Data-Driven Predictive Modeling of Neuronal Dynamics using Long Short-Term Memory

Benjamin Plaster and Gautam Kumar*

Department of Chemical and Materials Engineering, University of Idaho, Moscow, Idaho 83844 USA

* Correspondence: gkumar@uidaho.edu; Tel.: 1-208-885-1587

Abstract: Modeling brain dynamics to better understand and control complex behaviors underlying various cognitive brain functions are of interests to engineers, mathematicians, and physicists from the last several decades. With a motivation of developing computationally efficient models of brain dynamics to use in designing control-theoretic neurostimulation strategies, we have developed a novel data-driven approach in a long short-term memory (LSTM) neural network architecture to predict the temporal dynamics of complex systems over an extended long time-horizon in future. In contrast to recent LSTM-based dynamical modeling approaches that make use of multi-layer perceptrons or linear combination layers as output layers, our architecture uses a single fully connected output layer and reversed-order sequence-to-sequence mapping to improve short time-horizon prediction accuracy and to make multi-timestep predictions of dynamical behaviors. We demonstrate the efficacy of our approach in reconstructing the regular spiking to bursting dynamics exhibited by an experimentally-validated 9-dimensional Hodgkin-Huxley model of hippocampal CA1 pyramidal neurons. Through simulations, we show that our LSTM neural network can predict the multi-time scale temporal dynamics underlying various spiking patterns with reasonable accuracy. Moreover, our results show that the predictions improve with increasing predictive time-horizon in the multi-timestep deep LSTM neural network.

Keywords: Long short-term memory; Brain dynamics; Data-driven modeling; Complex systems

---

1. Introduction

Our brain generates highly complex nonlinear responses at multiple temporal scales, ranging from few milliseconds to several days, in response to external stimulus [1–3]. One of the long-time interests in computational neuroscience is to understand the dynamics underlying various cognitive and non-cognitive brain functions by developing computationally efficient modeling and analysis approaches. In the last four decades or so, several advancements have been made in the direction of dynamical modeling and analysis of brain dynamics [4–6]. In the context of modeling the dynamics of single neurons, several modeling approaches, ranging from detailed mechanism-based biophysiological modeling to simplified phenomenological/probabilistic modeling, have been developed to understand the diverse firing patterns (e.g., simple spiking to bursting) observed in electrophysiological experiments [7,8]. These models provide a detailed understanding of various ionic mechanisms that contribute to generate specific spiking patterns as well as allow to perform large-scale simulations to understand the dynamics underlying cognitive behaviors. However, most of these models are computationally expensive from the perspective of developing novel real-time neurostimulation strategies for controlling neuronal dynamics at single neurons and network levels. In this paper, we investigate purely data-driven long short-term memory (LSTM) based recurrent neural network (RNN) architectures in multi-timestep predictions of single neuron's dynamics for the use in developing novel neurostimulation strategies in an optimal control framework.

Availability of an abundant amount of data and advances in machine learning have recently revolutionized the field of predictive data-driven dynamical modeling of complex systems using neural networks (NNs) and deep learning approaches. Various nonlinear system identification approaches

have been developed to map static input-output relations using multi-layer perceptrons (MLPs) [9], [10], [11], [12] and their variations [13], [14]. Reinforcement learning has recently been explored in robotics dynamical modeling in [15]. NN architectures that make use of vanilla recurrent neural network (RNNs) elements have also been explored for nonlinear system identification and modeling in [16], [17], [18]. However, network architectures that make use of vanilla recurrent layers often suffer from the exploding or vanishing gradient problem when used to model dynamics over long time series horizons [19]. In [18], a highly specialized multi-phase training algorithm was used to ensure that the network did not suffer from this problem. LSTM based approaches to modeling dynamical systems [20], [21], [22] mitigate the vanishing gradient problem but suffer from poor early trajectory predictive performance when using long predictive horizons [23], [20]. LSTMs have been used to model high-dimensional chaotic systems [24], but these studies have been limited to single step prediction applications. Additionally, machine learning techniques have begun to be explored in neuroscientific modeling applications. Multi-layer Spiking Artificial Neural Networks (SANNs) for use in spatiotemporal spike pattern transformations have been developed in [25] and [26]. This is achieved by using novel approximations and surrogates of the partial derivatives of the spike train functions with respect to the weights. This partial derivative is typically problematic when used in backpropagation, as it is undefined at spike times for many neuronal models, rendering it incompatible with traditional backpropagation-based approaches. In [27], a novel gated recurrent unit (GRU) based encoder/decoder approach is used to learn and predict neuronal population dynamics and kinematic trajectories from single-trial spike train data.

