Recurrent Neural Networks With Missing Information Imputation For Medical Examination Data Prediction

Han-Gyu Kim*, Gil-Jin Jang†, Ho-Jin Choi*, Minho Kim‡, Young-Won Kim‡ and Jaehun Choi‡
*School of Computing, Korea Advanced Institute of Science and Technology, Daejeon, South Korea 34141
Email: {kimhangyu, hojinc}@kaist.ac.kr
†School of Electronics Engineering, Kyungpook National University, Daegu, South Korea 41566
Email: gjang@knu.ac.kr
‡Electronics and Telecommunications Research Institute, Daejeon, South Korea 34129
Email: {kimmh, everywkim, jhchoi}@etri.re.kr

***Abstract*—In this work, we use recurrent neural network (RNN) to predict the medical examination data with missing parts. There often exist missing parts in medical examination data due to various human factors, for instance, because human subjects occasionally miss their annual examinations. Such missing parts make it hard to predict the future examination data by machines. Thus, imputation of the missing information is needed for accurate prediction of medical examination data. Among various types of RNNs, we choose simple recurrent network (SRN) and long short-term memory (LSTM) to predict the missing information as well as the future medical examination data, as they show good performance in many relevant applications. In our proposed method, the temporal trajectories of the medical examination measurements are modelled by RNNs with the missed measurements compensated, which is then used to predict the future measurements to be used as diagnosing the diseases of the subjects in advance. We have carried out experiments using a medical examination database of Korean people for 12 consecutive years with 13 medical fields. In this database, 11500 people took the medical check-up every year, and 7400 people missed their examination occasionally. We use complete data to train RNNs, and the data with missing parts are used to evaluate the imputation and future measurement prediction performance. In terms of root mean squared error (RMSE) and source to noise ratio (SNR) between the prediction and the actual measurements, the experimental results show that the proposed RNNs predicts medical examination data much better than the conventional linear regression in most of the examination items.**

***Index Terms*—medical examination data prediction; long short-term memory; recurrent neural network**

I. Introduction

Annual health examination is usually mandated by health insurance programs for their subscribers, and most countries have the insurance as a nationwide, basic services to their people. To make an early diagnosis of the potential diseases of the subjects, it is important to have complete health checkup data in order to make the diagnosis as accurate as possible. Early diagnosis enables finding the diseases in their early stages and providing suitable treatments to the subjects, resulting in complete cure even for the terminal diseases such as cancers, heart attacks, strokes, etc. Analyzing health examination results from the human subjects and finding their abnormal behaviors leading to potential diseases require well-trained and long-experienced medical doctors, but it takes huge amount of time and cost to provide adequate number of such professional doctors for the complete care of all the healthcare subscribers. Researches have been made to use machine learning techniques in analyzing the annual healthcare examination results and making meaningful prediction of some specific diseases, such as breast cancer [1], cardiac diseases [2], and the health meteorological information [3]. However, those methods are developed to be suited to the specific types of diseases and require the knowledge of the medical experts.

Recently, among the machine learning techniques, artificial neural networks (ANNs) with deep-layered architecture, called deep neural networks (DNNs) [4], [5], [6], have enabled analyzing large amount of data more accurately and finding more significant prediction of the diseases. Especially when the input data is a sequence of numerical observations, recurrent neural networks (RNNs) can model the trajectories of the sequential data by adding recurrent paths to ANNs [7]. Among the RNNs, long short-term memory (LSTM) is designed to selectively use or ignore the recurrent information from the past observations, and showed superior performances over the traditional RNNs in solving many problems such as speech recognition [8], [9] and natural language understanding [10], [11]. However, RNNs have strong requirements that the time intervals between adjacent input sequences should be equal and all the observations are available, i.e., without any missing data. In health examination applications, people often miss the regular checkups or there exist a lot of errors in the measurements due to various human factors. Therefore, the conventional RNNs can be applied to the subjects who completed annual health checkups without any missing, but those subjects are less likely to have diseases than subjects with irregular health checkups.

