

Original Paper

Selective Prediction With Long Short-term Memory Using Unit-Wise Batch Standardization for Time Series Health Data Sets: Algorithm Development and Validation

Borum Nam¹, BS; Joo Young Kim², BS; In Young Kim^{2*}, MD; Baek Hwan Cho^{3*}, PhD

¹Department of Electronic Engineering, Hanyang University, Seoul, Republic of Korea

²Department of Biomedical Engineering, Hanyang University, Seoul, Republic of Korea

³Medical AI Research Center, Samsung Medical Center, Seoul, Republic of Korea

*these authors contributed equally

Corresponding Author:

Baek Hwan Cho, PhD

Medical AI Research Center

Samsung Medical Center

81, Irwon-ro, Gangnam-gu

Seoul, 06351

Republic of Korea

Phone: 82 234100885

Email: baekhwan.cho@samsung.com

Abstract

Background: In any health care system, both the classification of data and the confidence level of such classifications are important. Therefore, a selective prediction model is required to classify time series health data according to confidence levels of prediction.

Objective: This study aims to develop a method using long short-term memory (LSTM) models with a reject option for time series health data classification.

Methods: An existing selective prediction method was adopted to implement an option for rejecting a classification output in LSTM models. However, a conventional selection function approach to LSTM does not achieve acceptable performance during learning stages. To tackle this problem, we proposed a unit-wise batch standardization that attempts to normalize each hidden unit in LSTM to apply the structural characteristics of LSTM models that concern the selection function.

Results: The ability of our method to approximate the target confidence level was compared by coverage violations for 2 time series of health data sets consisting of human activity and arrhythmia. For both data sets, our approach yielded lower average coverage violations (0.98% and 1.79% for each data set) than those of the conventional approach. In addition, the classification performance when using the reject option was compared with that of other normalization methods. Our method demonstrated superior performance for selective risk (12.63% and 17.82% for each data set), false-positive rates (2.09% and 5.8% for each data set), and false-negative rates (10.58% and 17.24% for each data set).

Conclusions: Our normalization approach can help make selective predictions for time series health data. We expect this technique to enhance the confidence of users in classification systems and improve collaborative efforts between humans and artificial intelligence in the medical field through the use of classification that considers confidence.

(*JMIR Med Inform* 2022;10(3):e30587) doi: [10.2196/30587](https://doi.org/10.2196/30587)

KEYWORDS

artificial intelligence; recurrent neural networks; biomedical informatics; computer-aided analysis; mobile phone

Introduction

Background

High-performance networks have been used to enhance the quality and convenience of human life since the development of deep learning techniques. Deep learning networks are used in education, aviation, process management, entertainment, agriculture, and robotics. Artificial intelligence (AI) has made significant contributions to a variety of medical applications [1-3]. However, in a clinical setting, the output from AI as an accurate prediction is often insufficient and requires its interpretation for further decisions [4]. As medical AI systems can support efficient and accurate decisions, it is important not only to increase the accuracy of classification in deep learning networks but also to reduce errors, particularly those that can be fatal [5]. In addition, health care data tend to be complex, and neural networks have proven problematic in accurately recognizing patterns in this complexity [6]. The uncertainty of prediction measures the reliability of a prediction and must be considered in fields that require prudent decisions, such as medicine or autonomous driving [7]. Accordingly, in fields where minor errors can cause significant problems, applying a prediction model that can reject predictions when the confidence level is not high enough is helpful. To develop such a deep neural network, a selective prediction [8] method can be applied to use the confidence level in both training and test sessions.

Various biosignal sensors have been developed for human health care applications, and many algorithms have been developed to analyze the data produced by these sensors. Deep learning technologies have performed well when applied to data obtained from health care or medical sensors [9]. Classification models based on a deep neural network or convolutional neural network (CNN) have been used to classify health and medical data. In addition, biosignals and time series data from humans are used in diverse health care systems [10]. In various studies, recurrent neural network (RNN) models have been used to classify health and medical data, especially time series data. Among such models, RNNs have contributed significantly to the classification of time series data. Many studies have used RNN models to classify electronic health records obtained from clinical measurements [11], predict diseases using patient diagnostic histories [12-14], conduct health status analyses using biosignals [15-18], and classify health information from mobile and wearable sensors [19-22]. Previous studies have applied prediction confidence to classify image data, and prediction confidence can be considered for classifying time series health data using RNN models. However, little research has focused on how to use prediction confidence for time series health data.

Considering the specificity of time series health data, a model that can produce results according to the predicted confidence level and uses prediction confidence has the advantage of reducing fatal errors.

The selective prediction model can learn from certain samples that are sufficiently confident in their predictions. This means that such a model can ignore predictions when they are uncertain in training. In addition, the selective prediction model provides a confidence level for each test sample in the inference stage,

which can be used as a reference score in a medical situation. In early studies on selective prediction, neural network models with a reject option were used to obtain a specific confidence score from a trained model and as a model threshold to validate performance [23-25]. However, these methods calculate the prediction probability to select samples for training based on a threshold called the prediction confidence score.

Recently, research using the selective prediction model mainly consists of 2 parts. The first is to extract an appropriate prediction confidence score and the second is to make good use of the extracted prediction confidence score for the deep learning model. For extracting the prediction confidence score, methods have been designed in many studies. For example, the softmax response and Monte Carlo (MC) dropout methods use a confidence score from neural networks [26]. The softmax response method extracts a confidence score using maximum softmax values from neural networks, as described in the above methods, whereas an MC dropout estimates a confidence score using statistical approaches. However, MC dropout requires a high computational cost to optimize the problem quickly. Although Bayesian methods [27-29] can produce prediction confidence scores of RNNs [30], they are applicable only for natural language processing, which uses *many-to-many* RNNs with multiple sequence inputs and outputs. However, the predictive models in health care are usually *many-to-one* types that predict class using a health information time series as input, and it is helpful for medical staff to train a *many-to-one* predictive model for time series data that has a selective prediction ability. For a model using the prediction confidence score, a selective prediction model that learned both prediction and selection was developed [31]. On the basis of this method, SelectiveNet [32] has demonstrated potential possibilities for various applications, with the advantage of learning the selection and prediction simultaneously. However, the structure of the selective prediction model using long short-term memory (LSTM) has not been validated in previous studies. Thus, a well-designed selective prediction model for time series data is required.

Objective

In this study, a selective prediction model using LSTM [33] was implemented to classify time series health data. In particular, we considered a method that incorporates a reject option to control and measure prediction confidence for *many-to-one* classification tasks. As the selection function uses the output of the prediction model as an input, a suitable selection function structure must be devised. Therefore, methods to normalize the selection function were compared to achieve a structure suitable for classifying time series data with LSTM. To validate the LSTM selective prediction performance, we used coverage violations and selective risks for each data set. As high false-positive and false-negative rates can be critical factors in diagnoses, we also present the false-positive and false-negative rates of the LSTM selective prediction model. In summary, the goal of this study is to develop a selective prediction model for health data time series. The contributions of this study are (1) applying the latest selective prediction method with superior performance to classify time series health data using LSTM and (2) presenting the structure of the selection

function in the selective prediction model (especially the normalization method) for time series selective prediction.

