.

4.2. Date feature embedding. Admission patients and discharged patients show strong dependency over time (detailed in Section 3.2). For example, there are many admission patients on the weekday while few patients are admitted to the hospital over the weekend. Inspired by this observation, we additionally fuse date features of the forecasting timestamp as an external factor to improve forecasting accuracy. Therefore, we extract two types of date features: (i) the day of the week f_w, (ii) the indicator of holiday f_h. These two features are categorized attributes; we transform them into a low-dimensional vector by feeding them into different embedding layers separately, and then concatenate those embeddings to construct the final categorical vector $\mathbf{e} = [\mathbf{e}_w^T\ \mathbf{e}_f^T]^T$. It is worth noting that in our experiments, the dimensions of embedding results are calculated as

$$d_f = \left\lfloor \frac{\text{cat}_f + 1}{2} \right\rfloor, \tag{5}$$

where cat_f represents the number of categories of features. Therefore, the dimensions of $\mathbf{e}_w$ and $\mathbf{e}_h$ are 4 and 1, respectively.

4.3. Forecasting modules. Once we get the encoding results $\mathbf{c}$ and the embedding vector $\mathbf{e}$, we concatenate them and feed the concatenation vector into three forecasting modules simultaneously to forecast the number of patients. The reason we adopt three separate modules is that the three types of patients may contain different intrinsic relationships which cannot be modeled by a single module. These three forecasting modules are independent but have the same structure. Therefore, for brevity, we take the admission patient forecasting module as an example to illustrate the main idea.

The forecasting module consists of two linear layers that contain H hidden units and one hidden unit, respectively. The input of the admission patient forecasting module is the concatenation of the encoding result $\mathbf{c}$ and the embedding vector $\mathbf{e}$, i.e., $\mathbf{S} = [\mathbf{c}^T\ \mathbf{e}^T]^T$. By using linear layers, the input is compounded to construct a hidden representation, which models the complicated interaction. The output $\hat{y}_a$ of the module forecasts the number of admission patients at time t. The admitted patient forecasting module can be formulated as

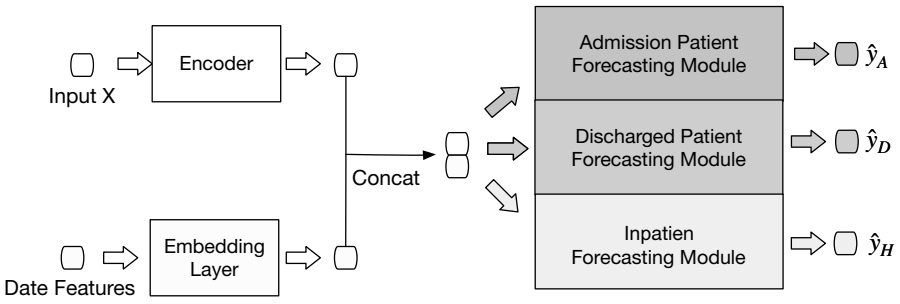


Fig. 6. Framework of HPNet.

follows:

$$Z_A^{(1)} = \sigma\left(\mathbf{W}_{\mathrm{A}}^{(1)}\mathbf{S} + \mathbf{b}_{\mathrm{A}}^{(1)}\right), \tag{6}$$

$$\hat{y}_a = \mathbf{W}_{\mathrm{A}}^{(2)} Z_A^{(1)} + \mathbf{b}_{\mathrm{A}}^{(2)}, \tag{7}$$

where $\mathbf{W}_{\mathrm{A}}^{(1)} \in \mathbb{R}^{(H+5)\times H}$, $\mathbf{b}_{\mathrm{A}}^{(1)} \in \mathbb{R}^{H}$, $\mathbf{W}_{\mathrm{A}}^{(2)} \in \mathbb{R}^{H\times 1}$ $\mathrm{W}^{(2)} \in \mathrm{R}^{\mathrm{H}\times 1}$, $\mathbf{b}_{\mathrm{A}}^{(2)} \in \mathbb{R}$ are learnable parameters. σ is the ReLu activation function. Similarly, we can obtain the forecasting results about discharged patients $\hat{y}_d$ and hospitalized patients $\hat{y}_p$ by other two forecasting modules.