In this work, we propose novel RNN models that can predict medical examination measurements of human subjects given

the records of previous examinations, even when there is any missing observation. Among the many types of the RNNs, simple recurrent network (SRN) [7] and LSTM are chosen. SRN is a basic form of RNN constructed by adding a recurrent path to the standard neural network from the output to the input layer, and the backpropagation through time is used to train the network weights [12]. LSTM has 3 different gates in the input, output, and cell state layers, that determine to connect or disconnect the paths from the previous outputs, so that variable dependency lengths over time can be modeled by the numerous combinations of the gate openings. The SRN and LSTM are composed of many layers, and the linearly combined input values are passed through non-linear activation functions in all the layers, so that any forms of non-linear temporal variations of the medical examinations be accurately estimated. In order to handle the missing elements in the input sequence, the generated RNN outputs are use to estimate the missing ones, to be used as inputs to the next step. The estimation is repeated a number of times to increase the missing data estimation accuracy. To evaluate the performance of the SRN and LSTM, experiments were performed on the real medical examination database. The complete data of 11,500 persons without any missing parts were used to train the SNR and LSTM. The last year measurement values of the examination data of 7,400 persons who occasionally missed the examinations were predicted and compared with the true measurements. To show the effectiveness of non-linear models, prediction results of the linear regression [13] are compared to the proposed methods in terms of root mean squared errors (RMSEs) and signal-to noise ratios (SNRs). The experiment result shows that the proposed methods significantly outperform the conventional linear regression.

The paper is organized as follows: Section II explains the detailed algorithm of the standard linear regression, and Section III describes the details of SRN and LSTM for the general data prediction problems. Section IV shows the detailed analysis of the experiment results, and Section V makes the conclusion.

II. Linear Regression

The regression algorithms solves the problem of finding out the relationship between targets and known variables, so that the target may be predicted when a known variable is given. We call the known variables as independent variables, and the targets as dependent variables.

One of the most generally used regression algorithm is linear regression, as linear regression has a simple assumption thatn dependent variables and independent variables has linear relationship has follows:

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{b}, \tag{1}$$

where $\mathbf{x}$ is a vector composed of independent variables, $\mathbf{y}$ is a vector composed of dependent variables, and $\mathbf{A}$ and $\mathbf{b}$ are a slope matrix and a bias vector, respectively, that describe the linear relationship between $\mathbf{x}$ and $\mathbf{y}$ [13].

In the time-series regression problem, the current time instance t becomes the independent variable, and the observation at t becomes the dependent one, as the following equation:

$$\hat{\mathbf{y}}_n(t) = \mathbf{a}_n t + \mathbf{b}_n, \tag{2}$$

where $\hat{\mathbf{y}}_n(t)$ is the estimate of the output at time t, n is the person index, and $\mathbf{a}$, $\mathbf{b}$ are the affine transform vector and the bias vector as defined in Equation 2. If the true values of $\hat{\mathbf{y}}_n(t)$ is given, the optimal approximates of $\mathbf{a}_n$ and $\mathbf{b}_n$ are obtained by minimizing the sum of the squared errors between the pairs of the predicted values and the true values, denoted by $\mathbf{y}_n(t)$, for those time indices where $\mathbf{y}_n(t)$ exist:

$$\mathbf{a}_n^*, \mathbf{b}_n^* = \arg\min_{\mathbf{a}_n, \mathbf{b}_n} \sum_{t:\mathbf{y}_n(t)\,\mathtt{exist}} \left\{\hat{\mathbf{y}}_n(t) - \mathbf{y}_n(t)\right\}^2. \tag{3}$$

III. Recurrent Neural Networks

In Section II, the time-series the medical examination data is modelled using the linear regression, which assumes that the prediction target and the time index have linear relationship. In this work, we assume that there exist non-linear relationship between the current estimation and the past observation. We model such non-linear relationship as follows:

$$\mathbf{y}_n(t) = f\left(\mathbf{x}_n(t-1)\right), \tag{4}$$

where $\mathbf{x}_n(t-1)$ is the observation at time $t-1$, and $f(\cdot)$ is a non-linear function that is assumed to be properly express the characteristics of data. In order to effectively approximate the non-linear function $f(\cdot)$ in Equation 4, we use two types of RNNs, which are the simple recurrent network (SRN) and the long short-term memory (LSTM).

