Blood Pressure Prediction via Recurrent Models with Contextual Layer

Xiaohan Li, Shu Wu, Liang Wang
Center for Research on Intelligent Perception and Computing
National Laboratory of Pattern Recognition
Institute of Automation, Chinese Academy of Sciences
shawnlxh@outlook.com, {shu.wu, wangliang}@nlpr.ia.ac.cn

ABSTRACT

Recently, the percentage of people with hypertension is increasing, and this phenomenon is widely concerned. At the same time, wireless home Blood Pressure (BP) monitors become accessible in people's life. Since machine learning methods have made important contributions in different fields, many researchers have tried to employ them in dealing with medical problems. However, the existing studies for BP prediction are all based on clinical data with short time ranges. Besides, there do not exist works which can jointly make use of historical measurement data (e.g. BP and heart rate) and contextual data (e.g. age, gender, BMI and altitude). Recurrent Neural Networks (RNNs), especially those using Long Short-Term Memory (LSTM) units, can capture long range dependencies, so they are effective in modeling variable-length sequences. In this paper, we propose a novel model named recurrent models with contextual layer, which can model the sequential measurement data and contextual data simultaneously to predict the trend of users' BP. We conduct our experiments on the BP data set collected from a type of wireless home BP monitors, and experimental results show that the proposed models outperform several competitive compared methods.

1. INTRODUCTION

In recent years, the population of people with hypertension is rising, and this phenomenon becomes a major global public health issue. According to World Health Organization (WHO) [27], more than 20% adults worldwide have hypertension till 2015. Latest studies have shown that patients with hypertension have a high risk of many diseases, such as stroke, heart failure, damage to eyes, etc [5, 8]. It is necessary for patients to continuously observate their blood pressure (BP). Hypertension guidelines [18] have recommended the application of self-monitoring of BP in clinical practice. Right now, about 70% of patients with hypertension regularly monitor their BP at home using the self-monitoring devices [24]. Home BP monitoring is widely regarded as an important complement of conventional office BP monitoring in the hypertension management [25].

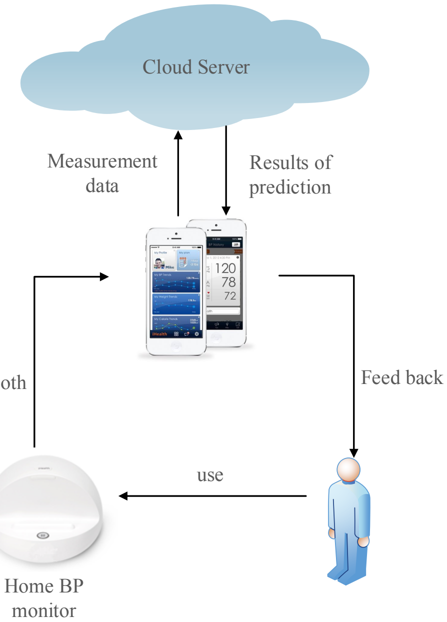


Figure 1: Overview of the workflow of wireless home BP monitors.

Conventional office BP is usually measured by the physician clinically. Compared to office BP monitoring, home BP monitoring can provide more controlled and more regular supervision. The use of home BP monitoring has been shown to improve patients' compliance [3] and avoid some biases including white-coat hypertension [33]. By connecting to smart phones, it provides a large amount of measurement data, reflecting users' recent physical condition. The data can be used to model the variation of users' BP and make the prediction of future BP.

The data set used in this paper is collected from a type of wireless home BP monitors. The workflow of the device is shown as Figure 1. The monitors synchronize the measurement data to the cloud server via smart phones, and the server trains the BP prediction model with the historical data. Users can get results of prediction made by the server in real time. On the cloud platform, it is easy to access a

WWW 2017, April 3–7, 2017, Perth, Australia.
ACM 978-1-4503-4913-0/17/04.
http://dx.doi.org/10.1145

great amount of users' measurement data, which is suitable to be analyzed by machine learning methods.

Recently, machine learning methods have been employed in dealing with numerous medical problems. Some researchers use machine learning methods to predict the risk of stroke [9], coronary heart disease [34] with office BP measurement data. However, none of them take advantage of sequential information. Some works [35] make prediction of BP based on contextual information of users (e.g. BMI, age, smoke level). Other works [30] use continuous BP measurements in the Intensive Care Unit (ICU) to predict BP. But all these works on BP prediction have difficulty in modeling contextual information and sequential measurements simultaneously. Moreover, none of these works use the data from home BP monitoring.

Support Vector Regression (SVR) [31], Gradient Boosted Regression Tree (GBRT) [10], Factorization Machine (FM) [29] and Multilayer Perceptron (MLP) are all classic regression models, and can be applied to BP prediction problem. Though they perform numerical prediction efficiently, they have two limitations. One is that they must use fixed-length features; the other is that they cannot take full advantage of the sequential dependency. Hidden Markov Model (HMM) is a probabilistic graphical model. It is often used for sequential prediction. However, this method can only combine several most recent components. To solve this problem, Recurrent Neural Network (RNN) and its variants, LSTM [17] and GRU [6] have been successfully employed to model temporal dependency for different applications, such as speech recognition [15], video modeling [4], sequential click prediction [36], location prediction [23] and recommendation system [22]. They can capture long time dependency and nonlinear dynamics. RNNs led to much improvement in all the tasks above.