4.4. Losses. Let ϕ comprise all the learnable parameters in the admitted patient forecasting module; we intend to tune them by minimizing the following objective function aggregating the differences between the forecasting results $\hat{y}_a$ and the true numbers y_a:

$$\arg\min_{\phi} \mathrm{Loss}_a = \frac{1}{N}\sum_{i=1}^{N}\left(y_{a_i} - \hat{y}_{a_i}\right)^2, \tag{8}$$

where N represents the number of training records.

Similarly, let θ and ψ be the learnable parameters in the discharged patient forecasting module and the inpatient forecasting module, respectively. They can be updated by minimizing the following objective functions:

$$\arg\min_{\theta} \mathrm{Loss}_d = \frac{1}{N}\sum_{i=1}^{N}\left(y_{d_i} - \hat{y}_{d_i}\right)^2, \tag{9}$$

$$\arg\min_{\psi} \mathrm{Loss}_p = \frac{1}{N}\sum_{i=1}^{N}\left(y_{p_i} - \hat{y}_{p_i}\right)^2. \tag{10}$$

To model the relationship between patients as shown in Eqn. (1), we further construct a constraint loss Loss_c to model the quantitative relationship between the three types of patients. The constraint loss is defined as

$$\mathrm{Loss}_c = \frac{1}{N}\sum_{i=1}^{N}\Big[\underbrace{\left(\hat{y}_{p_i}^{t} - y_{p_i}^{(t-1)}\right)}_{\text{inpatient difference}} - \underbrace{\left(\hat{y}_{a_i}^{t} - \hat{y}_{d_i}^{t}\right)}_{\substack{\text{forecasting patient}\\ \text{transition}}}\Big]^2. \tag{11}$$

Finally, we obtain the combined loss as follows:

$$\mathrm{Loss} = \mathrm{Loss}_a + \mathrm{Loss}_d + \mathrm{Loss}_p + \lambda \mathrm{Loss}_c, \tag{12}$$

where λ is the adjustable hyper-parameter to control the trade off.

5. Experiment

5.1. Settings. Experimental settings are as follows.

Data sets. Table 1 details the two data sets we used, namely ZJFirst and COVID-19. The former has been introduced in Section 3 and the latter is provided by the work of Dong *et al*. (2020). The raw COVID-19 data set contains the numbers of accumulated confirmed patients, accumulated recovery patients and accumulated dead patients. By differencing the data for two consecutive days, we could get the number of newly confirmed patients, newly recovered patients and newly dead patients. Then we calculated the number of currently confirmed patients that is analogous to the number of inpatients in ZJFirst dataset. The sum of newly recovered patients and newly dead patients is analogous to the number of discharged patients while the number of newly confirmed patients is analogous to the number of admission patients.

We divided the whole data set into the training set and test set in a ratio of 8:2. Then we split the training set into five subsets in chronological order, and then employed every unique subset as a validation set, and treated the subsets that occur before the validation set as

Table 1. Data description.

Dataset	ZJFirst	COVID-19
Time span	1/1/2015-12/31/2019	3/26/2020-11/20/2020
Admission patients	[18,1204]	[786,5138]
Discharged patients	[13, 932]	[16,5320]
Inpatients	[890,3068]	[3528,80808]

the training set. In this way, we can adopt five-fold cross validation for time series forecasting to prevent overfitting and evaluate model performance in a more robust way.

Model details. We use the min-max normalization to normalize the number of patients into [0, 1]. In the evaluation, we rescale the predicted values back to normal values. The learning rate is 0.001 and the batch size equals the total numbers of the whole training data set. We leverage the Adam optimizer (Kingma and Ba, 2014) for stochastic gradient descent. We adopt the early stop strategy where training will be terminated when the training loss does not decrease during consecutive 50 epochs. The numbers of hidden units H of the HPNet for two data sets are set to 800 and 2500, respectively, which are determined by a grid search experiment. All of the models are implemented by PyTOrch (Paszke *et al.*, 2019) and are trained on an INVIDIA 3080 GPU.