A. Simple Recurrent Network (SRN)

The SRN is one basic and simple type of RNNs [7]. In an SNR unit, the output layer has the recurrent connection to the input layer. The output of an SRN unit is computed by following equation:

$$\mathbf{z}(t) = \tanh(\mathbf{W}[\mathbf{a}(t); \mathbf{z}(t-1)] + \mathbf{b}), \tag{5}$$

where $\mathbf{z}(t)$ and $\mathbf{a}(t)$ are the output and input vectors of the SRN unit, respectively, $\mathbf{W}$ and $\mathbf{b}$ are the weight matrix and the bias vector of the SRN layer, $\tanh(\cdot)$ is the hyperbolic tangent activation function, and the operator $[\cdot;\cdot]$ represents vector concatenation. $\mathbf{W}$ and $\mathbf{b}$ are to be estimated in the training step. The activation function $\tanh(\cdot)$ guarantees the output of the SRN unit to be in the range of $(-1, 1)$, which helps the output of the SRN layer not to diverge with the long sequence of input.

In our work, we design the SRN network for medical examination result prediction as Figure 1. The network receives the medical examination result of previous year $\mathbf{x}_n(t-1)$ as input, and produces the medical examination result of current year $\mathbf{y}_n(t) = f_{SRN}\left(\mathbf{x}_n(t-1)\right)$ as output. We add batch normalization layer in the network for the robustness of network training [14]. Besides, the fully connected layer is also added as we want the whole network exports the output of

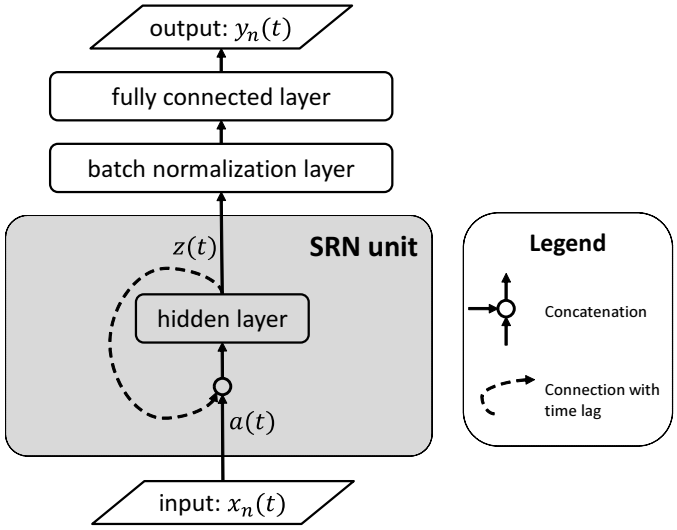

Fig. 1. The SRN network structure for medical examination data prediction.

desired dimension. The ReLU (rectified linear unit) activation function is used for the fully connected layer, as all values included in our medical examination dataset are positive.

For the network training process, we define the loss function of the network with the mean-squared error (MSE). The back-propagation through time is used to train the network [12]. In the inference stage, we predict the data of current year $\hat{\mathbf{y}}_n(t')$ using the data of previous years $\{\mathbf{x}_n(1), \mathbf{x}_n(2), \ldots, \mathbf{x}_n(t'-1)\}$, by sequentially putting those data of previous years into the already-trained SRN network.

B. Long Short-Term Memory

LSTM is a special variation of RNN which is popularly used in various areas. LSTM has ability to selectively store the useful information [15]. LSTM is generally known to work better than SRN [12]. In our work, we use LSTM instead of SRN to represent to predict the medical examination data.