In this paper, we have developed a novel deep LSTM neural network architecture, which can make multi-timestep predictions in large-scale dynamical systems. In particular, we use a reversed sequence-to-sequence mapping technique, developed for language translation applications of multi-layer LSTM networks in [28], and generalize the application of this technique to dynamical systems time-series forecasting. Figure 1 illustrates our overall approach.
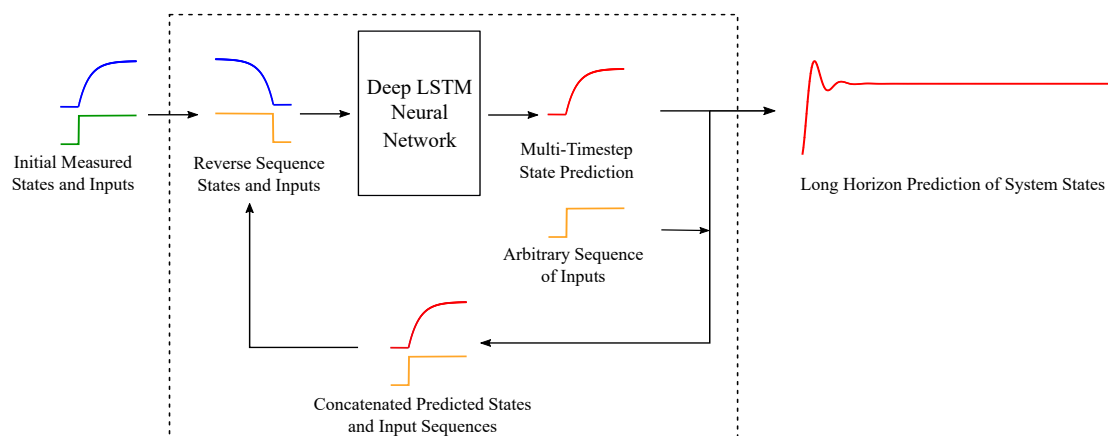
Figure 1. A schematic illustrating the overall data-driven approach developed in this paper for multi-timestep predictions of high-dimensional dynamical systems' behavior over a long time-horizon. An initial sequence of states and inputs are fed to the "Stacked LSTM Network" in a reverse-order for multi-timestep prediction of the system's states ("Reverse-order sequence-to-sequence mapping"). The predicted output from each stacked LSTM network is concatenated with the next sequence of inputs and fed into the next stacked LSTM network in a reverse-order to increase the predictive horizon. This process is iterated an arbitrary number of times, creating long dynamical predictions.

In contrast to existing approaches in modeling dynamical systems using neural networks, our architecture uses (1) stacked LSTM layers in conjunction with a single densely connected layer

to capture temporal dynamic features as well as input/output features, (2) sequence-to-sequence mapping, which enables multi-timestep predictions, and (3) reverse ordered input and measured state trajectories to the network, resulting in highly accurate early predictions and improved performance over long horizons. We show the efficacy of our developed approach in making stable multi-timestep predictions of various firing patterns exhibited by hippocampal CA1 pyramidal neurons, obtained from simulating an experimentally validated highly nonlinear 9-dimensional Hodgkin-Huxley model of CA1 pyramidal cell dynamics, over long time-horizons. Our approach is contingent on the network being trained on the entire state vector of the neuronal model.

The remaining paper is organized as follows. In Section 2, we describe our developed deep LSTM neural network architecture and methodological approach to data-driven multi-timestep predictions of dynamical systems. We show the efficacy of our approach in making stable multi-timestep predictions over long time-horizons of neuronal dynamics in Section 3 which is followed by a thorough discussion on the limitations of our approach in Section 4.

2. Neural Network Architecture, Algorithm and Approach

In Section 2.1, we describe our developed deep LSTM neural network architecture which combines stacked LSTMs with a fully-connected dense output layer. We describe the sequence-to-sequence mapping with reversed order input sequences used in this paper in Section 2.2. In Section 2.3, we provide the details on the synthetic data used to train our networks. Finally, in Section 2.4, we provide the details on the approach used to train the developed neural network architecture.