Methods

Selective Prediction

We examined the possibility of RNN models with a reject option using SelectiveNet [32], which has superior performance compared with existing selective prediction models. The overall structure of the model was based on the SelectiveNet [32] model with an LSTM; it is divided into selective and auxiliary predictions, as shown in Figure 1. The selective prediction is divided again into two steps: prediction and selection. Prediction involves the results of the LSTM model and the selection part extracts the predicted confidence level of the LSTM model. In this study, we propose unit-wise batch standardization (UBS) as part of the selection function. Selective prediction is performed using both the prediction and selection function results. An auxiliary prediction step using the LSTM prediction result to derive the final result with the selective prediction result was added to enhance prediction performance. As selective prediction is a prediction model using a deep learning model structure, it is optimized by a loss function. The entire model is trained by optimizing the selective prediction and auxiliary prediction steps simultaneously. Further details are provided in the *Optimization* section. LSTM was used for the RNN model for time series data classification.

A selective model was used to implement classification models with the reject option [34]. The selective model (f, g) consists of pairing a prediction function f and a selection function $g: X \rightarrow Y$ ($Y|0 \leq Y \leq 1$) (X is a set of inputs and Y is a set of outputs). When the data set is given as $S_m = \{(x_i, y_i)\}_{i=1}^m$ for supervised learning of

the classification model, the empirical risk of prediction function f becomes $\hat{r}(f, g|S_m) \triangleq \frac{1}{m} \sum_{i=1}^m \ell(f(x_i), y_i) g(x_i)$. When τ is a threshold, g acts as a qualifier of f and can be expressed as follows:

$$(f, g)(x) \triangleq \begin{cases} f(x), & \text{if } g(x) \geq \tau; \\ \text{don't know,} & \text{otherwise.} \end{cases} \quad (1)$$

Selective models can be controlled by coverage and risk values. When E_p is the expected probability, and ℓ is the loss function, we can define the coverage and risk as follows:

$$\begin{aligned} \phi(g) &\triangleq E_p[g(x)] \\ R(f, g) &\triangleq \frac{E_p[\ell(f(x), y)g(x)]}{\phi(g)} \end{aligned} \quad (2)$$

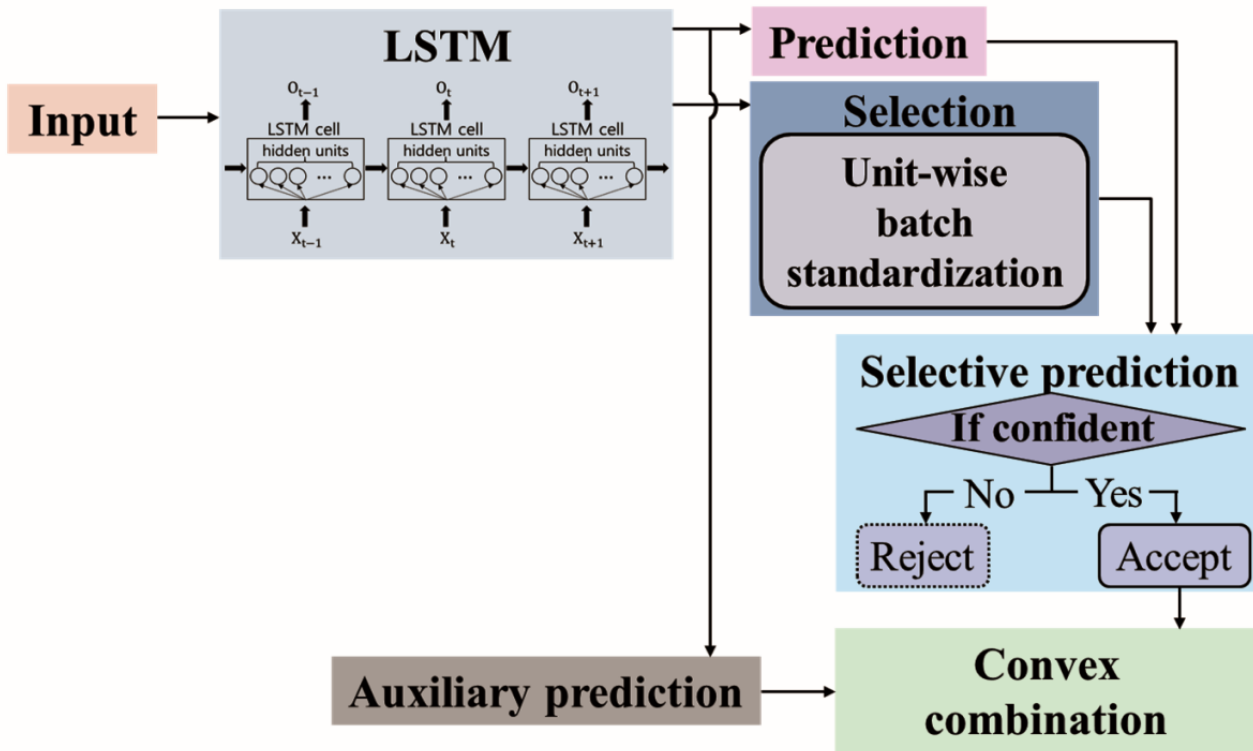
where $g(x)$ is the prediction confidence score, $\phi(g)$ is a coverage value that is the expected value of the prediction confidence scores for training samples, which is correlated with the number of selected samples during training. $R(f, g)$ is a selective risk that represents the error rate for predicting the selected samples using selective prediction. The corresponding selective risk for a data set $S_m = \{(x_i, y_i)\}_{i=1}^m$ is called the empirical selective risk and is defined as follows:

$$\hat{r}(f, g|S_m) \triangleq \frac{\frac{1}{m} \sum_{i=1}^m \ell(f(x_i), y_i) g(x_i)}{\hat{\phi}(g|S_m)} \quad (3)$$

The empirical coverage corresponding to the data set S_m is as follows:

$$\hat{\phi}(g|S_m) \triangleq \frac{1}{m} \sum_{i=1}^m g(x_i) \quad (4)$$

Figure 1. Long short-term memory model structure with a reject option. LSTM: long short-term memory.