An LSTM block is composed of cell state, input layer, input gate, forget gate and output gate. The historical information which LSTM considers to be useful is stored in the cell state $\mathbf{c}(t)$. The input gate decides which parts of the input are worth storing. Such decision is made by considering the current input $\mathbf{a}(t)$ and the previous output $\mathbf{z}(t-1)$ together:

$$\mathbf{g}_i(t) = \sigma(\mathbf{W}_i[\mathbf{a}(t); \mathbf{z}(t-1)] + \mathbf{b}_i), \tag{6}$$

where $\mathbf{g}_i(t)$ is the output of the input gate, $\mathbf{W}_i$ and $\mathbf{b}_i$ are weight matrix and bias vector of the input gate respectively, and $\sigma(\cdot)$ is the sigmoid activation function. The sigmoid activation function of the input gate guarantees that the output of the input gate $\mathbf{g}_i(t)$ is constrained in the range of $(0, 1)$: $\mathbf{g}_i(t)$ close to one allows the input information to be stored in the cell state; $\mathbf{g}_i(t)$ close to zero prevents the input information from being stored in the cell state. Such information selection is implemented by multiplying the input vector with the output of the input gate, which may be described with the equation as follows:

$$\tilde{\mathbf{c}}_i(t) = \tanh(\mathbf{W}[\mathbf{a}(t); \mathbf{z}(t-1)] + \mathbf{b}), \tag{7}$$

$$\mathbf{c}_i(t) = \tilde{\mathbf{c}}_i(t) * \mathbf{g}_i(t), \tag{8}$$

where $\mathbf{W}$ and $\mathbf{b}$ are weight matrix and bias vector of the input layer, $\tilde{\mathbf{c}}_i(t)$ is the information vector produced by the input layer using current input and previous output, $\mathbf{c}_i(t)$ is the selected information of the input layer that will be stored in the cell state, and the operator $*$ denotes the element-wise multiplication. The forget gate selects decides which part of the information stored in the cell state should be forgotten. Similar with the input gate, the cell state is masked with the output of the forget gate:

$$\mathbf{g}_f(t) = \sigma(\mathbf{W}_f[\mathbf{a}(t); \mathbf{z}(t-1)] + \mathbf{b}_f), \tag{9}$$

$$\mathbf{c}_f(t) = \mathbf{c}(t-1) * \mathbf{g}_f(t), \tag{10}$$

where $\mathbf{g}_f(t)$ is the output of the forget gate, $\mathbf{W}_f$ and $\mathbf{b}_f$ are weight matrix and bias vector of forget gate, and $\mathbf{c}_f(t)$ is the masked information of the cell state, which will remain in the next cell state. The forget gate also uses the sigmoid function as its activation function for the same reason with the input gate, which is to make $\mathbf{c}_f(t)$ to have a form of decision mask: $\mathbf{c}_f(t)$ close to one allows the corresponding information remain in the cell state, while $\mathbf{c}_f(t)$ close to zero forces the cell state to forget the corresponding information. The new cell state $\mathbf{c}(t)$ is generated by adding the information provided by input layer of Equation 8 and the information provided by the cell state of Equation 10 together as follows:

$$\mathbf{c}(t) = \mathbf{c}_i(t) + \mathbf{c}_f(t). \tag{11}$$

The output gate decides which part of the information should be exported as the final output of the whole LSTM block. The activation of the output gate $\mathbf{g}_o(t)$ is used to mask the output produced by the output layer, resulting in the final output $\mathbf{z}(t)$:

$$\mathbf{g}_o(t) = \sigma(\mathbf{W}_o[\mathbf{a}(t); \mathbf{z}(t-1)] + \mathbf{b}_o), \tag{12}$$

$$\mathbf{z}(t) = \tanh(\mathbf{c}(t)) * \mathbf{g}_o(t), \tag{13}$$

where $\mathbf{W}_o$ and $\mathbf{b}_o$ are weight matrix and bias vector of the output gate. The hyperbolic tangent function is used as the activation function of the output layer in order to force the output of the LSTM block to be in the range of $(-1, 1)$, which guarantees the network not to diverge for long sequence of input data.