2.1. Deep LSTM Neural Network Architecture

Long short-term memory (LSTM) neural networks [29] are a particular type of recurrent neural networks (RNNs) which mitigate the vanishing or exploding gradient problem during the network training while capturing both the long-term and the short-term temporal features in sequential time-series data processing [19]. Specifically, LSTM uses multiple gating variables that control the flow of information of a hidden cell state and assign temporal importance to the dynamical features that are present in the time series data flowing through the cell state. Figure 2 shows a schematic illustrating the internal gating operation in a single LSTM cell.
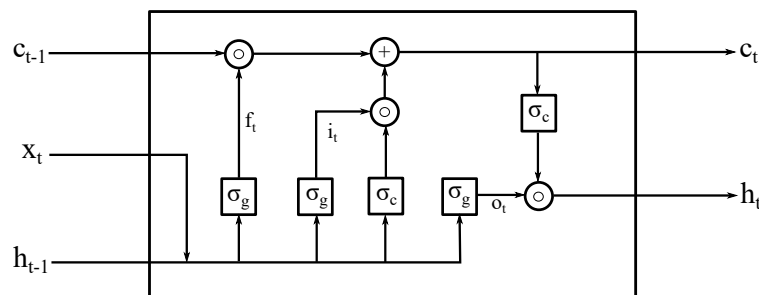
Figure 2. A schematic illustrating the internal gating operation in a single LSTM cell. The "+" represents an additive operation and the "∘" represents a multiplicative operation. σ_g is the sigmoidal activation function and σ_c is the hyperbolic tangent activation function.

A forward pass of information through a single LSTM cell is described by the following cell and gating state equations (reference):

$$c_t = f_t \circ c_{t-1} + i_t \circ \sigma_c(W_c x_t + U_c h_{t-1} + b_c), \tag{1a}$$

$$h_t = o_t \circ \sigma_c(c_t). \tag{1b}$$

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f), \tag{1c}$$

$$i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i), \tag{1d}$$

$$o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o), \tag{1e}$$

In equations (1a)-(1e), $c_t \in \mathbb{R}^h$ and $h_t \in \mathbb{R}^h$ represent the cell state vector and the hidden state vector, respectively, at time t. $f_t \in \mathbb{R}^h$, $i_t \in \mathbb{R}^h$, and $o_t \in \mathbb{R}^h$ are the "forget gate", "input gate", and "output gate" activation vector, respectively, at time t. $x_t \in \mathbb{R}^d$ is the input vector to the LSTM unit at time t, and $h_{t-1} \in \mathbb{R}^h$ is the previous time step hidden state vector passed back into the LSTM unit at time t. The matrices W_f, W_i, and W_o represent the input weights for the "forget gate", "input gate", and "output gate", respectively. The matrices U_f, U_i and U_o represent the weights of the recurrent connections for the "forget gate", "input gate", and "output gate", respectively. The vectors b_f b_i, and b_o represent the "forget gate", "input gate", and "output gate" biases, respectively. ∘ represents the element-wise multiplication. The function σ_g represents the sigmoidal activation function, and σ_c is the hyperbolic tangent activation functions.

In this paper, we use stacked LSTM network integrated with a fully connected feedforward output layer to make multi-timestep state predictions. The use of a single feedforward dense output layer allows the network to effectively learn the static input-output features, while the stacked LSTM network captures the temporal dynamical features. To appropriately select the optimum dimensionality of the hidden states in a single hidden layer, we systematically varied the number of hidden states in a sequence of $\{n, n^2, 2n^2, 4n^2, \cdots\}$, where n is the dimension of the system's state and evaluated the training performance for each case. We found that for our application ($n = 9$), a hidden state dimensionality of $4n^2 = 324$ was optimal in learning dynamical behaviors while avoiding overfitting. To select the number of hidden layers, we systematically increased the number of hidden layers of identical hidden state dimensionality (i.e., 324 states) and compared the network performance during the training. We found that increasing the number of hidden layers beyond 3 layers did not improve the network performance on the training and validation dataset. Thus, we fixed the number of hidden layers to 3 in our study. Throughout this paper, we utilized stateless LSTMs which reset the internal cell and hidden states to zero after processing and performing gradient descent for a given minibatch. We initialized the network weights using the Xavier method [30]. Specifically, the initial weights were drawn from a uniform distribution using

$$W_{ij} \sim \mathcal{U}\left(-\frac{6}{\sqrt{n_j + n_{j+1}}}, \frac{6}{\sqrt{n_j + n_{j+1}}}\right), \tag{2}$$

where n_j is the dimensionality of the input units in the weight tensor, and n_{j+1} is the dimensionality of the output units in the weight tensor.