Evaluation metrics. We use three metrics for evaluation, namely the mean absolute error (MAE), the mean absolute percentage error (MAPE) and the root mean squared error(RMSE). They are defined as follows:

$$\text{MAE} = \frac{1}{n}\sum_{t=1}^{n} \mid y_i - \hat{y}_i \mid, \tag{13}$$

$$\text{MAPE} = \frac{1}{n}\sum_{t=1}^{n} \frac{\mid y_i - \hat{y}_i \mid}{y_i}, \tag{14}$$

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_i - \hat{y}_i)^2}, \tag{15}$$

where y_i and $\hat{y}_i$ are ground truth and the corresponding predicted value, and n is the total number of all available ground truth. Note that MAE and RMSE are more affected by larger values, while MAPE receives more penalty from smaller values. To remove the randomness, we run five times for each method and calculate the average metrics as the final results.

Baselines and variants. We compare the forecasting performance of our model against the following baselines: seasonal ARIMA (SARIMA) (Box *et al.*, 2015), Holt-Winters (Chatfield and Yar, 1988), support vector regression (SVR) (Drucker *et al.*, 1997), random forest (RF) (Liaw and Wiener, 2002), XGBoost (Chen and Guestrin, 2016), multi-layer perceptron (MLP) (Greenwood, 1997), long short term memory network (LSTM) (Neil *et al.*, 2016) and N-Beats (Oreshkin *et al.*, 2019). Besides, we also compare the performance of the HPNet with its two variants to study the effectiveness of different components of the HPNet. Specifically, the first variant is HPNet/od, where date features are excluded. The second variant is denoted as HPNet/oc, where the constraint loss is not considered.

5.2. Results on ZJFirst.

5.2.1. Overall performance comparison. Table 2 displays the experimental results on the ZJFirst data set. We compare the forecasting accuracy of different methods on the three types of patients, and we can conclude the following insights.

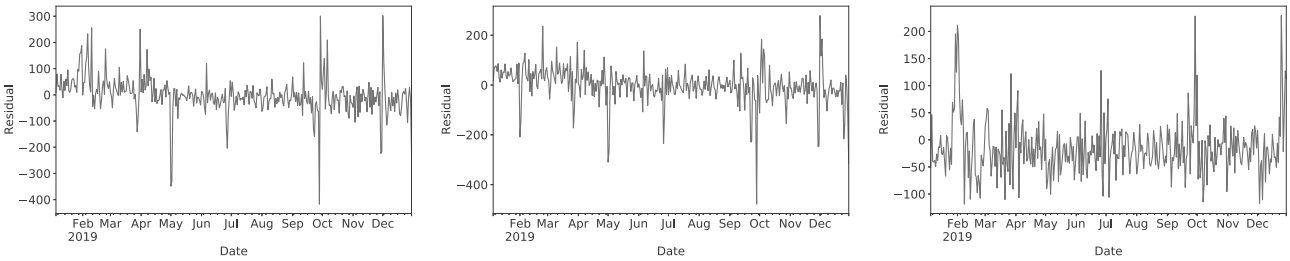
Admission and discharged patient prediction. From the perspectives of admission patients and discharged patients, we can observe that the HPNet achieves the best performance among all methods according to Table 2. For instance, HPNet outperforms the runner-up baseline (MLP) by yielding 31.51% lower MAE, 25.97% lower MAPE, and 26.09% lower RMSE respectively for admission patients, reporting 24.84% lower MAE, 26.22% lower MAPE, and 21.05% lower RMSE, respectively, for discharged patients. This indicates the proposed multi-task architecture is helpful to improve the prediction accuracy of admission patients and discharged patients. Besides, N-Beats is less accurate than MLP and LSTM. The indicates that the proposed input features are helpful for forecasting. Among all baselines, Holt-Winters and SARIMA have the lowest prediction accuracy in terms of MAPE, which indicates they are not suitable for long-term forecasting.