In our work, we design LSTM network structure for medical examination data prediction as Figure 2. The proposed LSTM network predicts the data of current year $\mathbf{y}_n(t) = f_{LSTM}(\mathbf{x}_n(t-1))$, with receiving the data of previous year $\mathbf{x}_n(t-1)$ as input. Similar with the SRN network described in Section III-A, we use the batch normalization layer to stabilize the training process of the LSTM network. Besides, the fully-connected layer is inserted into our proposed LSTM network to obtain the output of desired dimension. The ReLU activation function is used as the activation function of the

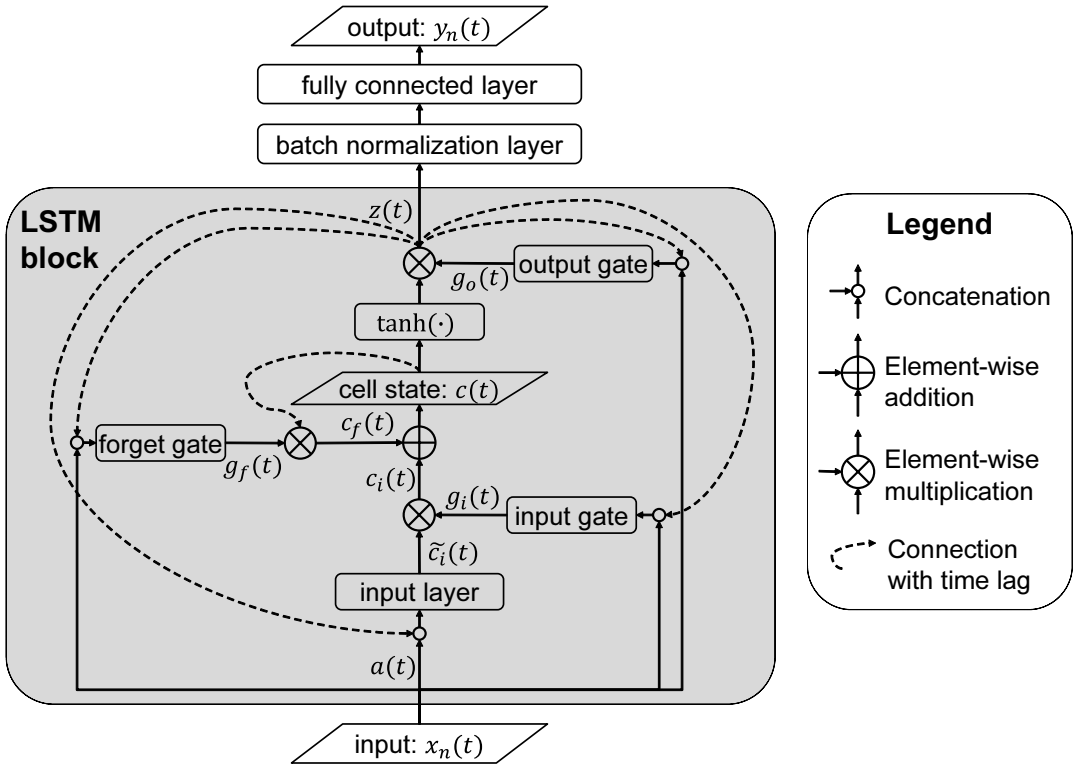

Fig. 2. The LSTM network structure for medical examination data prediction.

fully-connected layer as our medical examination data to be predicted is all positive.

In the training step, the MSE is used as the loss function. The back-propagation through time is used for network training. In the inference step, the medical examination data of current year $\hat{\mathbf{y}}_n(t')$ is predicted by providing the previous data $\{\mathbf{x}_n(1), \mathbf{x}_n(2), \ldots, \mathbf{x}_n(t'-1)\}$ as sequential inputs for already-trained LSTM network.

C. Missing Data Compensation Using RNNs

Compared to linear regression introduced in Section II, RNN has a disadvantage that RNN cannot handle the sequential data with missing parts, as our RNNs model non-linear relationship between consecutive data according to Equation 5.

In order to handle data with missing parts, we propose missing data imputation using RNNs. In our proposed method, the trained RNNs are used both for missing data imputation and target data prediction. In this method, the data prediction is done as Algorithm 1, where we assume that the first observation appears at time T_0 and the prediction target that we are interested in is at time T. When there are no missing data, the RNN is processed normally; when there appears missing data, the output of the RNN in the previous time step is used as the input of the current time step. With such missing data imputation method, the target data with missing parts may be predicted by our proposed RNNs.