In this paper, to predict the blood pressure, we propose recurrent models with contextual layer, called RNN with Contextual Layer (RNN-CL) and LSTM with Contextual Layer (LSTM-CL). Users' historical measurement data (e.g. heart rate, BP) is arranged in chronological order. This type of data is used as the input of the hidden layer in the recurrent structure. In this way, the dependency among blood pressure measurements will be embedded into the recurrent structure. Besides, contextual data (e.g. age, gender, BMI and altitude) is helpful for making more personalized and accurate prediction of BP. For example, a user with a high Body Mass Index (BMI) value tends to have high BP [5]. Therefore, we add an extra layer to deal with contextual data before the output layer. The activation function of the extra layer is Rectified Linear Unit (ReLU), which can avoid vanishing gradient problem. Finally, we combine the features extracted from sequential measurement data with contextual data to achieve a joint representation as the final users' BP condition. This representation is used to make prediction of the accurate BP value. For the situation that the users' contextual data are vacant, we fill up the vacant ones by several imputation strategies. The learning algorithm of our models is a variant of back propagation through time algorithm. It is used to learn parameters. Our contributions are listed as follows:

- We introduce the recurrent models with contextual layer to make numerical prediction of users' BP. The recurrent structure processes variable-length sequences of users' historical measurements, and an extra layer extracts the features from contextual data. We combine these two structures for better inferring users' future BP.
- We conduct experiments on a data set based on the data collected from a type of wireless BP monitors. The experiments reveal that our model can significantly improve the accuracy of BP prediction. Moreover, the results show that KNN is the most suitable method to fill up missing contextual data.

The rest of the paper is organized as follows. In section 2, we review some related work on numerical prediction models and machine learning applications for medical problems. Section 3 details the recurrent models with contextual layer and the learning algorithm. In section 4 we introduce the data we use in our experiments and the imputation strategies for filling up the missing data. In section 5, we report and analyze experimental results. Section 6 concludes our work and discusses future research.

2. RELATED WORKS

In this section, we briefly review several related works on classic regression and sequential prediction methods. Then we introduce some applications of machine learning for medical treatment, especially the works on BP.

2.1 Numerical Prediction Models

Linear Regression is the basic regression model. Its loss is calculated by the difference between the output from the model and the target. Only when the output is exactly equivalent to the target can the loss be equal to zero. Support Vector Regression (SVR) [31] is the application of Support Vector Machine (SVM) for the case of regression. It is efficient and well-performed, and is usually considered as a baseline. Gradient Boosted Regression Trees (GBRT) [10] is typically used with decision trees of a fixed size as base learners and optimized by a boosting technique. GBRT is an accurate and effective off-the-shelf procedure, so it is often used in a variety of areas such as web search ranking and click through rate prediction. Multilayer Perceptron (MLP), which usually consists of three layers, is the most basic artificial neural network. If the output layer of MLP is linear regression, it can make numerical prediction. However, since the input of these methods must be fixed-length and they don't have sequential structure, these methods have difficulty in capturing the sequential dependency of the variable-length data.

Considering the limitation of the methods above, some methods based on sequential structure are more suitable for our task. Hidden Markov Model (HMM), which is known for the application in speech recognition [28], is designed for sequential prediction. However, it is only suitable for classification, but not for numerical prediction. Recurrent Neural Network (RNN) is an effective approach to sequential prediction. Unlike the feedforward neural network, RNN can use their memory information to process sequences of inputs. Now many studies leverage RNN to model the temporal dependency within the data, and most of them get the state-of-the-art results on many practical tasks. For example, RNN language model [26] take full advantage of long-span sequential information to handle the massive language corpus, and they achieve a better results than traditional neural networks language model. Moreover, RNN also brings much improvement in speech recognition [20], and machine translation [1]. LSTM is a variation of RNN. It replaces the activation function with the LSTM unit, which can process time series with very long time lags of unknown size between important events. It outperforms traditional RNN in numerous applications, such as handwriting recognition [14] and speech recognition [15].

2.2 Machine Learning for Medical Data

In the recent works, various machine learning methods are applied for the medical problems. Aleks et al. use dynamic Bayesian network to remove the data artifacts [11]. Somanchi et al. make early prediction of cardiac arrest with SVM [32]. Recently neural networks have also been employed in this field. Dabek et al. use artificial neural network to predict psychological conditions such as anxiety, behavioral disorders and depression [7]. Hammerla et al. assess Parkinson's disease with deep belief network based on data collected in naturalistic settings [16]. Lipton et al. use LSTM to diagnose with clinical medical data [21]. They recognize patterns in variable-length time series of clinical measurements to make the multilabel classification of diagnoses.

Furthermore, some researchers have tried to take full advantage of BP measurement data. They employ the machine learning and statistics methods to analyze this data on many tasks, for instance, assessing the risk of stroke with logistic regression model [9], predicting coronary heart disease with cox proportional hazards regression model [34], etc. Researchers also attempt to predict accurate BP value based on decision tree [13] and neural network model [35]. Sideris et al. predict BP with recurrent neural network using measurement data from patients in ICU [30]. However, these works are all based on office BP monitoring data, and none of them jointly use the measurement data and contextual data to make the prediction.

3. RECURRENT MODELS WITH CONTEXTUAL LAYER

In this section, we introduce our proposed recurrent models with contextual layer. Firstly we introduce the notations, then present the proposed models thoroughly at first, and finally demonstrate the process of training the models.

3.1 Problem Formation

The parameter t represents the time step in the sequence. In our data set, at time t for each user the measurement instance (e.g. BP and heart rate) is denoted by $x(t)$ and user's contextual data (e.g. age, gender, BMI and altitude) denoted by $u(t)$. Given the measurement instances in time order and user's contextual data, we can train the models to predict the user's next BP measurement y. The overall notations are shown in the Table 1.