Compared with the two variants, HPNet yields better performance in terms of three metrics from the perspectives of two types of patients. Specifically, HPNet produces 21.7% MAE, 30.5% MAPE and 22.7% RMSE reductions on the average for admission patients, and gives rise to 21.8% MAE, 20.6% MAPE, 28.5% RMSE reductions on the average for discharged patients. This indicates that excluding date features and the constraint loss will deteriorate the capability of HPNet significantly, which proves the effectiveness of the proposed date features and constraint loss.

Inpatient prediction. From the perspective of inpatients, we can find the following observations according to Table 2. First, instead of HPNet, XGBoost achieves the best performance in terms of MAE and MAPE, while HPNet achieves the runner-up performance in terms of MAE and MAPE. On the contrary, in terms of RMSE, HPNet achieves the best performance while the XGBoost achieves the runner-up performance. This indicates that

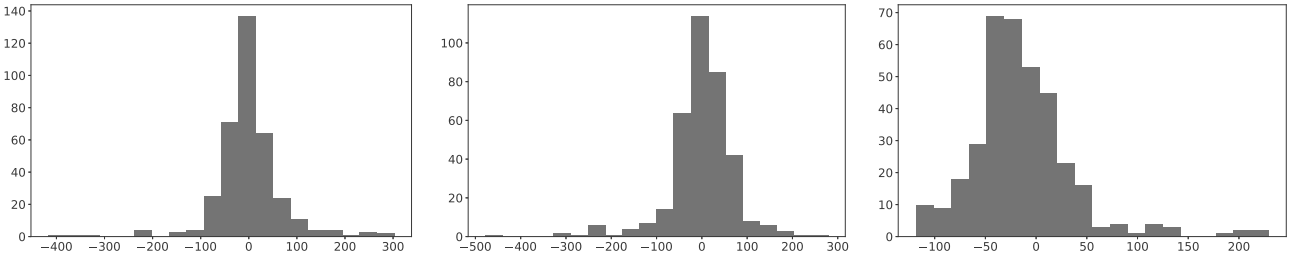
Table 2. Results of comparison on ZJFirst.

Methods	Admission patients			Discharged patients			Inpatients		
	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE
Holt-Winters	62.235	39.481%	119.754	100.394	52.845%	139.023	325.466	15.233%	419.741
SARIMA	62.898	40.534%	116.007	71.734	40.849%	121.434	173.895	8.762%	297.135
SVR	69.054	28.703%	106.311	89.702	34.891%	119.273	38.204	1.633%	66.086
RF	99.800	34.379%	154.530	100.091	30.305%	150.510	39.964	1.684%	58.309
XGBoost	88.556	37.999%	147.204	96.302	30.508%	143.690	**35.450**	**1.513%**	53.253
MLP	61.497	20.180%	93.098	65.459	22.285%	95.697	49.684	2.065%	68.005
LSTM	90.343	39.595%	124.508	82.915	30.926%	119.764	52.356	2.259%	73.118
N-Beats	129.901	42.476%	228.186	118.855	38.894%	188.821	37.980	1.554%	52.406
HPNet/oc	53.799	21.320%	92.546	62.971	20.061%	109.541	50.540	2.146%	68.343
HPNet/od	53.840	21.695%	85.683	62.844	21.371%	102.013	45.354	1.950%	67.978
HPNet	**42.118**	**14.939%**	**68.808**	**49.201**	**16.442%**	**75.550**	36.058	1.571%	**51.663**



(a) admission patients (b) discharged patients (c) inpatients

Fig. 7. Residual results of HPNet.



(a) admission patients (b) discharged patients (c) inpatients

Fig. 8. Histogram of residual results.

HPNet is capable of achieving lower error when the ground truths are abnormal. On the other hand, this also indicates that the HPNet loses its dominating advantage in forecasting accuracy for inpatients. The reason behind this may be that the introduction of the constraint loss will drive the model to learn the quantitative relationship of the three types of patients, thus influencing the model to focus on forecasting the number of inpatients to a certain extent.