Algorithm 1 RNN-based data prediction with missing information

Initialization: $t \leftarrow T_0$
while $t < T$ **do**
  if $x(t)$ is missing **then**
    $\hat{x}(t) \leftarrow \hat{y}(t)$ (missing data imputation)
    $\hat{y}(t+1) \leftarrow f_{RNN}(\hat{x}(t))$
  else
    $\hat{y}(t+1) \leftarrow f_{RNN}(x(t))$
  end if
  $t \leftarrow t+1$
end while
$\hat{y}_{\texttt{target}} = \hat{y}(T)$

IV. Experiment Result

A. Experiment Design

In our experiment, we conducted three medical examination data prediction methods, which are the linear regression (LR), the simple recurrent network (SRN) and the long short-term memory (LSTM). We used Korean medical examination database called NHIS-NSC, which contains complete data of 11500 persons in 2002-2013, and contains data with missing information of 7400 persons. This database contains 13 items, where nine of them are the items we tend to predict, and the remaining four items are sex, age, weight and height, which we are not interested in predicting. Nine prediction target items

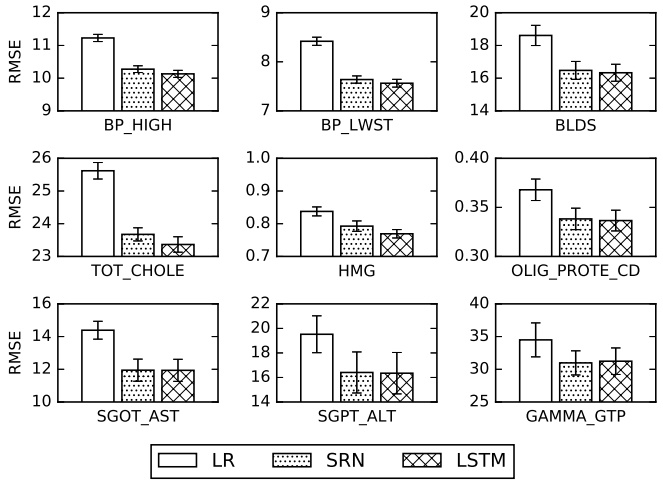

Fig. 3. RMSE of the complete dataset.

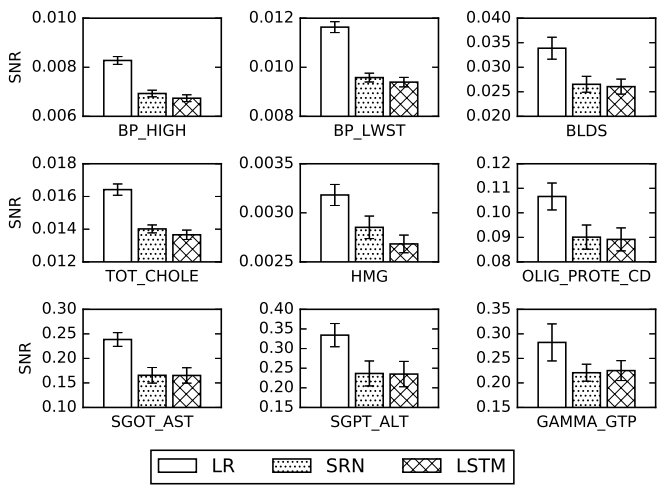

Fig. 4. SNR of the complete dataset.

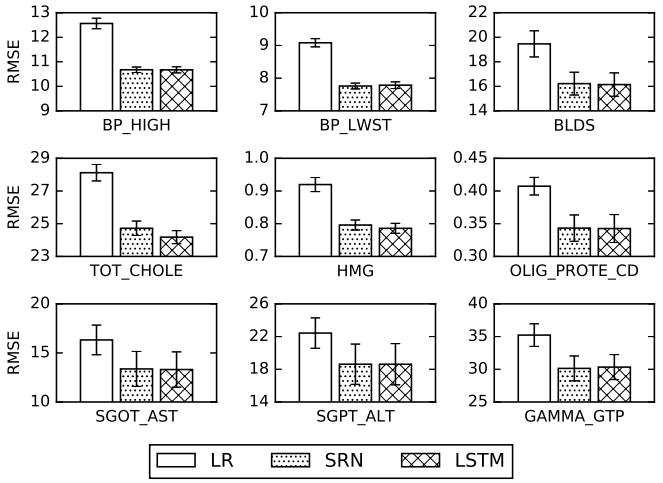

Fig. 5. RMSE of the dataset with missing parts.