Table 1: Notations

Notation	Explanation
t	the step in the sequence
$x(t)$, $u(t)$	measurement data, contextual data
$y(t)$, $\hat{y}(t)$	real and predicted BP value
$i(t)$,$f(t)$,$o(t)$	input, forget and output gate of LSTM unit
$b_i, b_f, b_o, b_{c\_in}, b$	bias
σ	the activation function sigmoid
$c\_in(t)$	the input of LSTM cell
$c(t)$	the current state of LSTM cell
$h_1(t)$, $h_2(t)$	the first and the second hidden layer
$e_o(t)$, e_h	the gradient of output layer and $h_2(t)$
$\mathbf{V}(t)$, $\mathbf{T}(t)$, $\mathbf{P}(t)$	the weights of output layer, $h_2(t)$ and $h_1(t)$
α	the learning rate

3.2 Recurrent Neural Network (RNN)

Recurrent neural network (RNN) is a kind of artificial neural network. Unlike feedforward neural network, its connections between units form a directed cycle. Specifically, RNN trains one input vector at a time. In time step t, the hidden layer $h_1(t-1)$ in time step $t-1$ will be combined with the input $x(t)$, and then they jointly connect to the hidden layer $h_1(t)$. The formulation of each hidden layer in RNN is:

$$h_1(t) = tanh(\mathbf{P}x(t) + \mathbf{W}h_1(t-1)), \tag{1}$$

where $\mathbf{P}$ and $\mathbf{W}$ are the weights of the recurrent structure. This recurrent structure enables RNN to use their internal memory to process inputs with arbitrary lengths. However, because of the vanishing gradient problem, the range of input that can be in practice accessed by standard RNN is quite limited.

3.3 Long Short-Term Memory (LSTM)

Unlike traditional RNN, Long Short-Term Memory (LSTM) replaces the activation function of the neurons to a unit with an ingenious inner structure called LSTM [17]. LSTM doesn't have the vanishing gradient problem, and can store the memory of input thousands of discrete time steps before. The LSTM unit in our paper uses memory cells with forget gates [12]. The following equations represent the process of parameter update.

$$i(t) = \sigma(\mathbf{W}_{xi}x(t) + \mathbf{W}_{hi}h_1(t-1) + b_i), \tag{2}$$

$$f(t) = \sigma(\mathbf{W}_{xf}x(t) + \mathbf{W}_{hf}h_1(t-1) + b_f), \tag{3}$$

$$o(t) = \sigma(\mathbf{W}_{xo}x(t) + \mathbf{W}_{ho}h_1(t-1) + b_o), \tag{4}$$

$$c\_in(t) = tanh(\mathbf{W}_{xc}x(t) + \mathbf{W}_{hc}h_1(t-1) + b_{c\_in}), \tag{5}$$

$$c(t) = f(t) \cdot c(t-1) + i(t) \cdot c\_in(t), \tag{6}$$

$$h_1(t) = o(t) \cdot tanh(c(t)). \tag{7}$$

In these equations, σ represents the sigmoid function. $h_1(t-1)$ stands for the previous output of LSTM unit. We denote the input, forget, output gates respectively as i, f, o, and c_in is the input of the LSTM cell. $tanh$ is the activation function of c_in. The cell's state transition is shown in the Formula (6), and the current state is $c(t)$ that is calculated by the previous state $c(t-1)$ and the gates in the LSTM unit. $h_1(t)$ in the Formula (7) is the output of LSTM unit at the current time step t.

3.4 Recurrent Models with Contextual Layer

Our proposed recurrent models with contextual layer simultaneously utilize both the historical measurement data and contextual data to make the prediction of users' BP. In traditonal sequential prediction methods, all the inputs are sequential and can be arranged in time order. However, in our BP data set, the contextual data is mostly filled by users themselves, and it will be constant since then. Therefore, we view this data as a supplement to our results. We add an extra layer to process the contextual data individually. The recurrent structure models the features of sequential data, and then the recurrent structure is combined with contextual layer to form a joint representation. With this representation, we can make the prediction of the future BP value.

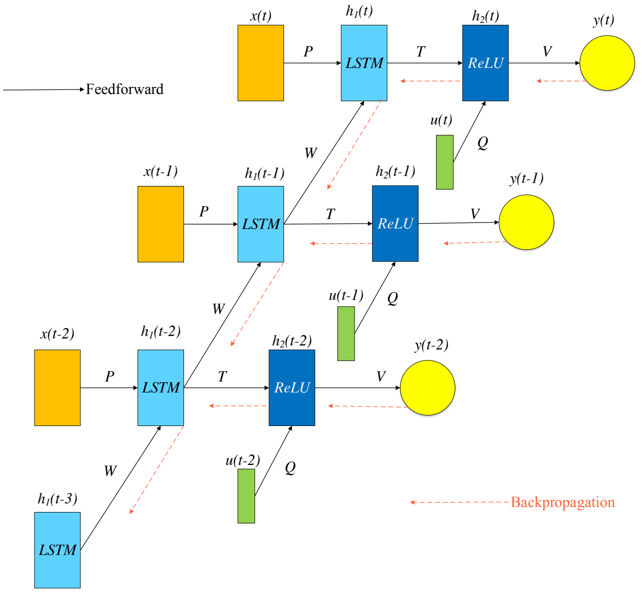


Figure 2: Overview of our LSTM-CL model.

RNN/LSTM with Contextual Layer. As shown in Figure 2, in our proposed model, we add an extra hidden layer $h_2(t)$ after the general hidden layer $h_1(t)$ to process the contextual data. This model is named LSTM with Contextual Layer (LSTM-CL). Contextual data are combined with the output of hidden layer $h_1(t)$ collectively as input connecting to the other hidden layer $h_2(t)$. In Figure 2, the full line represents the process of feedforward, and the dotted line stands for the backpropagation through time. RNN-CL is similar to LSTM-CL, just replacing the LSTM unit with the *tanh* function.

In LSTM-CL, we take features $h_1(t)$ extracted from the historical measurement data by the first hidden layer and contextual data $u(t)$ together as the input of the next hidden layer $h_2(t)$. The description of this process is

$$h_2(t) = ReLU(h_1(t)\mathbf{T} + u(t)\mathbf{Q}), \tag{8}$$

where $ReLU(x) = max(0, x)$ is the $ReLU$ function that we use for non-linear activation. We choose $ReLU$ because if we choose *tanh* or *sigmoid*, the vanishing gradient problem [19] will stymie the gradient propagating from the second hidden layer to the first hidden layer. $h_2(t)$ is the joint representation of contextual data and measurement data. Then we apply linear regression to get the result of prediction $\hat{y}(t)$. $\hat{y}(t)$ is calculated as

$$\hat{y}(t) = h_2(t)\mathbf{V} + b, \tag{9}$$

where $\mathbf{V}$ is the parameters of linear regression and b is the bias. Note that, we add L2 penalty of weights $\mathbf{V}$ to the model in the practical experiments. In this way, the formulas above will be modified slightly to avoid overfitting.