Secondly, according to the obtained results, we can observe that the errors of forecasting admission patients and discharged patients are relatively higher than those of inpatients in terms of MAPE. This is because the daily number of admission patients or discharged patients are stochastic. It is influenced by many factors, including weather, healthy status of patients, even their wealth status. Therefore, errors may appear when the forecasting is exclusively based on historical data. Nevertheless, the number of inpatients is relatively stable, where historical data can achieve more accurate forecasting. Besides, due to the large base number, the MAPE of the forecasting of inpatients is small.

Residuals. To check whether HPNet overfits the training dataset, we further study the residuals of forecasting results. Figure 7 presents the residual results along the timeline, and Fig. 8 displays the histogram of the forecasting residual. We can observe that the residuals

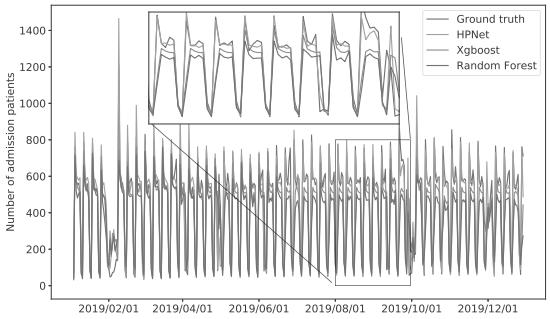
are approximately Gaussian distributed with a mean zero. This implies that HPNet does not overfit the training data set. However, the p-values of the Ljung–Box test are less than 0.05, which indicates that residuals are still correlated with one another. This means that the proposed method HPNet can be further improved, which will be investigated in our future work.

Summary. In this part, we compare the proposed approach with seven baselines and two variants on the ZJFirst data set. From the results shown in Table 2, we can conclude that the HPNet is the best approach for prediction since it shows excellent performance when forecasting numbers of admission patients and discharged patients. Although it does not yield the best performance when forecasting inpatients, it is still competitive (almost 1% MAE higher than the best method). Secondly, the proposed date features and constraint loss are very helpful to improve the forecasting accuracy for all cases. Thirdly, because we combine three forecasting modules into a unified model, we only need to tune hyper-parameters once, which is more convenient than baselines.

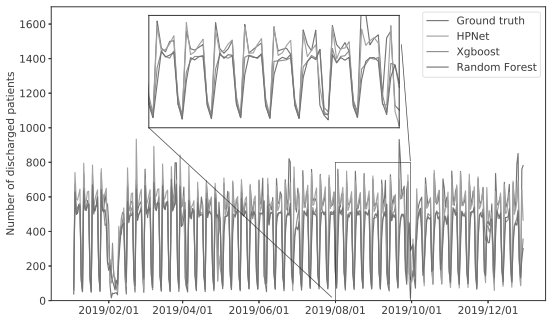
5.2.2. Case study. In the previous section, we have studied the accuracy of the HPNet quantitatively. In this section, we visualize the forecasting results to study the performance of the HPNet qualitatively. The forecasting span is from 2019/01/05 to 2019/12/31. For clarity, we plot the forecasting results of HPNet against the other two competitive baselines, namely random forests and XGBoost. The visualization results can be found in Fig. 9. To study the obvious differences between different approaches, we then zoom in on the forecasting curves from 2019/08/01 to 2019/08/31, which can be seen in the black box of each figure.

We can find that all these methods can capture the trends of patients approximately. However, we can observe the superiority of the HPNet when we look inside the forecasting results of the admission patients and discharged patients. As we can see from Fig. 9(a) and (b), although these methods can be more accurate in the case of a low number of patients, they will produce obvious differences when the number of patients is large. Specifically, the forecasting results of HPNet are closer to the ground truth. This accounts for the superiority of the HPNet that it can maintain better accuracy when predicting high traffic of patients. As for inpatients, three methods are competitive along time.