To generate a long time-horizon dynamical prediction beyond the multi-timestep prediction by a single stacked deep LSTM neural network (shown as "Deep LSTM" in Figure 3), we used an iterative approach as described here. We made copies of the trained single stacked LSTM network and connected them in the feedforward manner in a sequence. We concatenated the sequence of predicted output from the previous stacked LSTM network with an equivalent length sequence of new inputs to the system and fed them in the reverse sequence order to the next stacked LSTM network. Figure 3 illustrates this iterative approach.

2.2. Sequence to Sequence Mapping with Neural Networks

To make multi-timestep predictions of dynamical systems' outputs using the deep LSTM neural network architecture described in the previous section (Section 2.1), we formulate the problem of mapping trajectories of the network inputs to the trajectories of the predicted outputs as a reverse order sequence-to-sequence mapping problem. The central idea of the reverse order sequence-to-sequence mapping approach is to feed the inputs to the network in reverse order such that the network perceives the first input as the last and the last input as the first. Although this approach has been developed and applied in language translation applications [28], it has never been considered in the context of predicting dynamical systems behaviors from time-series data. Figure 4 illustrates the basic idea
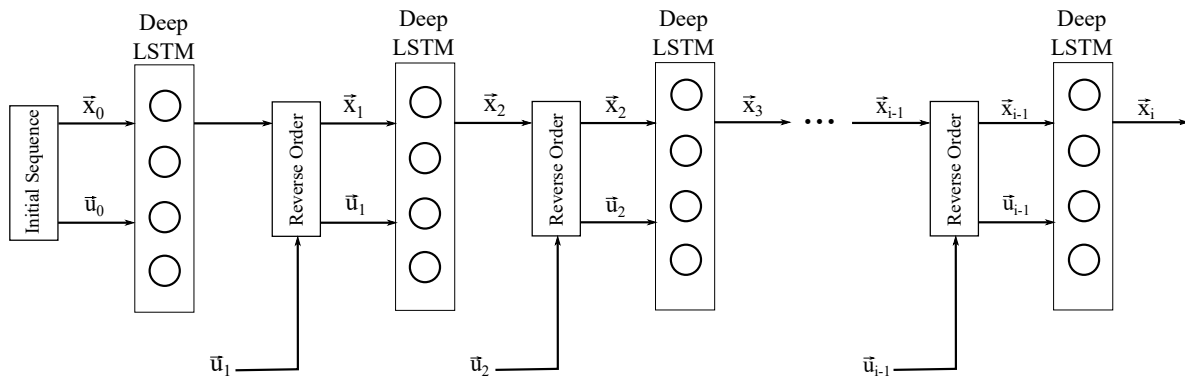
Figure 3. Iterative prediction of the system's outputs over a long time-horizon. Each "Deep LSTM" receives the predicted sequence of outputs from the previous "Deep LSTM" and an equivalent length of new system's inputs in reverse order and predict the next sequence of outputs of same time duration in future.

of the reverse order sequence-to-sequence mapping approach for translating letters (inputs) to their numerical indices (outputs).
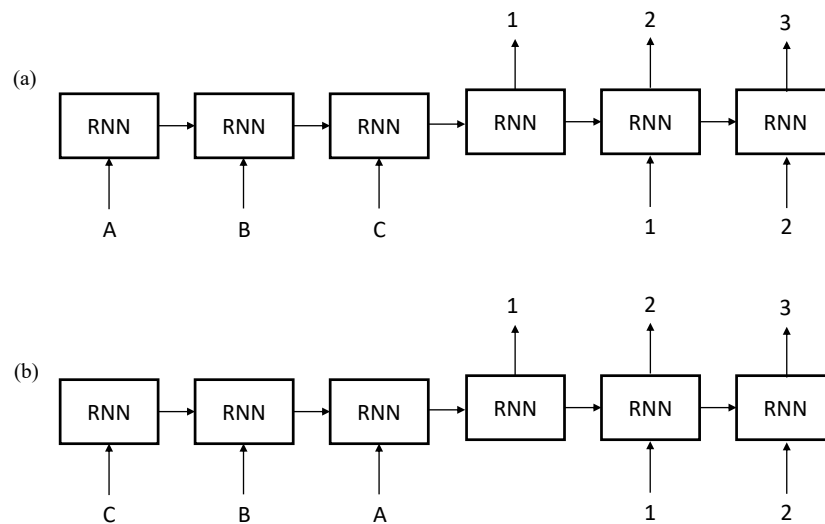
Figure 4. Forward and reversed sequence-to-sequence mapping approach for translating letters (inputs) to their numerical indices (outputs) in recurrent neural network (RNN). (a) shows the forward sequence-to-sequence mapping approach. The input is fed into the network in the same sequence as the desired output. The "distance" between all corresponding inputs and outputs is uniform. (b) shows the reversed sequence-to-sequence mapping approach. This approach introduces a temporal symmetry between input and output sequences while keeping the average "distance" between the corresponding inputs and outputs same as the forward approach. As shown in (b), $A \rightarrow 1$ is the shortest "distance" to map, $B \rightarrow 2$ the second, and $C \rightarrow 3$ the furthest.