Loss function. In the last layer of our model, we choose linear regression to accomplish numeric prediction. The loss function we use is quadratic loss function, whose main idea is to minimize the error between the target and the output of the model.

$$loss(\hat{y}, y) = \frac{1}{2}(y(t) - \hat{y}(t))^2 \tag{10}$$

The 1/2 in the formula above is to simplify the process of taking partial derivation of the loss.

3.5 Learning Algorithm

In this section, we introduce the learning process of our proposed RNN-CL and LSTM-CL models with Back Propagation Through Time (BPTT).

The main idea of BPTT is that the state of hidden layer from previous time step is simply regarded as an additional input. In this way, we can unfold the network to a deep neural network, and then backpropagation can be applied to train this model. Figure 2 shows the deep neural network unfolded through time.

The learning process is shown as follows. First we need to calculate the errors of the output layer. From the loss function we proposed before, we can compute the errors of the linear regression as

$$e_o(t) = y(t) - \hat{y}(t), \tag{11}$$

and $e_o(t)$ is what we need to propagate to other layers to modify parameters. Then the weights V between the output and the second hidden layer can be updated as

$$\mathbf{V}(t+1) = \mathbf{V}(t) - \alpha \cdot e_o(t) \cdot h_2(t), \tag{12}$$

where α in the formula above is the learning rate. Then we can get errors propagated from the output layer to the second hidden layer as

$$e_{h2}(t) = e_o(t)\mathbf{V}^{\mathrm{T}} * \begin{cases} 0 & x < 0 \\ \vec{1} & x > 0 \end{cases}, \tag{13}$$

where * is the element-wise product, and $\vec{1}$ is the vector that all elements equal to one. After calculating the $e_{h2}(t)$, we can easily modify the weights Q as

$$\mathbf{Q}(t+1) = \mathbf{Q}(t) - \alpha \cdot u(t)^{\mathrm{T}} \cdot e_{h2}(t) \tag{14}$$

where u represents the contextual data. After computing the weights Q, we can update weights T between the two hidden layers. This process is similar to updating Q as

$$\mathbf{T}(t+1) = \mathbf{T}(t) - \alpha \cdot h_2(t)^{\mathrm{T}} \cdot e_{h2}(t). \tag{15}$$

Finally, we can get gradients of errors propagated from the output layer to the first hidden layer as

$$e_{h1}(t) = e_{h2}(t)\mathbf{T}(t), \tag{16}$$

and the following process is same as traditional BPTT.

4. DATA

4.1 Data Description

As shown in Figure 1, our data is collected from the cloud server of the wireless home BP monitors. The data is composed of two parts: *contextual data* including users' profile

Table 2: Data Description

Measurement Data	Description
Systolic BP, diastolic BP	Continuous Variable (in mm Hg)
Heart rate	Continuous Variable (in bpm)
month of the measurement	One-hot Vector
Whether takeing drugs	Binary Variable (Yes, No)
Contextual Data	**Description**
BMI	Continuous Variable (in kg/m2)
Gender	Binary Variable (Male, Female)
Age	Continuous Variable (in year)
Altitude	Continuous Variable (in meter)
Longitude and Latitude	Continuous Variable (in degree)

initialized by users when they register the accounts, and locations recorded by GPS; *measurement data*, including BP, heart rate and so on, collected by the BP monitors. There are 6 variables we can obtain in *contextual data*: BMI, gender, age, altitude, longitude and latitude of a user u. *Measurement data* consists of 5 variables: systolic BP, diastolic BP, heart rate, time of the measurement, whether taking drugs or not. We also add a feature about the interval between two measurements. Details of data is provided in Table 2.

The basic statistics of our data are summarized as follows. 50.4% users are between 44 and 54 years old. 59.3% users are male. As for the BP measurement, 52% users tend to measure their BP from 6 a.m. to 8 a.m. and from 7 p.m. to 10 p.m.. User's condition on BP is divided into six categories, which is listed in Table 3. We find that the largest proportion of users are in stage 1 hypertension. Patients with stage 1 hypertension care most about their BP condition. By predicting the trend of BP, we can help them control their BP. We also analyse the relationship between BMI and BP. We divide users by their BMI, and then we calculate the average BP in each group. The results are shown in Figure 3. From Figure 3 we can see, it is an obvious tendency that BP rises with the increase of BMI.

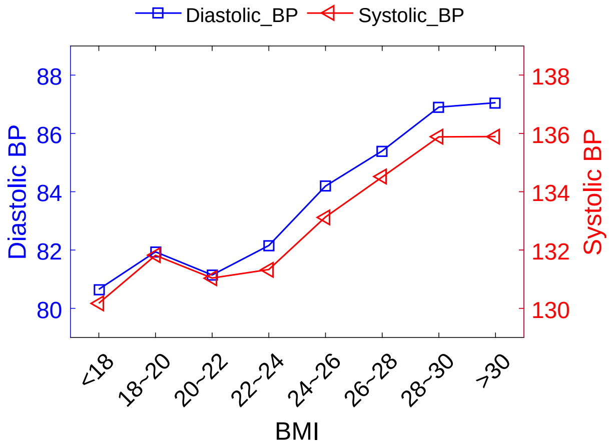


Figure 3: The relationship bwtween BMI and BP. From the picture we can see, BP rises with the increase of BMI.