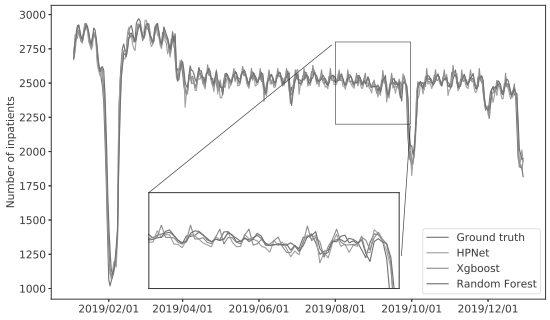
5.3. Results on COVID-19. Experimental results on the second data set have demonstrated the superiority of our framework again. From Table 3, we can observe that HPNet shows significant improvements against baselines on the COVID-19 data set, validating its great generality in different applications. Specifically, HPNet yields



(a) admission patients



(b) discharged patients



(c) hospitalized patients

Fig. 9. Visualization of forecasting results.

10.07%, 8.03%, 39.69% improvements compared with the runner-up method in terms of MAE, MAPE and RMSE, respectively, for admission patients. HPNet produces 1.33%, 6.53%, 3.15% improvements compared with the runner-up method in terms of MAE, MAPE and RMSE, respectively, for discharged patients. HPNet gives rise to 6.54%, 9.39%, 1.31% improvements in comparison with the runner-up method in terms of MAE, MAPE and RMSE, respectively, for inpatients.

6. Conclusions

In this paper, we propose an end-to-end neural network based forecasting model to forecast the number of hospitalization patients. In contrast to the previous works, we forecast three types of patients simultaneously by

Table 3. Comparison of results on COVID-19.

Methods	Admission patients			Discharged patients			Inpatients		
	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE
Holt-Winters	791.091	27.184%	1130.137	638.303	31.852%	750.628	15148.708	37.498%	15361.337
SARIMA	501.626	16.241%	831.765	329.607	14.340%	508.230	9027.069	19.774%	11835.930
SVR	693.875	23.658%	992.065	763.715	44.308%	781.960	1937.368	4.556%	2129.506
RF	745.472	25.669%	1054.149	201.520	10.776%	241.592	3554.312	8.846%	3845.745
Xgboost	577.045	19.871%	820.461	171.334	8.548%	**230.681**	8138.861	18.501%	9476.800
MLP	995.279	36.432%	1245.147	336.899	20.238%	362.518	2863.093	6.533%	3379.343
LSTM	709.742	24.746%	974.778	199.448	9.698%	295.463	2289.664	5.051%	2962.883
N-Bats	354.512	14.552%	571.722	311.038	16.220%	391.788	948.690	2.280%	**1039.345**
HPNet	**318.795**	**13.383%**	**344.815**	**169.048**	**7.990%**	237.945	**886.691**	**2.066%**	1052.908

using a constraint loss to study their relationship. They are admission patients, discharged patients and inpatients. We propose a multi-task strategy and a constraint loss to improve the forecasting accuracy. The experimental results over two real-world data sets demonstrate the superiority of our approach.

In the future, we will explore more effects affecting the patient flows to strengthen our model. At present, we only use historical data of patients to train our model and forecast the results, without considering many external factors, like weather, breaking news and so on. In the future, we will try more external factors to improve the accuracy further. Second, the structure of the three forecasting modules can be improved by introducing more complex but efficient structures, like an attention mechanism, convolution neural networks and so on. We will explore more promising structures to improve the performance of the model. Besides, we will try to apply our method to other scenarios, such as forecasting patient flows for emergency departments.

References

Bergs, J., Heerinckx, P. and Verelst, S. (2014). Knowing what to expect, forecasting monthly emergency department visits: A time-series analysis, *International Emergency Nursing* **22**(2): 112–115.

Bingi, K. and Prusty, B.R. (2021). Forecasting models for chaotic fractional-order oscillators using neural networks, *International Journal of Applied Mathematics and Computer Science* **31**(3): 387–398, DOI: 10.34768/amcs-2021-0026.