Optimization

An optimization method was used to constrain coverage and reduce the selective risk [31]. The selective prediction model was optimized by the loss functions in equations 6, 7, and 8. This loss function simultaneously regulates the prediction and selection steps. Hence, the selective prediction was regulated to lower the error rate, which is the selective risk for the selected samples according to the prediction confidence. In addition, the selection step was optimized to select training samples based on the predefined target coverage so that the selection step would reject predictions below the confidence level. The target coverage is a controlling hyperparameter for the model to learn the amount of data to be selected during training. On the basis of this, we trained the model so that the coverage value was as close to the target coverage as possible. The target coverage c is in the range $0 < c \leq 1$. When the parameter set of the selective model (f, g) is Θ the optimization of the selective model is as follows:

$$\theta^* = \underset{\theta \in \Theta}{\operatorname{argmin}} (R(f_\theta, g_\theta)) \quad (5)$$

$$s.t. \phi(g_\theta) \geq c$$

The f_θ and g_θ in the selective prediction were optimized by equation 6. It is necessary to constrain coverage and reduce risk (error) for selective prediction. We used the interior point method for optimization [35]. The following unconstrained objective is used to optimize the selective prediction model for a data set S_m :

$$\mathcal{L}_{(f,g)} \triangleq \hat{r}_\ell(f, g|S_m) + \lambda \Psi(c - \hat{\phi}(g|S_m)) \quad (6)$$

$$\Psi(a) \triangleq \max(0, a)^2$$

where c is the target coverage, and λ is a hyperparameter that controls the coverage constraints. Using equation 6, the selection function g is optimized to produce an appropriate prediction confidence score, and the selective prediction is optimized to reduce the selective risk \hat{r}_ℓ . The empirical coverage value $\hat{\phi}(g|S_m)$ is probabilistically calculated using the selection function. The Ψ allows the coverage value $\hat{\phi}(g|S_m)$ to approximate the target coverage during the training session. The auxiliary classification loss is optimized using the loss function \mathcal{L}_h . Overall, optimization can be defined using a convex combination expressed by the following equations:

$$\mathcal{L}_h = \hat{r}(h|S_m) = \frac{1}{m} \sum_{i=1}^m \ell(h(x_i), y_i) \quad (7)$$

$$\mathcal{L} = \alpha \mathcal{L}_{(f,g)} + (1 - \alpha) \mathcal{L}_h \quad (8)$$

where α is another user-controlled parameter for the weights between the selective and auxiliary predictions.

UBS Procedure

In this study, a new selection function structure for LSTM models was designed. The basic frame of the selection function structure was based on a CNN-based model from a previous study [32] that used batch normalization [36] for the selection function. The detailed structure and parameters were determined through a grid search. The output shape of the *many-to-one* structure LSTM is (n_batch, n_hidden_unit) , with conventional batch normalization, applying the same mean and variance to all units. However, this method of normalization ignores the features of each hidden unit in the LSTM output. To address this problem, we applied a new UBS that normalizes the batch

derived from an original batch normalization [36] while preserving the hidden-unit features captured for each training sample. As shown in Table 1, UBS uses a fully connected layer that maintains the LSTM output's shape while generating the output and standardizing the batch, as shown in Figure 2. When

batch normalization is applied to CNNs, normalization factors (mean and variance) are obtained from each input channel [37]. However, to preserve hidden units' individual features, we calculated normalization factors obtained from each LSTM's hidden unit.

Table 1. Detailed structure of the selective prediction step.

Layer	Input shape	Output shape
LSTM ^a	(n_batch, n_time steps, n_features)	(n_batch, n_hidden unit)
FC1 ^{b,c}	(n_batch, n_hidden unit)	(n_batch, n_hidden unit)
FC2 ^{b,d}	(n_batch, n_hidden unit)	(n_batch, n_hidden unit)
ReLU ^{b,e}	(n_batch, n_hidden unit)	(n_batch, n_hidden unit)
UBS ^{b,f}	(n_batch, n_hidden unit)	(n_batch, n_hidden unit)
FC3 ^g	(n_batch, n_hidden unit)	(n_batch, 1)
Sigmoid	(n_batch, 1)	(n_batch, 1)

^aLSTM: long short-term memory.

^bThe layer retains the input.

^cFC1: fully connected layer 1.

^dFC2: fully connected layer 2.

^eReLU: rectified linear unit.

^fUBS: unit-wise batch standardization.

^gFC3: fully connected layer 3.

Figure 2. Algorithm of unit-wise batch standardization. LSTM: long short-term memory; ReLU: rectified linear unit.**Algorithm : Unit-wise batch standardization (UBS)****Input :** Values of x over after ReLU layer in the selection function O_s :

$$O_s = \begin{bmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mn} \end{bmatrix}$$

 m : mini batch size, n : number of hidden units in LSTMScale and shift factors : $\gamma = \{r_{1\dots n}\}$, $\beta = \{b_{1\dots n}\}$

Parameters to be learned (tiling scale and shift factors) :

$$\gamma = \begin{bmatrix} r_1 & \cdots & r_n \\ \vdots & \ddots & \vdots \\ r_1 & \cdots & r_n \end{bmatrix}, \beta = \begin{bmatrix} b_1 & \cdots & b_n \\ \vdots & \ddots & \vdots \\ b_1 & \cdots & b_n \end{bmatrix}$$

Output : Input standardization along axis 0 :Unit-wise batch standardization $y = \text{UBS}(O_s)$ **for** $j = 1$ to n **do**

$$\mu_j \leftarrow \frac{1}{m} \sum_{i=1}^m x_{ij} \quad // \text{ mini-batch mean for the } j\text{th hidden unit}$$

$$\sigma_j^2 \leftarrow \frac{1}{m} \sum_{i=1}^m (x_{ij} - \mu_j)^2 \quad // \text{ mini-batch variance for the } j\text{th hidden unit}$$

for $i = 1$ to m **do**

$$\hat{x}_{ij} \leftarrow \frac{x_{ij} - \mu_j}{\sqrt{\sigma_j^2 + \epsilon}} \quad // \text{ normalize for the } j\text{th hidden unit}$$

end for**end for**

$$\tilde{y} \leftarrow \hat{x} \quad // \text{ Unit-wise batch standardization}$$

$$y \leftarrow \gamma \tilde{y} + \beta \quad // \text{ scale and shift}$$

Performance Evaluation

In a health care system, a misdiagnosis involving a type 2 error may imply serious repercussions, and incorrect judgment involving a type 1 error may increase user fatigue. Therefore, we verified the performance of the algorithm by checking false-positive and false-negative rates. The false-positive rate (also known as type 1 error, fall-out, or false-alarm ratio) was calculated as the ratio between the number of negative events incorrectly identified as positive and the total number of actual negative events. The false-negative rate (type 2 error) was calculated as the number of samples misclassified as negative out of the total number of positive events.

Experiment**Overview****Data Sets**

This study was reviewed and approved by the institutional review board (#HYUIRB-202111-003) of the Hanyang University, and the requirement for informed consent was waived. A widely used public database was employed to verify

the applicability of the selective prediction model to time series health care data. Considering that the purpose of selective prediction is to reject uncertain predictions, we selected two data sets containing classes that can be misclassified [38-42]: the *human activity recognition using smartphones* and the *Massachusetts Institute of Technology-Beth Israel Hospital (MIT-BIH)* data sets. Detailed descriptions of the data sets have been provided below.

Human Activity Recognition Using Smartphones Data Set

This data set consists of human gait signals monitored by an accelerometer and gyroscope with 6 different activity classes [43]. The signal was measured by attaching Samsung Galaxy S2 smartphones with embedded inertial sensors to the waists of 30 subjects aged 19 to 48 years. Each subject performed six activities (standing, sitting, laying, walking, walking upstairs, and walking downstairs) at least two times for 12 to 15 seconds. The 3-axial linear acceleration and angular velocity were measured at 50 Hz using an embedded accelerometer and gyroscope. The experiments were video-recorded to label the data manually. The signals were preprocessed using a median filter and a third-order low-pass Butterworth filter with a 20-Hz

cutoff frequency and then sampled in sliding windows of 2.56 seconds with 50% overlap (128 readings/window). A total of 10,299 data points were recorded. The training data were randomly selected from 70% of the data set, and the remaining data set was used for the test. The x, y, and z components of the body accelerometer, body gyroscope, and total (gravitational and body) accelerometers were treated as 9 input features. Each sample contained 128 sequences.