4.2 Dealing with Missing data

Table 3: Classification of blood pressure for adults

Category	Systolic BP (mm Hg)	Diastolic BP (mm Hg)	Percentage (%)
Perfect	90 - 119	60 - 79	20.7
Normal	120 - 129	80 - 84	18.1
High normal	130 - 139	85 - 89	19.8
Stage 1 hypertension	140 - 159	90 - 99	**28.8**
Stage 2 hypertension	160 - 179	100 - 109	9.6
Stage 3 hypertension	$\geqslant$ 180	$\geqslant$ 110	3.0

In our task, when users register their accounts, they will also fill their contextual data. However, some users are not willing to share their personal information to the Internet, so their contextual data is vacant. In statistics, imputation is a common approach to deal with missing data. Imputation is that missing data is replaced with substituted values. Once all missing values have been imputed, the dataset can then be analysed with standard methods for complete data. Mean imputation, regression imputation and KNN imputation [2] are applied in our paper.

Mean imputation is that any missing value is replaced with the mean of that features in all other cases. This is one of the simplest imputation methods.

Regression imputation is that the missing value is predicted by a regression model. The regression model is employed to predict a feature based on other features. In our paper, the input instance of the regression model is a nine-dimentional vector m which consists of the systolic BP, diastolic BP and heart rate in the last three months. We can get the approximate contextual data $\hat{u}$ by the following formula:

$$\hat{u} = \mathbf{R}m + b, \tag{17}$$

where $\mathbf{R}$ is the parameter to be learned by gradient descent during the training process.

KNN imputation is the use of K Nearest Neighbour (KNN) algorithm to estimate missing data. The main idea of KNN is to find the k nearest samples in feature space.

The input instance of KNN imputation is the same as regression imputation. We choose the k closest users who have complete contextual data in the feature space as neighbors, and then we fill the user's contextual data with the most common neighbor among them. The distance metric we use is Euclidean distance, which can be computed as

$$dist(x_i, y_i) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}, \tag{18}$$

where x_i, y_i represent the features, and smaller distance indicates higher similarity.

In experiments section, we will compare the performances of these strategies in our data set.

5. EXPERIMENTS

In this section, first we introduce the settings of our experiments, and then report the experiment results with further analysis.

5.1 Data Preprocessing

To avoid some biases caused by irrelevant factors, it is necessary to preprocess the data.

First, since BP value can be affected by numerous factors, single measurement cannot represent the condition of one's

BP. So we treat the average of the BP values of a month as a measurement value of this user.

Second, we choose the users with long measuring time spans. Since hypertension is a chronic disease, users always measure their BP as a routine. With just several measurements, the measurement data of the user may not reflect the real pattern of his BP. In this paper, users who use the monitor within a short time range will be neglected. We select users who measure their BP for more than five months.

Third, values of all features should be normalized to the same range. Normalization is a fundamental job in data processing. After we adjust values on different scales to a notionally common scale, the effect of the magnitude of numerical values in different features would be eliminated, only leaving the fluctuation trends of these features. Min-Max or Z-score normalization are commonly used. In this paper, we adopt Min-Max to rescale the data by using the following formula:

$$Value_{new} = \frac{Value_{origin} - Value_{min}}{Value_{max} - Value_{min}}. \tag{19}$$

Finally, because some users don't measure their BP every month, we use the twelve-dimentional one-hot vectors to represent the respective month in a year. The interval between this observation and the previous observation is also an effective feature. Furthermore, we transform the height and weight information of the user to a general index named Body Mass Index (BMI), which can reflect the level of obesity of the user. Some medical studies [5] show that obesity plays an important role in affecting people's BP.

5.2 Experiment Settings

The measuring time of our data ranges from January 1st in 2015 to July 31st in 2016. We set the BP of last month in the data set as the target. The data before the target month is the training set. The training process is that we predict the BP of last month in training set with the data before this month, which is to modify the parameters of the models. The whole data set is the testing set. The testing process is that we predict the target with the data before target month, which is to test the performance of the models.

We select the users whose measuring time spans are more than five months. The number of these users is 12671, but only 5894 of them have complete users' contextual data. In order to confirm the validity of contextual data and compare different strategies of filling up the missing data, we divide the whole data into two data sets:

- **dataset 1** All 5894 users in this data set have complete contextual data. We conduct our experiments on this data set to compare our model with other models.
- **dataset 2** Measuring time spans of 12671 users in this data set are more than five months. Part of these users have approximate contextual data. We compare different strategies of filling up the missing data based on this data set. (See Sec 5.5.2)

Because the RNN-CL and LSTM-CL models are proposed for sequential BP prediction, we can also use our models to predict the users' BP value in the second or third month after the last month in training set. To test its feasibility, we try to expand the interval between the target month and training months by dropping the last one or two months in training data. In this way, we can get three data sets. The description of these three data sets are as follows:

- **The next month** The target month is the next month after the last month in training set.
- **The second month** By dropping the last month in training set, the target month is the second month after the new training set.
- **the third month** By dropping the last one and two months in training set, the target month is the third month after the new training set.

5.3 Compared Methods

We compare our RNN-CL and LSTM-CL with six comparative methods of different categories.

- **SVR** is a classic regression model based on SVM. We use LibSVM[1] to implement the method.
- **GBRT** is a boosted method whose base learners are decision trees. We use the GradientBoostingRegressor API in the scikit-learn[2].
- **FM** is also a regression model based on matrix factorization. We implement it with LibFM[3].
- **MLP** is a basic neural network containing one hidden layer, and the output layer is linear regression model.
- **RNN** is an improvement of ANN, whose hidden layer can directly become the input of the next time step. The inputs of the models above can only be fixed length, but RNN and the following model, LSTM, can use variable-length inputs.
- **LSTM** replaces the activation function to an LSTM unit, and this unit can better record the historical memory than RNN. The implementation of this model is based on Tensorflow[4].