As shown in Figure 4, in the forward sequence-to-sequence mapping approach (Figure 4(a)), i.e., $A, B, C \rightarrow 1, 2, 3$, the distance between all mappings is same (i.e., 3 "units"). In the reverse sequence-to-sequence mapping approach (Figure 4(b)), the network receives the input in a reverse order to map to the target output sequence, i.e., $C, B, A \rightarrow 1, 2, 3$. As noted here, the average distance between the mappings remains the same for both approaches (i.e. 3 "units") but the reverse order approach introduces short and long-term symmetric temporal dependencies between inputs and outputs. These short and long-term symmetric temporal dependencies provide improved predictive performance over long temporal horizons [28].

2.3. Synthetic Data

Hippocampal CA1 pyramidal neurons exhibit various multi-timescale firing patterns (from simple spiking to bursting) and play an essential role in shaping spatial and episodic memory [31]. In the last two decades, several biophysiological models of the CA1 pyramidal (CA1Py) neurons ranging from single compartmental biophysiological and phenomenological models [32–34] to detailed morphology-based multi-compartmental models [35–41] have been developed to understand the contributions of various ion-channels in diverse firing patterns (e.g., simple spiking to bursting) exhibited by the CA1Py neurons.

In this paper, we use an experimentally validated 9-dimensional nonlinear model of CA1 pyramidal neuron in the Hodgkin-Huxley formalism given in [32] to generate the synthetic data for the network training and validation. The model exhibits several different bifurcations to the external stimulating current and has shown its capability in generating diverse firing patterns observed in electrophysiological recordings from CA1 pyramidal cells under various stimulating currents. Figure 5 shows three different firing patterns generated from this model based on the three different regimes of the applied input currents.
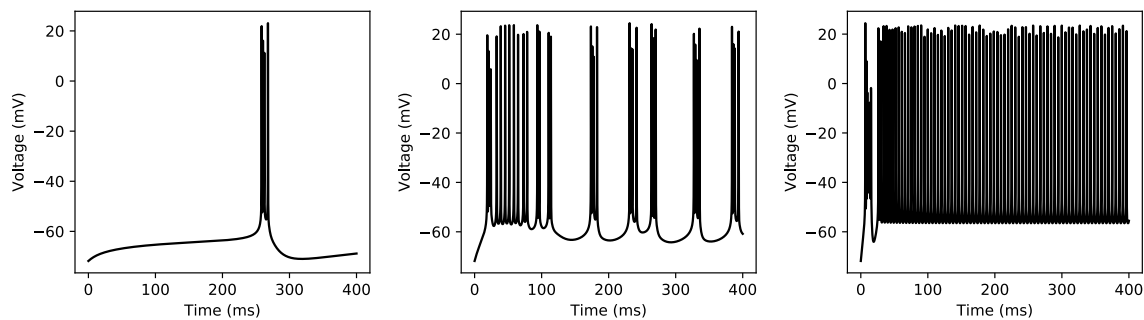
Figure 5. Diversity in the spiking patterns of hippocampal CA1 pyramidal neurons to applied currents. (a) Regular bursting in response to the external current of 0.23 nA. (b) Irregular bursting in response to the external current of 1.0 nA. (c) Plateau potentials followed by regular spiking in response to the external current of 3.0 nA.

To construct the synthetic training and validation dataset for the deep LSTM neural networks we designed in this paper with different predictive horizons, we simulated the Hodgkin-Huxley model of CA1 pyramidal neuron given in [32] (see Appendix A for the details of the model) for 1000 ms duration for 2000 constant stimulating currents, sampled uniformly between $I = 0.0$ nA and $I = 3.0$ nA. From these 2000 examples, we randomly and uniformly drew 50 samples (i.e., 10^4 data points) of the desired predictive horizon as the input/output sequence data for training and validation. As described in Section 2.4, we used 1/32 of these data points for validation, i.e., 96,875 data points for the training and 3,125 data points for the validation. Since our deep LSTM neural network takes an initial sequence of outputs of appropriate predictive horizon length (i.e., $N_p = 1, 50, 100, 200$) as an input sequence to make the next time-horizon prediction of equivalent length of sequence, we assume that this initial output sequence data is available to the deep LSTM neural network throughout our simulations.