MIT-BIH Arrhythmia Data Set

This data set contains 48 half-hour excerpts of two-channel ambulatory electrocardiogram (ECG) recordings from 47 subjects [44]. The recordings were digitized at 360 samples per second per channel with 11-bit resolution over a 10-mV range and annotated independently by 2 or more cardiologists. The data set is publicly available in the PhysioNet [45] database. All protected health information was removed and deidentified using record numbers. A method described in a previous study was used for preprocessing data [46]. First, ECG signals were divided into 10-second intervals. Subsequently, the signal was normalized between 0 and 1. Where the median of the R-R time interval in the ECG signal was T, the time from the R peak to 1.2 T was used as 1 segment. Because the length of the segment changes every 10 seconds, the length of the entire data set is zero-padded based on the longest time. The data set consisted of 109,446 data points with a sampling frequency of 125 Hz. Each data set contained 187 sequences grouped into five classes: N (normal beat), S (supraventricular premature beat), V (premature ventricular contraction), F (fusion of ventricular and normal beats), and Q (unclassifiable beat). Unclassifiable data were not included in this study. As the data for each class were highly imbalanced, 800 data samples were randomly extracted from each class [46]. The data set was sampled for every run, and result was expressed as an average of the results. The data set was then randomly divided into sets: 80% for training and 20% for testing.

Model Architecture and Parameters Setting

Overview

In this study, a selective prediction model was developed using LSTM. Deep learning models such as LSTM are considered effective for extracting meaningful features from raw data. No feature extractor was used in this study because a deep learning model is suitable for use with raw data. The prediction model architecture was determined and optimized based on previous studies, and hyperparameters were optimized using an extensive grid search [47,48]. The details for each data set are described below.

Human Activity Recognition Using Smartphones Data Set

The LSTM model for the human activity recognition using smartphones data set had a single layer with 2 cells and 32

hidden units. For parameter setting, the learning rate was 0.0005, and the L2 regularization was set at a lambda of 0.00005. The mini batch size was 919, and the training epoch was 500. The optimal α and λ were 0.6 and 200, respectively.

MIT-BIH Arrhythmia Data Set

The LSTM model for the MIT-BIH arrhythmia data set had a single layer with 2 cells and 48 hidden units, a learning rate of 0.0001, a minibatch size of 640, and a training epoch of 2000. The optimal α was 0.2, and the optimal λ was 4.

Comparison Method

To prove that the UBS is effective for developing a proper selection function in an LSTM model with a reject option, we compared it with conventional batch normalization and a model without normalization. The false-positive and false-negative rates were also calculated, and a standard LSTM model without a selection function was used as the baseline.

Results

LSTM Performance for Prediction

The baseline models should be optimized for LSTM models without a selection function for each data set. Therefore, we validated the LSTM model prediction performance without any selection. The test accuracies of the LSTM models optimized without a selection step for the human activity recognition using smartphones data set and the MIT-BIH arrhythmia data set are 92.35% and 97.23% for each data set. The precision of the model was 91.72% and the recall was 91.54% for the Human Activity Recognition Using Smartphones data set. For the MIT-BIH arrhythmia data set, the precision of the model was 87.13% and the recall was 78.64%. The F1-score for each data set were 91.63% and 82.67%, respectively.

Coverage Violation

After setting the target coverage, the empirical coverage of the test set was calculated for each normalization method. The target coverage rates were obtained from a previous study [32]. As the target coverage is the target threshold, it should be set to a sufficiently reliable value. Therefore, the target coverages were set at 0.85, 0.90, and 0.95. The difference between the target coverage and the actual coverage value is called *coverage violation*, which estimates the extent to which the model can learn to select the samples as instructed by the target coverage hyperparameter. The experimental results for each data set are listed in Table 2. The coverage value was averaged for 5 different runs. As shown in Table 2, the empirical coverage with UBS produced superior results as they converged on the target coverage, whereas other normalization approaches showed relatively poor results.

Table 2. Empirical coverage of the human activity recognition (HAR) using smartphones and the Massachusetts Institute of Technology-Beth Israel Hospital (MIT-BIH) arrhythmia data sets by different normalization methods. Target coverage was set before training.

Target coverage	HAR using smartphones data set			MIT-BIH arrhythmia data set		
	Normalization method of selective prediction			Normalization method of selective prediction		
	UBS ^a	BN ^b	Without normalization ^c	UBS	BN	Without normalization
0.95, mean (SD)	0.9660 (0.0029)	0.9996 (0.0001)	0.9986 (0.0002)	0.9564 (0.0019)	0.9680 (0.0067)	1.0000 (0)
0.90, mean (SD)	0.9053 (0.0035)	0.9980 (0.0001)	0.9984 (0.0001)	0.9084 (0.0055)	0.9998 (0.0001)	1.0000 (0)
0.85, mean (SD)	0.8582 (0.0007)	0.9237 (0.0026)	0.9986 (0.0002)	0.8888 (0.0016)	0.9518 (0.0001)	1.0000 (0)
Average violation, %	0.98	7.38	9.85	1.79	7.32	10.00

^aUBS: unit-wise batch standardization.

^bBN: batch normalization (a normalization method using the mean and variance obtained from the input batch).

^cWithout normalization means that there was no normalization in the selection function structure.

Selective Risk (Error Rate)

The selective risks for each normalization method are presented in [Table 3](#). The selective risk value was averaged from 5 different runs. In the selective prediction model with LSTM,

the selective risk increased with coverage. UBS normalization achieved relatively superior performance with various target coverages compared with conventional batch normalization. If normalization was not applied, the risk varied widely.

Table 3. Selective risk of the human activity recognition (HAR) using smartphones and the Massachusetts Institute of Technology-Beth Israel Hospital (MIT-BIH) arrhythmia data sets by different normalization methods.

Target coverage	HAR using smartphones data set			MIT-BIH arrhythmia data set		
	Normalization method of selective prediction			Normalization method of selective prediction		
	UBS ^a	BN ^b	Without normalization ^c	UBS	BN	Without normalization
0.95, mean (SD)	0.1423 (0.0041)	0.1611 (0.0445)	0.1476 (0.0068)	0.1970 (0.0038)	0.2175 (0.0108)	0.2000 (0.4472)
0.90, mean (SD)	0.1232 (0.0042)	0.1283 (0.0067)	0.1312 (0.0139)	0.1791 (0.0050)	0.3200 (0.1095)	0.2000 (0.4472)
0.85, mean (SD)	0.1136 (0.0060)	0.1170 (0.0024)	0.1267 (0.0145)	0.1585 (0.0028)	0.1967 (0.0064)	0.2000 (0.4472)
Average risk	0.1264	0.1355	0.1352	0.1782	0.2447	0.2

^aUBS: unit-wise batch standardization.