5.4 Evaluation Metrics

To measure the performance of these methods in predicting the BP value, we choose the most popular metrics, Root Mean Square Error (RMSE) and Mean Average Precision (MAE):

$$RMSE = \sqrt{\frac{\sum\limits_{i \in \Omega_{test}} (r_i - \hat{r}_i)^2}{n_{test}}}\ , \tag{20}$$

$$MAE = \frac{\sum\limits_{i \in \Omega_{test}} |r_i - \hat{r}_i|}{n_{test}}\ , \tag{21}$$

where Ω_{test} is the denotation of the testing set and n_{test} is the denotation of the total number of the users in the testing set. In these two metrics, the smaller value represents the better performance.

[1] http://https://www.csie.ntu.edu.tw/ cjlin/libsvm/

[2] http://scikit-learn.org

[3] http://www.libfm.org/

[4] https://www.tensorflow.org

Table 4: Experiments on diastolic BP and systolic BP and further prediction of the second and third month after the training months, measured by MAE and RMSE. Dataset 1 has complete contextual data and dataset 2 contains some users with approximate contextual data filled by KNN.

Data	Model	Diastolic Blood Pressure						Systolic Blood Pressure					
		The next month		The second month		The third month		The next month		The second month		The third month	
		MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
1	GBRT	3.4831	4.5676	4.1234	5.3826	4.4898	5.8153	5.6916	7.5010	6.7265	8.7802	7.2860	9.4772
	FM	3.4859	4.5877	4.1252	5.3977	4.5192	5.8523	5.6333	7.4431	6.6911	8.7386	7.3120	9.4939
	SVR	3.4034	4.5064	4.1098	5.3888	4.4891	5.8232	5.5728	7.3976	6.6929	8.7716	7.3236	9.5344
	MLP	3.4267	4.5193	4.1031	5.3662	4.4785	5.8082	5.6125	7.4097	6.6657	8.7089	7.2948	9.4782
	RNN	3.3691	4.4735	4.0185	5.3050	4.3326	5.6553	5.4246	7.2277	6.4762	8.5100	7.0837	9.2359
	RNN-CL	3.3374	4.4306	3.9910	5.2669	4.3177	5.6307	5.3981	7.1931	6.4470	8.4755	7.0597	9.2252
	LSTM	3.2917	4.3475	3.9123	5.1305	4.2761	5.5678	5.3672	7.0987	6.3233	8.2793	6.9715	9.0800
	LSTM-CL	**3.2558**	**4.2963**	**3.8553**	**5.0753**	**4.2420**	**5.5187**	**5.2906**	**7.0157**	**6.2991**	**8.2654**	**6.9243**	**9.0319**
2	GBRT	3.4608	4.5482	4.1166	5.3808	4.4765	5.7937	5.6359	7.4458	6.6729	8.7257	7.2628	9.4375
	FM	3.4141	4.5140	4.0950	5.3631	4.4710	5.7936	5.5895	7.4009	6.6668	8.7233	7.2817	9.4553
	SVR	3.4039	4.5115	4.0994	5.3768	4.4782	5.8075	5.5562	7.3876	6.6763	8.7548	7.3045	9.5034
	MLP	3.4172	4.5174	4.0943	5.3574	4.4766	5.7989	5.5993	7.4039	6.6552	8.7043	7.2830	9.4513
	RNN	3.3611	4.4613	4.0136	5.2824	4.3032	5.6123	5.4213	7.2140	6.4602	8.4906	7.0609	9.2284
	RNN-CL	3.3424	4.4339	3.9826	5.2559	4.2983	5.6097	5.4005	7.1905	6.4570	8.4889	7.0584	9.2266
	LSTM	3.2639	4.3168	3.9091	5.1271	4.2503	5.5431	5.3098	7.0308	6.3164	8.2544	6.9415	9.0579
	LSTM-CL	**3.2291**	**4.2681**	**3.8505**	**5.0718**	**4.2436**	**5.5358**	**5.2725**	**6.9668**	**6.2514**	**8.1900**	**6.9204**	**9.0435**

5.5 Analysis of Results

5.5.1 Comparison of Prediction Performance

Table 4 illustrates the performance on dataset 1 and dataset 2 with MAE and RMSE. We make predictions of diastolic BP and systolic BP respectively. In order to better display our model's effectiveness, we do experiments on datasets **the next month**, **the second month** and **the third month** respectively.

According to the results in Table 4, we get the following conclusions:

First, sequential prediction models outperform all the non-sequential models. For example, on dataset 1, LSTM improves the MAE of predicting diastolic BP in the next month by 6.53%, 6.60%, 4.34% and 4.99% compared to GBRT, FM, SVR and MLP. For RMSE, the improvements are 5.94%, 6.35%, 4.66% and 4.93%. The similar results have been shown in other experiment settings. From that we can draw the conclusion that sequential information is effective in the prediction of BP values.

Second, our proposed LSTM-CL achieves the best performance on all settings. RNN-CL also outperforms RNN on all settings. To demonstrate the effectiveness of using contextual layer, we compare RNN with RNN-CL, and LSTM with LSTM-CL. Taking the results of predicting diastolic BP in the next month on dataset 2 for example, LSTM-CL improves the MAE and RMSE by 1.06% and 1.13% compared to LSTM, and the improvements of RNN-CL are 0.556% and 0.614% compared to RNN. From all above we can see, contextual layer is able to improve the accuracy of the BP prediction.

Third, when we consider the situation of making further prediction, we can see that the performances of sequential prediction models are still far exceed the non-sequential models, and LSTM-CL still obtains the best performance. It shows that the accuracy of our proposed model can be stably better than other methods over different experiment settings.

Finally, systolic BP and diastolic BP predictions have the same increasing trend on each dataset. Besides, MAE and RMSE of systolic BP prediction are all always much larger than the corresponding values of diastolic BP prediction, which means that it is more difficult and challenging to predict accurate systolic BP.

Table 5: Performance comparison on different imputation strategies, measured by MAE and RMSE. The same setting in dataset 1 data set is the baseline.