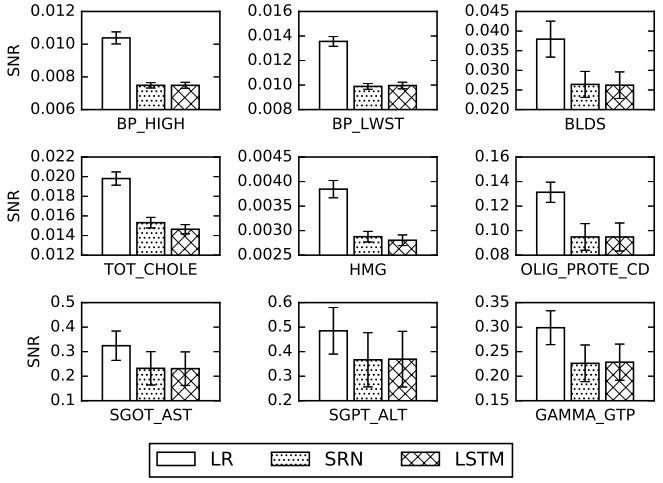

Fig. 6. SNR of the dataset with missing partss.

are listed in Table I. The remaining 4 items are only used as the input of the prediction system, which means they are only included in $\mathbf{x}_n(t-1)$, but excluded from $\mathbf{y}_n(t)$.

In order to evaluate the performance of the prediction systems fairly, the ten0fcross validation experiment was conducted [16]. In our experiment, we conducted ten-fold cross validation: complete medical examination data of 11500 persons was divided into ten parts where each part contains data of 1150 persons; eight of them were used as training set, one of them was used as validation set, and the remaining one of them and the data set with missing parts were used as test set. In the test step, the medical examination result of 2013 is predicted, using the data of 2002-2012.

B. Experiment Result

In our experiment, we used two criteria for performance evaluation. The first evaluation criteria is the root mean-squared error (RMSE), which is computed as follows,

$$RMSE = \sqrt{\frac{1}{N}\sum_{n=1}^{N} ||\hat{\mathbf{y}}_n(t) - \mathbf{y}_n(t)||^2}, \tag{14}$$

where N is the number of persons, $\hat{\mathbf{y}}_n(t)$ is prediction result, and $\mathbf{y}_n(t)$ is the true data. The second evaluation criteria is the source-to-noise ratio (SNR), which is computed using the following equation:

$$SNR = \frac{\sum_{n=1}^{N} ||\hat{\mathbf{y}}_n(t) - \mathbf{y}_n(t)||^2}{\sum_{n=1}^{N} ||\mathbf{y}_n(t)||^2}. \tag{15}$$

RMSE and SNR are similar evaluation criteria. A better prediction algorithm should show smaller RMSE and smaller SNR. Compared to RMSE, SNR is less affected by the scale of the data.

The means and the standard errors of RMSEs and SNRs of all items of complete dataset are shown in Figures 3 and 4. The

TABLE I
MEDICAL EXAMINATION ITEMS TO BE PREDICTED.

Abbreviation	Item name
BP_HIGH	highest blood pressure
BP_LWST	lowest blood pressure
BLDS	blood sugar
TOT_CHOLE	total cholesterol
HMG	hemoglobin
OLIG_PROTE_CD	urinary protein
SGOT_AST	serum glutamic oxaloacetic transaminase (aspartate transaminase)
SGPT_ALT	serum glutamate-pyruvate transaminase (alanine transaminase)
GAMMA_GTP	gamma-glutamyl transpeptidase

prediction result on complete dataset shows that our proposed methods based on RNNs outperform the baseline method which is linear regression, both for RMSE and SNR, and for all items predicted in our experiment. The error bars shown in Figures 3 and 4 imply that the performance improvement of our proposed methods is significant, as almost all the error bars of proposed methods do not overlap with the corresponding of error bars of linear regression. All items except GAMMA _GTP are predicted significantly better by using RNNs than by using linear regression. The performance for GAMMA_G TP is still better, but the difference is not significant.