^bBN: batch normalization (a normalization method using the mean and variance obtained from the input batch).

^cWithout normalization means that there was no normalization in the selection function structure.

False-Positive and False-Negative Rates

As the selective prediction model produced classification results only when it was confident about its own classification, we expected that both false-positive and false-negative rates would

decrease. The false-positive and false-negative rates of each data set were calculated from the results of the model that achieved the best performance among 5 different runs ([Tables 4 and 5](#)). The baseline models were well-optimized LSTM models without a selection function for each data set.

Table 4. False-positive rates of the human activity recognition (HAR) using smartphones and the Massachusetts Institute of Technology-Beth Israel Hospital (MIT-BIH) arrhythmia data sets by different normalization methods.

Target coverage	HAR using smartphones data set				MIT-BIH arrhythmia data set			
	Normalization method of selective prediction			General prediction ^a	Normalization method of selective prediction			General prediction
	UBS ^b	BN ^c	Without normalization ^d		UBS	BN	Without normalization	
0.95, %	2.04	2.59	2.65	N/A ^e	6.34	7.67	6.93	N/A
0.90, %	2.00	3.00	2.63	N/A	5.39	6.98	6.77	N/A
0.85, %	2.22	3.02	2.63	N/A	5.66	7.03	7.97	N/A
Average false-positive rate, %	2.09	2.87	2.64	2.89	5.80	7.23	7.22	6.44

^aGeneral prediction is the long short-term memory classification model's false-positive rate without a selection function.

^bUBS: unit-wise batch standardization.

^cBN: batch normalization (a normalization method using the mean and variance obtained from the input batch).

^dWithout normalization means that there was no normalization in the selection function structure.

^eN/A: not applicable.

Table 5. False-negative rates of the human activity recognition (HAR) using smartphones and the Massachusetts Institute of Technology-Beth Israel Hospital (MIT-BIH) arrhythmia data sets by different normalization methods.

Target coverage	HAR using smartphones data set				MIT-BIH arrhythmia data set			
	Normalization method of selective prediction			General prediction ^a	Normalization method of selective prediction			General prediction
	UBS ^b	BN ^c	Without normalization ^d		UBS	BN	Without normalization	
0.95, %	10.18	17.17	12.69	N/A ^e	18.82	23.33	20.78	N/A
0.90, %	10.72	15.04	13.05	N/A	16.48	20.94	20.31	N/A
0.85, %	10.85	14.46	12.94	N/A	16.41	21.44	23.91	N/A
Average false-negative rate, %	10.58	15.56	12.89	14.48	17.24	21.90	21.67	26.47

^aGeneral prediction is the long short-term memory classification model's false-positive rate without a selection function.

^bUBS: unit-wise batch standardization.

^cBN: batch normalization; which is a normalization method using the mean and variance obtained from the input batch.

^dWithout normalization means that there was no normalization in the selection function structure.

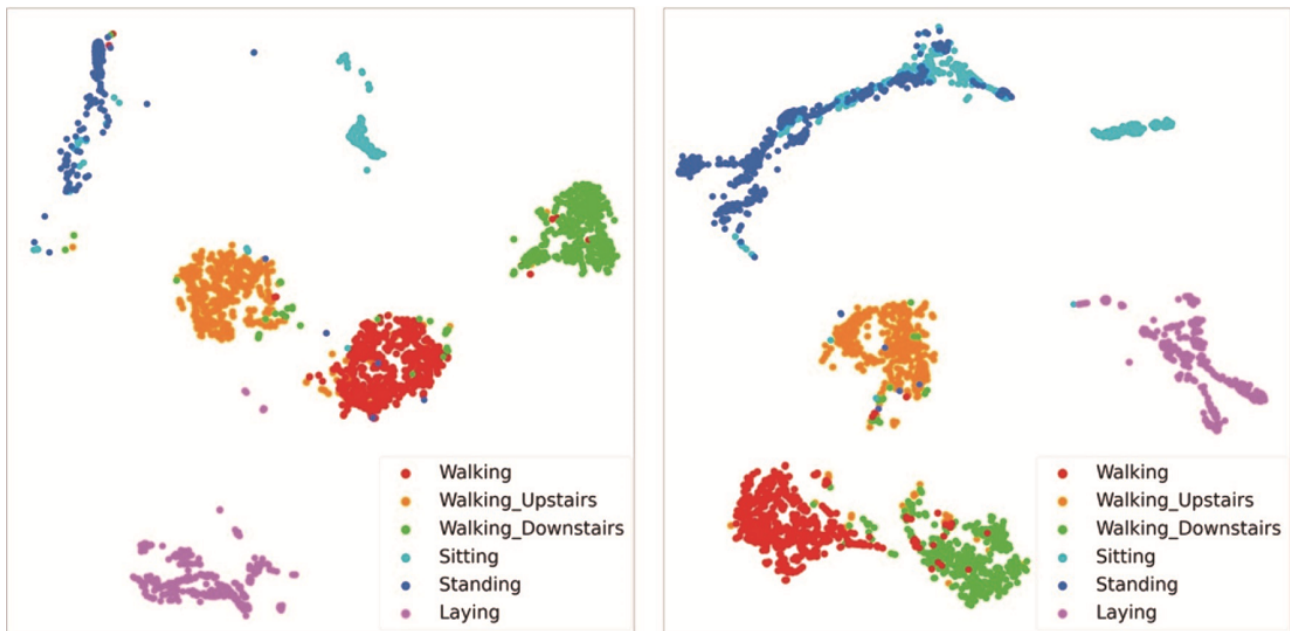
^eN/A: not applicable.

Learned Feature Representation

Figure 3 shows the visualization of the features learned from the LSTM models using t-distributed stochastic neighbor embedding [49]. Figure 3 (left) depicts the test set sample that was not rejected when the target coverage was set at 0.95. The data set used in the visualization was the test set for the human

activity recognition using smartphones data set. The *Sitting* (cyan) and *Standing* samples (blue) are more mixed in Figure 3 (right) than in Figure 3 (left). The *Walking_Down_Stairs* (green), *Walking_Up_Stairs* (orange), and *Walking* samples (red) are closely clustered in Figure 3 (left), whereas some of them overlap in Figure 3 (right).

Figure 3. t-Distributed stochastic neighbor embedding visualizations of learned features using all test samples in the human activity recognition using smartphones data set. Left: Long short-term memory with a reject option using unit-wise batch standardization results when the target coverage was 0.95. Rejected samples were not included in this figure. Right: long short-term memory model results without a reject option.



Discussion

Principal Findings

Our objective is to develop a selective prediction model using LSTM. The developed selective prediction model rejected samples using the confidence level of classifications. This selective prediction model with a reject option was trained to determine whether to obtain a classification based on targeted coverage. If the model's classification confidence was low, the model rejected the classification and did not apply information to backpropagate on samples. As a result, the selective prediction model was trained mainly using samples that had a sufficient confidence level, which guaranteed reliability and low error rates for samples that were not rejected. To implement selective prediction for LSTM, we conducted an experiment to identify a method of normalization that could improve the performance of the selection function.