Strategies	Diastolic BP		Systolic BP	
	MAE	RMSE	MAE	RMSE
Baseline	3.2558	4.2963	5.2906	7.0157
Mean	3.2509	4.2914	5.2883	7.0089
Regression	3.2632	4.3023	5.3044	7.0251
KNN	3.2291	4.2681	5.2725	6.9668

5.5.2 Comparison of Imputation Strategies

In this section, we investigate three imputation strategies of filling up the missing data and compare their performance. We use dataset 2 predicting the next month in the experiments. The baseline is the dataset 1 in the same experiment setting.

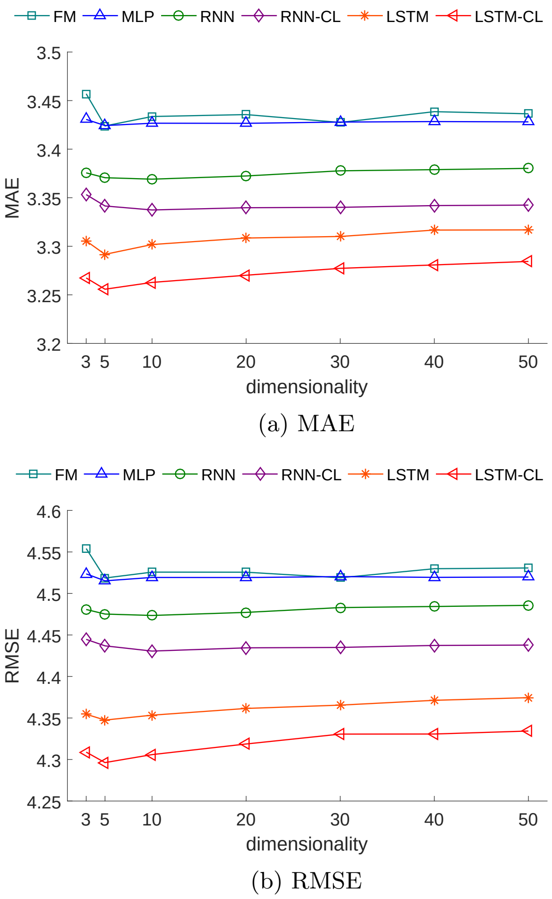
According to the results in Table 5, we get the following conclusions:

KNN imputation achieves the best result. It improves the MAE and RMSE of baseline by 0.820% and 0.656%. Mean imputation outperforms the baseline by 0.151% and 0.114%. However, the results of regression imputation are worse than baseline. We infer that regression model is overfitting during the training process. This phenomenon may be another source of bias. These results demonstrate that KNN is the best method to fill up the missing data.

Moveover, the trends in diastolic BP and systolic BP are similar, so these strategies can be applied in both diastolic BP data and systolic BP data.

5.5.3 Dimensionality Analysis

Dimensionality analysis is shown in Figure 4. We compare the performance with different dimensionalities (3, 5, 10, 20,



(a) MAE

(b) RMSE

Figure 4: Performance comparison of six models with different dimensionalities.

30, 40 and 50). We compare six comparative models with varying dimensionality. From Figure 4 we can see, LSTM-CL achieves the best results in any dimensionality. Different methods have similar trend on MAE and RMSE. MLP is much stable with the poorest performance.

Moreover, the best results of RNN and RNN-CL are on $d = 10$; LSTM and LSTM-CL are on $d = 5$. We infer that these models can hardly learn the parameters well when the dimensionality is big and the collecting data is not very large. Both LSTM and LSTM-CL are overfitting if the dimensionality becomes too large. Since the LSTM contains more parameters than RNN, the dimensionality of LSTM should be smaller than RNN.

6. CONCLUSION

In this paper, we propose the recurrent models with contextual layer to predict users' BP values. It extracts features from sequential measurement data and contextual data respectively by using the hidden layers of recurrent structure and add a contextual layer. Then this two kinds of data can be jointly processed to make the prediction. Given the condition that a part of users' contextual data is vacant, we use three strategies to fill up the missing data. In experiments, the results show that our models lead to great improvement over the existing models. Moveover, the experiments prove that KNN is the most suitable approach to fill up the missing contextual data.

In the future, we will further investigate the following directions. First, we can extend our model to deal with other medical problems. RNN-CL and LSTM-CL can also be universal models to deal with similar problems like predicting blood sugar and estimating the risk of stroke. Second, we need to find a method to jointly use the home BP monitoring data and office BP monitoring data. However, these two kinds of data have different features. So how to combine these features is the main challenge.

7. ACKNOWLEDGMENTS

The Shu Wu and Xiaohan Li contributed equally to this paper and are listed as joint first authors. This work is jointly supported by National Key Research and Development Program (2016YFB1001000), National Natural Science Foundation of China (61403390, U1435221), CCF-Tencent Open Fund and CCF-Venustech Hongyan Research Fund.

8. REFERENCES

[1] M. Auli, M. Galley, C. Quirk, and G. Zweig. Joint language and translation modeling with recurrent neural networks. *American Journal of Psychoanalysis*, 74(2):212–3, 2013.

[2] G. E. Batista and M. C. Monard. An analysis of four missing data treatment methods for supervised learning. *Applied Artificial Intelligence*, 17(5-6):519–533, 2003.

[3] F. P. Cappuccio, S. M. Kerry, L. Forbes, and A. Donald. Blood pressure control by home monitoring: Meta-analysis of randomised trials. *Bmj*, 329(7458):145, 2004.

[4] P. R. Chang and J. T. Hu. Optimal nonlinear adaptive prediction and modeling of mpeg video in atm networks using pipelined recurrent neural networks. *IEEE Journal on Selected Areas in Communications*, 15(6):1087–1100, 2006.

[5] A. V. Chobanian, G. L. Bakris, H. R. Black, W. C. Cushman, L. A. Green, J. L. Izzo Jr, D. W. Jones, B. J. Materson, S. Oparil, J. T. Wright Jr, et al. The seventh report of the joint national committee on prevention, detection, evaluation, and treatment of high blood pressure: the jnc 7 report. *Jama*, 289(19):2560–2571, 2003.