The means and standard errors of the RMSEs and the SNRs for dataset with missing information are shown in Figures 5 and 6, respectively. Similar with the experiment result for the complete data, the proposed methods based on RNNs work much better than linear regression, both for RMSE and SNR, and for all predicted items. Furthermore, the performance improvement is also significant for data with missing information, as almost all the error bars of RNNs and linear regression do not overlap. The significant performance difference is shown in BP_HIGH,BP_LWST,BLDS, TOT_C HOLE, HMG, OLIG_PROTE_CD and GAMMA_GTP. For S GOT_AST and SGPT_ALT, the proposed methods still show better performance than linear regression but the difference is not significant.

Besides, all four figures show that SRN and LSTM show similar performances. There are no significant performance difference between two proposed methods.

V. CONCLUSION

In this work, we propose medical examination data prediction methods using SRN and LSTM for medical examination data prediction. For comparison, we also implement data prediction method based on linear regression as the baseline method. In order to successfully apply SRN and LSTM to real data which occasionally contains missing information, we propose missing data imputation using RNNs. The prediction experiment on medical examination data of Korean persons shows that SRN and LSTM works significantly better than linear regression for almost all medical examination items, both for RMSE and SNR.

The prediction made by our methods may help find potential patients before they have their medical examination. Those potential patients will be strongly recommended not to miss their medical examination, or will be suggested to take specific type of medical examination which may fit their health state.

In future work, we will try design deeper RNNs to further improve the performance of medical data prediction.

ACKNOWLEDGMENTS

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. NRF-2016R1E1A2020559 and No. 2010-0028631), and by the ICT R&D programof MSIP/IITP [B0101-15-247, Development of open ICT healing platform using personal health data]. This study used NHIS-NSC data (NHIS-2015-2-017) made by National Health Insurance Service (NHIS) in Korea. The authors declare no conflict of interest with NHIS.

REFERENCES

[1] L. Ahmad, A. Eshlaghy, A. Poorebrahimi, M. Ebrahimi, and A. Razavi, "Using three machine learning techniques for predicting breast cancer recurrence," *Journal of Health and Medical Informatics*, vol. 4, no. 124, 4 2013.

[2] P. Hachesu, M. Ahmadi, S. Alizadeh, and F. Sadoughi, "Use of data mining techniques to determine and predict length of stay of cardiac patients," *Healthcare Informatics Research*, vol. 19, no. 2, pp. 121–129, 6 2013.

[3] J. Oh and B. Kim, "Prediction model for demands of the health meteorological information using a decision tree method," *Asian Nursing Research*, vol. 4, no. 3, pp. 151–162, 9 2010.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[5] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 30–42, 2012.

[6] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[7] J. L. Elman, "Distributed representations, simple recurrent networks, and grammatical structure," *Machine learning*, vol. 7, no. 2-3, pp. 195–225, 1991.

[8] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *2013 IEEE international conference on acoustics, speech and signal processing*. IEEE, 2013, pp. 6645–6649.

[9] K. Yao, B. Peng, Y. Zhang, D. Yu, G. Zweig, and Y. Shi, "Spoken language understanding using long short-term memory neural networks," in *Spoken Language Technology Workshop (SLT), 2014 IEEE*. IEEE, 2014, pp. 189–194.

[10] T.-H. Wen, M. Gasic, N. Mrksic, P.-H. Su, D. Vandyke, and S. Young, "Semantically conditioned LSTM-based natural language generation for spoken dialogue systems," *arXiv preprint arXiv:1508.01745*, 2015.

[11] W. Zaremba, I. Sutskever, and O. Vinyals, "Recurrent neural network regularization," *arXiv preprint arXiv:1409.2329*, 2014.
[12] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space odyssey," *arXiv preprint arXiv:1503.04069*, 2015.
[13] X. Yan, *Linear regression analysis: theory and computing*. World Scientific, 2009.
[14] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.
[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
[16] C. Bishop, *Pattern Recognition and Machine Learning*. Springer, January 2006.