In health care systems, high accuracy is important, but low false-positive and false-negative rates are also essential. To handle various time series data obtained from a health care system, we devised a selective prediction model with LSTM using an effective selection function and focused on the structure of the function. As shown in Table 1, the output of the *many-to-one* LSTM includes hidden-unit information. Our goal was to deal with LSTMs that have *many-to-one* structures, but conventional batch normalization normalizes all batches at once. To tackle this problem, we devised UBS as a special method of normalization that attempts to normalize each hidden unit in LSTM. The false-positive and false-negative rates for each data set were meaningful. For each target coverage, the selective prediction model with UBS was superior to the model with batch normalization and the model without normalization (Tables 4 and 5). These findings show that a selective function using UBS can decrease false-positive and false-negative rates. On this basis, we interpreted that the model with UBS can learn

class-specific features and consider which samples to reject in the training phase.

UBS also helped the model be trained based on target coverage and reduced selective risk. Using 2 public health data sets, the empirical coverage violation of the selective prediction was lower than that of the other 2 methods. The selection function with the UBS had the lowest selective risk (Table 3). The MIT-BIH arrhythmia data set results show that the coverage of the model without normalization was high regardless of the target coverage. These findings imply that the selective function without normalization did not perform as desired. We assumed that these results were based on whether the normalization methods considered hidden-unit characteristics of LSTM.

Regarding the learned feature representation, the classification model with the reject option differed from existing models. In Figure 3, a classification model with the reject option achieved relatively better classification performance than the conventional model without the reject option because the selective prediction LSTM model did not learn the features from samples with a low confidence level. As reported in a previous study [32], this suggests that representational capacity was not wasted because the model was trained mainly on samples with a high confidence level using selective prediction. Using this property, selective prediction allows humans to classify samples with low reliability and act as a second opinion in health care applications. In summary, the selective prediction model successfully classified samples based on high confidence-level features and simultaneously reduced the error rate by using the reject option.

Although our research supports the possibility of generating LSTM models with selective prediction, challenges remain. First, interpretation of the visualization of the learned features is limited in this study and needs to be addressed in further studies. Second, when LSTM was used for selective prediction, it was difficult to optimize parameters that control selection

functions, such as α and λ , for each data set. During the experiments, we used only 2 data sets for testing and targeted only the reject option to determine the confidence level of classifications. In future studies, efficient optimization methods should be devised and applied to various models using various data sets.

Conclusions

In this study, we developed LSTM classification models with a reject option to classify medical data time series. To develop the LSTM classification models with the reject option, UBS was applied. The UBS achieved superior performance (concerning coverage, risk, and false-positive and false-negative

rates) compared with 2 other methods of normalization in experiments using 2 public time series data sets.

If the performance in classifying nonrejected samples can be maximized by adjusting coverage or selective risks, humans can trust the output of a highly confident AI model and spend more time on other rejected samples (low confidence). The final performance (human+AI) can be maximized by appropriate automation using selective prediction.

To the best of our knowledge, this is the first study demonstrating the possibility of an LSTM classification model with a reject option for time series data. Our findings may apply to various other time series data sets that require reliability.

Acknowledgments

This research was supported by the Bio and Medical Technology Development Program of the National Research Foundation, which is funded by the Korean government, Ministry of Science and ICT (NRF-2017M3A9E1064781) and the Technology Innovation Program (Alchemist Project, 20012461) funded by the Korean Ministry of Trade, Industry, and Energy.

Authors' Contributions

This study was originally conceived by BRN. BRN developed a deep learning model and wrote draft of manuscript as a lead author. Data extraction and preprocessing was conducted by BRN and JYK. IYK and BHC jointly supervised this project as co-corresponding authors. All authors provided critical feedback and approved the final manuscript.

Conflicts of Interest

None declared.

References

1. Rong G, Mendez A, Bou Assi E, Zhao B, Sawan M. Artificial intelligence in healthcare: review and prediction case studies. *Engineering* 2020;6(3):291-301. [doi: [10.1016/j.eng.2019.08.015](https://doi.org/10.1016/j.eng.2019.08.015)]
2. Moon JH, Lee DY, Cha WC, Chung MJ, Lee KS, Cho BH, et al. Automatic stenosis recognition from coronary angiography using convolutional neural networks. *Comput Methods Programs Biomed* 2021;198:105819 [FREE Full text] [doi: [10.1016/j.cmpb.2020.105819](https://doi.org/10.1016/j.cmpb.2020.105819)] [Medline: [33213972](https://pubmed.ncbi.nlm.nih.gov/33213972/)]
3. Kim JY, Ro K, You S, Nam BR, Yook S, Park HS, et al. Development of an automatic muscle atrophy measuring algorithm to calculate the ratio of supraspinatus in supraspinous fossa using deep learning. *Comput Methods Programs Biomed* 2019;182:105063. [doi: [10.1016/j.cmpb.2019.105063](https://doi.org/10.1016/j.cmpb.2019.105063)] [Medline: [31505380](https://pubmed.ncbi.nlm.nih.gov/31505380/)]
4. Xue Q, Chuah MC. Explainable deep learning based medical diagnostic system. *Smart Health* 2019;13:100068. [doi: [10.1016/j.smhl.2019.03.002](https://doi.org/10.1016/j.smhl.2019.03.002)]
5. Colak E, Moreland R, Ghassemi M. Five principles for the intelligent use of AI in medical imaging. *Intensive Care Med* 2021;47(2):154-156. [doi: [10.1007/s00134-020-06316-8](https://doi.org/10.1007/s00134-020-06316-8)] [Medline: [33449134](https://pubmed.ncbi.nlm.nih.gov/33449134/)]
6. Quinn TP, Senadeera M, Jacobs S, Coghlan S, Le V. Trust and medical AI: the challenges we face and the expertise needed to overcome them. *J Am Med Inform Assoc* 2021;28(4):890-894 [FREE Full text] [doi: [10.1093/jamia/ocaa268](https://doi.org/10.1093/jamia/ocaa268)] [Medline: [33340404](https://pubmed.ncbi.nlm.nih.gov/33340404/)]
7. Hengstler M, Enkel E, Duelli S. Applied artificial intelligence and trust—the case of autonomous vehicles and medical assistance devices. *Technol Forecast Soc Change* 2016;105:105-120. [doi: [10.1016/j.techfore.2015.12.014](https://doi.org/10.1016/j.techfore.2015.12.014)]
8. Chow CK. An optimum character recognition system using decision functions. *IRE Trans Electron Comput* 1957;EC-6(4):247-254. [doi: [10.1109/tec.1957.5222035](https://doi.org/10.1109/tec.1957.5222035)]
9. Wang J, Chen Y, Hao S, Peng X, Hu L. Deep learning for sensor-based activity recognition: a survey. *Pattern Recognit Lett* 2019;119:3-11. [doi: [10.1016/j.patrec.2018.02.010](https://doi.org/10.1016/j.patrec.2018.02.010)]
10. Miotto R, Wang F, Wang S, Jiang X, Dudley JT. Deep learning for healthcare: review, opportunities and challenges. *Brief Bioinform* 2018;19(6):1236-1246 [FREE Full text] [doi: [10.1093/bib/bbx044](https://doi.org/10.1093/bib/bbx044)] [Medline: [28481991](https://pubmed.ncbi.nlm.nih.gov/28481991/)]
11. Lipton ZC, Kale DC, Elkan C, Wetzel RC. Learning to diagnose with LSTM recurrent neural networks. In: *Proceedings of the 4th International Conference on Learning Representations*. 2016 Presented at: ICLR '16; May 2-4, 2016; San Juan, Puerto Rico. [doi: [10.1093/acref/9780195301731.013.43262](https://doi.org/10.1093/acref/9780195301731.013.43262)]
12. Choi E, Schuetz A, Stewart WF, Sun J. Using recurrent neural network models for early detection of heart failure onset. *J Am Med Inform Assoc* 2017;24(2):361-370 [FREE Full text] [doi: [10.1093/jamia/ocw112](https://doi.org/10.1093/jamia/ocw112)] [Medline: [27521897](https://pubmed.ncbi.nlm.nih.gov/27521897/)]