[6] J. Chung, C. Gulcehre, K. H. Cho, and Y. Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *Eprint Arxiv*, 2014.

[7] F. Dabek and J. J. Caban. *A Neural Network Based Model for Predicting Psychological Conditions*. 2015.

[8] G. B. Ehret, K. M. Rice, M. Bochud, A. D. Johnson, D. I. Chasman, A. V. Smith, M. D. Tobin, G. C. Verwoert, S. J. Hwang, and V. Pihur. Genetic variants in novel pathways influence blood pressure and cardiovascular disease risk. *Nature*, 478(7367):103–109, 2011.

[9] X. H. Fang, W. T. Longstreth Jr, S. C. Li, R. A. Kronmal, X. M. Cheng, W. Z. Wang, S. Wu, X. L. Du, and X. Y. Dai. Longitudinal study of blood pressure and stroke in over 37,000 people in china. *Cerebrovascular Diseases*, 11(3):225–9, 2001.

[10] J. H. Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

[11] M. Gerrard, F. X. Gibbons, A. E. Houlihan, M. L. Stock, and E. A. Pomery. Probabilistic detection of short events, with application to critical care monitoring. In *NIPS*, pages 49–56, 2008.

[12] A. Gers F., J. Schmidhuber, and F. Cummins. Learning to forget: Continual prediction with lstm. *Neural Computation*, 12(10):2451–71, 2000.

[13] H. F. Golino, L. S. Amaral, S. F. Duarte, C. M. Gomes, T. J. Soares, L. A. Dos Reis, and J. Santos. Predicting increased blood pressure using machine learning. *Journal of Obesity*, 2014(5):637635–637635, 2014.

[14] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, and J. Schmidhuber. A novel connectionist system for unconstrained handwriting recognition. *TPAMI*, 31(5):855–868, 2009.

[15] A. Graves, A. R. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. 1(2003):6645–6649, 2013.

[16] N. Hammerla, J. Fisher, P. Andras, L. Rochester, R. Walker, and T. Ploetz. Pd disease state assessment in naturalistic environments using deep learning. 2015.

[17] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[18] P. A. James, S. Oparil, B. L. Carter, W. C. Cushman, C. Dennison-Himmelfarb, J. Handler, D. T. Lackland, M. L. LeFevre, T. D. MacKenzie, O. Ogedegbe, et al. 2014 evidence-based guideline for the management of high blood pressure in adults: report from the panel members appointed to the eighth joint national committee (jnc 8). *Jama*, 311(5):507–520, 2014.

[19] J. Kolen and S. Kremer. *Gradient Flow in Recurrent Nets: The Difficulty of Learning LongTerm Dependencies*. Wiley-IEEE Press, 2009.

[20] S. Kombrink, T. Mikolov, M. Karafiát, and L. Burget. Recurrent neural network based language modeling in meeting recognition. In *INTERSPEECH*, volume 11, pages 2877–2880, 2011.

[21] Z. C. Lipton, D. C. Kale, C. Elkan, and R. Wetzell. Learning to diagnose with lstm recurrent neural networks. *Computer Science*, 2015.

[22] Q. Liu, S. Wu, D. Wang, Z. Li, and L. Wang. Context-aware sequential recommendation. 2016.

[23] Q. Liu, S. Wu, L. Wang, and T. Tan. Predicting the next location: A recurrent model with spatial and temporal contexts. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[24] A. G. Logan, A. Dunai, W. J. Mcisaac, M. J. Irvine, and A. Tisler. Attitudes of primary care physicians and their patients about home blood pressure monitoring in ontario. *Journal of Hypertension*, 26(3):446–52, 2008.

[25] A. F. Members, G. Mancia, R. Fagard, K. Narkiewicz, J. Redon, A. Zanchetti, M. BÃűhm, T. Christiaens, R. Cifkova, and G. D. Backer. 2013 esh/esc guidelines for the management of arterial hypertension. *European Heart Journal*, 34(28):2159–219, 2013.

[26] T. Mikolov, M. Karafiát, L. Burget, J. Cernockỳ, and S. Khudanpur. Recurrent neural network based language model. In *Interspeech*, volume 2, page 3, 2010.

[27] W. H. Organization. Q&As on hypertension. `http://www.who.int/features/qa/82/en/`, 2015.

[28] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[29] S. Rendle. Factorization machines. In *ICDM*, pages 995–1000. IEEE, 2010.

[30] C. Sideris, H. Kalantarian, E. Nemati, and M. Sarrafzadeh. Building continuous arterial blood pressure prediction models using recurrent networks. In *IEEE International Conference on Smart Computing*, pages 1–5, 2016.

[31] A. J. Smola and B. Schölkopf. A tutorial on support vector regression. *Statistics and computing*, 14(3):199–222, 2004.

[32] S. Somanchi, S. Adhikari, A. Lin, E. Eneva, and R. Ghani. Early prediction of cardiac arrest (code blue) using electronic medical records. In *ACM SIGKDD*, pages 2119–2126, 2015.

[33] G. S. Stergiou and E. G. Nasothimiou. Hypertension: Does home telemonitoring improve hypertension management? *Nature Reviews Nephrology*, 7(9):493–5, 2011.

[34] P. W. Wilson, R. B. D'Agostino, D. Levy, A. M. Belanger, H. Silbershatz, and W. B. Kannel. Prediction of coronary heart disease using risk factor categories. *Circulation*, 97(18):1837–1847, 1998.

[35] T. H. Wu, K. H. Pang, and W. Y. Kwong. Predicting systolic blood pressure using machine learning. pages 1–6, 2015.

[36] Y. Zhang, H. Dai, C. Xu, J. Feng, T. Wang, J. Bian, B. Wang, and T. Y. Liu. Sequential click prediction for sponsored search with recurrent neural networks. *Computer Science*, 2014.