Box, G.E., Jenkins, G.M., Reinsel, G.C. and Ljung, G.M. (2015). *Time Series Analysis: Forecasting and Control*, Wiley, Hoboken.

Chatfield, C. and Yar, M. (1988). Holt–Winters forecasting: Some practical issues, *Journal of the Royal Statistical Society: Series D (The Statistician)* **37**(2): 129–140.

Chen, T. and Guestrin, C. (2016). XGBoost: A scalable tree boosting system, *International Conference on Knowledge Discovery and Data Mining, San Francisco, USA*, pp. 785–794.

Chen, Z.-M., Wei, X.-S., Wang, P. and Guo, Y. (2019). Multi-label image recognition with graph convolutional networks, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, USA*, pp. 5177–5186.

Cheng, Y. (2019). *Joint Training for Neural Machine Translation*, Springer theses, Springer, Singapore, chapter "Semi-supervised learning for neural machine translation", pp. 25–40.

Cross, S.S., Harrison, R.F. and Kennedy, R.L. (1995). Introduction to neural networks, *The Lancet* **346**(8982): 1075–1079.

Dai, J., He, K. and Sun, J. (2016). Instance-aware semantic segmentation via multi-task network cascades, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, USA*, pp. 3150–3158.

Dong, E., Du, H. and Gardner, L. (2020). An interactive web-based dashboard to track COVID-19 in real time, *The Lancet Infectious Diseases* **20**(5): 533–534.

Drucker, H., Burges, C.J., Kaufman, L., Smola, A., Vapnik, V. (1997). Support vector regression machines, *Advances in Neural Information Processing Systems* **9**: 155–161.

Greenwood, G.W. (1997). Training multiple-layer perceptrons to recognize attractors, *IEEE Transactions on Evolutionary Computation* **1**(4): 244–248.

Hadavandi, E., Shavandi, H., Ghanbari, A. and Abbasian-Naghneh, S. (2012). Developing a hybrid artificial intelligence model for outpatient visits forecasting in hospitals, *Applied Soft Computing* **12**(2): 700–711.

Hou, C., Wu, J., Cao, B. and Fan, J. (2021). A deep-learning prediction model for imbalanced time series data forecasting, *Big Data Mining and Analytics* **4**(4): 266–278.

Jiang, S., Chin, K.-S. and Tsui, K.L. (2018). A universal deep learning approach for modeling the flow of patients under different severities, *Computer Methods and Programs in Biomedicine* **154**: 191–203.

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

Kowal, M., Skobel, M., Gramacki, A. and Korbicz, J. (2021). Breast cancer nuclei segmentation and classification based

on a deep learning approach, *International Journal of Applied Mathematics and Computer Science* **31**(1): 85–106, DOI: 10.34768/amcs-2021-0007.

Ledersnaider, D.L. and Channon, B.S. (1998). Sdm95—Reducing aggregate care team costs through optimal patient placement, *JONA: The Journal of Nursing Administration* **28**(10): 48–54.

Li, Y., Ki, Y., Sun, N., Wu, F., Zheng, C., Sun, N. (2015). Predictive analysis of outpatient visits to a grade 3, class A hospital using ARIMA model, *Proceedings of the 2014 International Symposium on Information Technology (ISIT 2014), Dalian, China*, pp. 301–304.

Liaw, A. and Wiener, M. (2002). Classification and regression by randomForest, *R News* **2**(3): 18–22.

Luo, L., Luo, L., Zhang, X. and He, X. (2017). Hospital daily outpatient visits forecasting using a combinatorial model based on ARIMA and SES models, *BMC Health Services Research* **17**(1): 1–13.

Mackay, M. (2001). Practical experience with bed occupancy management and planning systems: An Australian view, *Health Care Management Science* **4**(1): 47–56.

Makridakis, S., Spiliotis, E. and Assimakopoulos, V. (2018). Statistical and machine learning forecasting methods: Concerns and ways forward, *PloS One* **13**(3): 1–26.