13. Razavian N, Marcus J, Sontag D. Multi-task prediction of disease onsets from longitudinal laboratory tests. In: Proceedings of the 1st Machine Learning for Healthcare Conference. 2016 Presented at: PMLR '16; August 19-20, 2016; Los Angeles, CA p. 73-100.
14. Reddy BK, Delen D. Predicting hospital readmission for lupus patients: an RNN-LSTM-based deep-learning methodology. *Comput Biol Med* 2018;101:199-209. [doi: [10.1016/j.compbiomed.2018.08.029](https://doi.org/10.1016/j.compbiomed.2018.08.029)] [Medline: [30195164](https://pubmed.ncbi.nlm.nih.gov/30195164/)]
15. Şentürk Ü, Yücedağ I, Polat K. Repetitive neural network (RNN) based blood pressure estimation using PPG and ECG signals. In: 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies. 2018 Presented at: ISMSIT '18; October 19-21, 2018; Ankara, Turkey p. 1-4. [doi: [10.1109/ismsit.2018.8567071](https://doi.org/10.1109/ismsit.2018.8567071)]
16. Su P, Ding XR, Zhang YT, Liu J, Miao F, Zhao N. Long-term blood pressure prediction with deep recurrent neural networks. In: IEEE EMBS International Conference on Biomedical & Health Informatics. 2018 Presented at: BHI '18; March 4-7, 2018; Las Vegas, NV p. 323-328. [doi: [10.1109/bhi.2018.8333434](https://doi.org/10.1109/bhi.2018.8333434)]
17. Xu X, Jeong S, Li J. Interpretation of electrocardiogram (ECG) rhythm by combined CNN and BiLSTM. *IEEE Access* 2020;8:125380-125388. [doi: [10.1109/access.2020.3006707](https://doi.org/10.1109/access.2020.3006707)]
18. Rana A, Kim KK. ECG heartbeat classification using a single layer LSTM model. In: International SoC Design Conference. 2019 Presented at: ISOCC '19; October 6-9, 2019; Jeju, South Korea p. 267-268. [doi: [10.1109/isocc47750.2019.9027740](https://doi.org/10.1109/isocc47750.2019.9027740)]
19. Hernández F, Suárez LF, Villamizar J, Altuve M. Human activity recognition on smartphones using a bidirectional LSTM network. In: XXII Symposium on Image, Signal Processing and Artificial Vision. 2019 Presented at: STSIVA '19; April 24-26, 2019; Bucaramanga, Colombia p. 1-5. [doi: [10.1109/stsiva.2019.8730249](https://doi.org/10.1109/stsiva.2019.8730249)]
20. Hammerla NY, Halloran S, Plötz T. Deep, convolutional, and recurrent models for human activity recognition using wearables. In: Proceedings of the 25th International Joint Conference on Artificial Intelligence. 2016 Presented at: IJCAI '16; July 9-15, 2016; New York, NY p. 1533.
21. Chowdhury SS, Hasan MS, Sharmin R. Robust heart rate estimation from PPG signals with intense motion artifacts using cascade of adaptive filter and recurrent neural network. In: 2019 IEEE Region 10 Conference. 2019 Presented at: TENCON '19; October 17-20, 2019; Kochi, India p. 1952-1957. [doi: [10.1109/tencon.2019.8929692](https://doi.org/10.1109/tencon.2019.8929692)]
22. Zhao Y, Yang R, Chevalier G, Xu X, Zhang Z. Deep residual Bidir-LSTM for human activity recognition using wearable sensors. *Math Probl Eng* 2018;2018:1-13. [doi: [10.1155/2018/7316954](https://doi.org/10.1155/2018/7316954)]
23. Cordella LP, De Stefano C, Tortorella F, Vento M. A method for improving classification reliability of multilayer perceptrons. *IEEE Trans Neural Netw* 1995;6(5):1140-1147. [doi: [10.1109/72.410358](https://doi.org/10.1109/72.410358)] [Medline: [18263404](https://pubmed.ncbi.nlm.nih.gov/18263404/)]
24. De Stefano C, Sansone C, Vento M. To reject or not to reject: that is the question-an answer in case of neural classifiers. *IEEE Trans Syst, Man, Cybern C* 2000;30(1):84-94. [doi: [10.1109/5326.827457](https://doi.org/10.1109/5326.827457)]
25. El-Yaniv R, Wiener Y. Pointwise tracking the optimal regression function. In: Advances in Neural Information Processing Systems 25. 2012 Presented at: NIPS '12; December 3-8, 2012; Lake Tahoe, NV p. 2042-2050.
26. Geifman Y, El-Yaniv R. Selective classification for deep neural networks. In: Proceedings of the 31st International Conference on Neural Information Processing Systems. 2017 Presented at: NIPS '17; December 4-9, 2017; Long Beach, CA p. 4885-4894. [doi: [10.7551/mitpress/11474.003.0014](https://doi.org/10.7551/mitpress/11474.003.0014)]
27. Blundell C, Cornebise J, Kavukcuoglu K, Wierstra D. Weight uncertainty in neural network. In: Proceedings of The 32nd International Conference on Machine Learning. 2015 Presented at: ICML '15; July 6-11, 2015; Lille, France.
28. Lipton Z, Li X, Gao J, Li L, Ahmed F, Deng L. BBQ-networks: efficient exploration in deep reinforcement learning for task-oriented dialogue systems. In: Proceedings of the 32nd AAAI Conference on Artificial Intelligence. 2018 Presented at: AAAI '18; February 2-7, 2018; New Orleans, LA.
29. Houthoofd R, Chen X, Duan Y, Schulman J, De Turck F, Abbeel P. VIME: variational information maximizing exploration. In: Proceedings of the 30th International Conference on Neural Information Processing Systems. 2016 Presented at: NIPS '16; December 5-10, 2016; Barcelona, Spain p. 1117-1125. [doi: [10.1016/S0377-0427\(00\)00433-7](https://doi.org/10.1016/S0377-0427(00)00433-7)]
30. Fortunato M, Blundell C, Vinyals O. Bayesian recurrent neural networks. *arXiv (forthcoming)* 2017:1-14 [FREE Full text]
31. Cortes C, DeSalvo G, Mohri M. Learning with rejection. In: Proceedings of the 27th International Conference on Algorithmic Learning Theory. 2016 Presented at: ALT '16; October 19-21, 2016; Bari, Italy p. 67-82. [doi: [10.1007/978-3-319-46379-7_5](https://doi.org/10.1007/978-3-319-46379-7_5)]
32. Geifman Y, El-Yaniv R. SelectiveNet: a deep neural network with an integrated reject option. In: Proceedings of The 36th International Conference on Machine Learning. 2019 Presented at: ICML '19; June 10-15, 2019; Long Beach, CA.
33. Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Comput* 1997;9(8):1735-1780. [doi: [10.1162/neco.1997.9.8.1735](https://doi.org/10.1162/neco.1997.9.8.1735)] [Medline: [9377276](https://pubmed.ncbi.nlm.nih.gov/9377276/)]
34. El-Yaniv R, Wiener Y. On the foundations of noise-free selective classification. *J Mach Learn Res* 2010;11(53):1605-1641.
35. Potra FA, Wright SJ. Interior-point methods. *J Comput Appl Math* 2000;124(1-2):281-302. [doi: [10.1016/s0377-0427\(00\)00433-7](https://doi.org/10.1016/s0377-0427(00)00433-7)]
36. Ioffe S, Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift. In: Proceedings of the 32nd International Conference on Machine Learning. 2015 Presented at: ICML '15; July 6-11, 2015; Lille, France p. 448-456.
37. Wu Y, He K. Group normalization. In: Proceedings of the 15th European Conference on Computer Vision. 2018 Presented at: ECCV '18; September 8-14, 2018; Munich, Germany p. 3-19. [doi: [10.1007/978-3-030-01261-8_1](https://doi.org/10.1007/978-3-030-01261-8_1)]