Menke, N.B., Caputo, N., Fraser, R., Haber, J., Shields, C. and Menke, M.N. (2014). A retrospective analysis of the utility of an artificial neural network to predicted volume, *The American Journal of Emergency Medicine* **32**(6): 614–617.

Nair, V. and Hinton, G.E. (2010). Rectified linear units improve restricted Boltzmann machines, *Proceedings of the 27th International Conference on Machine Learning, Haifa, Israel*, pp. 807–814.

Nassif, A.B., Shahin, I., Attili, I., Azzeh, M. and Shaalan, K. (2019). Speech recognition using deep neural networks: A systematic review, *IEEE Access* **7**: 19143–19165.

Neil, D., Pfeiffer, M. and Liu, S.-C. (2016). Phased LSTM: Accelerating recurrent network training for long or event-based sequences, *International Conference on Neural Information Processing Systems, Barcelona, Spain*, pp. 3889–3897.

Oreshkin, B.N., Carpov, D., Chapados, N. and Bengio, Y. (2019). N-BEATS: Neural basis expansion analysis for interpretable time series forecasting, *arXiv:* 1905.10437.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J. and Chintala, S. (2019). PyTOrch: An imperative style, high-performance deep learning library, *Advances in Neural Information Processing Systems* **32**: 8026–8037.

Sanh, V., Wolf, T. and Ruder, S. (2019). A hierarchical multi-task approach for learning embeddings from semantic tasks, *Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, USA*, pp. 6949–6956.

Sun, Y., Heng, B.H., Seow, Y.T. and Seow, E. (2009). Forecasting daily attendances at an emergency department to aid resource planning, *BMC Emergency Medicine* **9**(1): 1–9.

Wang, Y. and Gu, J. (2014). Hybridization of support vector regression and firefly algorithm for diarrhoeal outpatient visits forecasting, *IEEE International Conference on Tools with Artificial Intelligence, Limassol, Cyprus*, pp. 70–74.

Zhang, J., Zheng, Y., Sun, J. and Qi, D. (2019). Flow prediction in spatio-temporal networks based on multitask deep learning, *IEEE Transactions on Knowledge and Data Engineering* **32**(3): 468–478.

Zhang, Y. and Yang, Q. (2021). A survey on multi-task learning, *IEEE Transactions on Knowledge & Data Engineering* (0**34**(12): 5586–5609.

Min Zhou received his BS and MS degrees in biomedical engineering from Zhejiang University, China, in 1993 and 1999, respectively. He joined the First Affiliated Hospital of the Zhejiang University School of Medicine, Hangzhou, China, in 1993. He has published more than 10 papers related to medical information. His current research interests include medical image analysis and processing, biomedical signal analysis.

Xiaoxiao Huang received her MS degree in computer science and technology from the Zhejiang University of Technology, China, in 2019. She joined the First Affiliated Hospital of the Zhejiang University School of Medicine, Hangzhou, China, in 2019. Her current research interests include medical image analysis and processing, biomedical signal analysis.

Haipeng Liu received his BS and MS degrees in electrical engineering and biomedical engineering from Zhejiang University, Hangzhou, China, in 2012 and 2015, respectively, and his PhD degree in medical sciences from the Chinese University of Hong Kong, in 2018. From 2019 to 2020, he was a research fellow with the Medical Technology and Research Center, Anglia Ruskin University. Since 2020, he has been a research fellow with Coventry University, UK. He is the author of 51 journal articles and seven IEEE conference papers. His research interests include biomechanics, physiological measurement, and simulation of cardiovascular diseases.

Dingchang Zheng received his BS degree in biomedical engineering from Zhejiang University, China, and his PhD degree in medical physics from Newcastle University, UK. He is currently a professor of healthcare technology and a research theme leader with Coventry University. He is the winner of the IPEM Martin Black Annual Prize (2011) and the Institution of Engineering and Technology (IET) JA Lodge Award 2009 for Recognizing and Promoting Outstanding Work in the Field of Research and Development in Medical Engineering.

Received: 23 January 2022
Revised: 16 May 2022
Re-revised: 9 August 2022
Accepted: 9 August 2022