38. Bulbul E, Cetin A, Dogru IA. Human activity recognition using smartphones. In: Proceedings of the 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies. 2018 Presented at: ISMSIT '18; October 19-21, 2018; Ankara, Turkey p. 1-6. [doi: [10.1109/ismsit.2018.8567275](https://doi.org/10.1109/ismsit.2018.8567275)]
39. Wan S, Qi L, Xu X, Tong C, Gu Z. Deep learning models for real-time human activity recognition with smartphones. *Mobile Netw Appl* 2019;25(2):743-755. [doi: [10.1007/s11036-019-01445-x](https://doi.org/10.1007/s11036-019-01445-x)]
40. Xia K, Huang J, Wang H. LSTM-CNN architecture for human activity recognition. *IEEE Access* 2020;8:56855-56866. [doi: [10.1109/access.2020.2982225](https://doi.org/10.1109/access.2020.2982225)]
41. Ge Z, Zhu Z, Feng P, Zhang S, Wang J, Zhou B. ECG-signal classification using SVM with multi-feature. In: The 8th IEEE International Symposium on Next-Generation Electronics. 2019 Presented at: ISNE '19; October 9-10, 2019; Zhengzhou, China. [doi: [10.1109/isne.2019.8896430](https://doi.org/10.1109/isne.2019.8896430)]
42. Desai U, Martis RJ, Nayak CG, Sarika K, Seshikala G. Machine intelligent diagnosis of ECG for arrhythmia classification using DWT, ICA and SVM techniques. In: 2015 Annual IEEE India Conference. 2015 Presented at: INDICON '15; December 17-20, 2015; New Delhi, India. [doi: [10.1109/indicon.2015.7443220](https://doi.org/10.1109/indicon.2015.7443220)]
43. Anguita D, Ghio A, Oneto L, Parra X, Reyes-Ortiz JL. A public domain dataset for human activity recognition using smartphones. In: Proceedings of 2013 European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. 2013 Presented at: ESANN '13; April 24-26, 2013; Bruges, Belgium p. 437-442.
44. Moody GB, Mark RG. The impact of the MIT-BIH arrhythmia database. *IEEE Eng Med Biol Mag* 2001;20(3):45-50. [doi: [10.1109/51.932724](https://doi.org/10.1109/51.932724)] [Medline: [11446209](https://pubmed.ncbi.nlm.nih.gov/11446209/)]
45. Goldberger AL, Amaral LA, Glass L, Hausdorff JM, Ivanov PC, Mark RG, et al. PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals. *Circulation* 2000;101(23):E215-E220. [doi: [10.1161/01.cir.101.23.e215](https://doi.org/10.1161/01.cir.101.23.e215)] [Medline: [10851218](https://pubmed.ncbi.nlm.nih.gov/10851218/)]
46. Kachuee M, Fazeli S, Sarrafzadeh M. ECG heartbeat classification: a deep transferable representation. In: 2018 IEEE International Conference on Healthcare Informatics. 2018 Presented at: ICHI '18; June 4-7, 2018; New York, NY p. 443-444. [doi: [10.1109/ichi.2018.00092](https://doi.org/10.1109/ichi.2018.00092)]
47. Meng L, Zhao B, Chang B, Huang G, Sun W, Tung F, et al. Interpretable spatio-temporal attention for video action recognition. In: 2019 IEEE/CVF International Conference on Computer Vision Workshop. 2019 Presented at: ICCVW '19; October 27-28, 2019; Seoul, South Korea p. 1513-1522. [doi: [10.1109/iccvw.2019.00189](https://doi.org/10.1109/iccvw.2019.00189)]
48. Zhang J, Fan DP, Dai Y, Anwar S, Saleh FS, Zhang T, et al. UC-Net: uncertainty inspired RGB-D saliency detection via conditional variational autoencoders. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020 Presented at: CVPR '20; June 13-19, 2020; Seattle, WA p. 8579-8588. [doi: [10.1109/cvpr42600.2020.00861](https://doi.org/10.1109/cvpr42600.2020.00861)]
49. Van der Maaten L, Hinton G. Visualizing data using t-SNE. *J Mach Learn Res* 2008;9(86):2579-2605.

Abbreviations

AI: artificial intelligence

CNN: convolutional neural network

ECG: electrocardiogram

LSTM: long short-term memory

MC: Monte Carlo

MIT-BIH: Massachusetts Institute of Technology-Beth Israel Hospital

RNN: recurrent neural network

UBS: unit-wise batch standardization

Edited by C Lovis; submitted 21.05.21; peer-reviewed by L Chen, D Oladele; comments to author 27.09.21; revised version received 16.11.21; accepted 02.01.22; published 15.03.22

Please cite as:

Nam B, Kim JY, Kim IY, Cho BH

Selective Prediction With Long Short-term Memory Using Unit-Wise Batch Standardization for Time Series Health Data Sets: Algorithm Development and Validation

JMIR Med Inform 2022;10(3):e30587

URL: <https://medinform.jmir.org/2022/3/e30587>

doi: [10.2196/30587](https://doi.org/10.2196/30587)

PMID:

Attribution License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work, first published in JMIR Medical Informatics, is properly cited. The complete bibliographic information, a link to the original publication on <https://medinform.jmir.org/>, as well as this copyright and license information must be included.