2.4. Network Training

We formulated the following optimization problem to train a set of network weights θ:

$$\theta^* = \arg\min_{\theta} \mathcal{L}(\theta), \tag{3}$$

where the loss function $\mathcal{L}(\theta)$ is given by

$$\mathcal{L}(\theta) = \frac{1}{N_P} \sum_{k=0}^{N_P} (\vec{x}(k) - \hat{x}(k|\theta))^T (\vec{x}(k) - \hat{x}(k|\theta)). \tag{4}$$

Here N_P represents the length of horizon over which the predictions are made, $\vec{x}(k)$ is the known state vector at time step k, and $\hat{x}(k|\theta)$ is the neural network's prediction of the state vector at time k, given θ.

To solve the optimization problem (3)-(4), we used the standard supervised backpropagation learning algorithm [42], [43], [44] along with the Adaptive Moment Estimation (Adam) method [45]. The Adam method is a first-order gradient-based optimization algorithm and uses lower-order moments of the gradients between layers to optimize a stochastic objective function.

Given the network parameter $\theta^{(i)}$ and the loss function $\mathcal{L}(\theta)$, where i represents the algorithm's training iteration, the parameter update is given by [45]

$$m_\theta^{(i+1)} \leftarrow \beta_1 m_\theta^{(i)} + (1 - \beta_1) \nabla_\theta \mathcal{L}^{(i)} \tag{5}$$

$$v_\theta^{(i+1)} \leftarrow \beta_2 m_\theta^{(i)} + (1 - \beta_2)(\nabla_\theta \mathcal{L}^{(i)})^2 \tag{6}$$

$$\hat{m}_\theta = \frac{m_\theta^{(i+1)}}{1 - (\beta_1)^{i+1}} \tag{7}$$

$$\hat{v}_\theta = \frac{v_\theta^{(i+1)}}{1 - (\beta_2)^{(i+1)}} \tag{8}$$

$$\theta^{(i+1)} \leftarrow \theta^{(i)} - \eta \frac{\hat{m}_\theta}{\sqrt{\hat{v}_\theta} + \epsilon} \tag{9}$$

where m_θ is the first moment of the weights in a layer, v_θ is the second moment of the weights in a layer, η is the learning rate, β_1 and β_2 are the exponential decay rates for the moment estimates, ∇ is the differential gradient operator, and ϵ is a small scalar term to help numerical stability. Throughout this work, we used $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\eta = 0.001$ [45].

It should be noted that there is a tradeoff between the predictive time-horizon of deep LSTM neural network and the computational cost involved in training the network over the predictive horizon. As the predictive horizon increases, the computational cost of training the network over that horizon increases significantly for an equivalent number of examples. To keep the computational tractability in our simulations, all networks with long predictive horizons (i.e., $N_P = 50, 100, 200$) were trained for 200 epochs except the one-step predictive network which was trained for 1,000 epochs.

For all training sets throughout this paper, we used the validation to training data ratio as $1/32$. We set the minibatch size for training to 32. We performed all the training and computation in the TensorFlow computational framework on a discrete server running CentOS 7 with twin Nvidia GTX 1080Ti GPUs equipped with 11 Gb of VRAM.

3. Simulation Results

In this section, we present our simulation results on predicting the multi-timescale spiking dynamics exhibited by hippocampal CA1 pyramidal neurons over a long time-horizon using our developed deep LSTM neural network architecture described in Section 2. We trained 4 LSTM networks for making one timestep prediction (equivalently, 0.1 ms), 50 timesteps prediction (equivalently, 5 ms), 100 timesteps prediction (equivalently, 10 ms), and 200 timesteps prediction (equivalently, 20 ms). Figure 6 shows the training and validation loss for these 4 LSTM networks.

Using the iterative approach described in Section 2.1, we simulated the LSTM networks over 500 ms of time duration under different initial conditions and stimulating input currents between three different regimes of dynamical responses ("Regular Spiking" ($I \in [2.3, 3.0]$ nA), "Irregular Bursting"
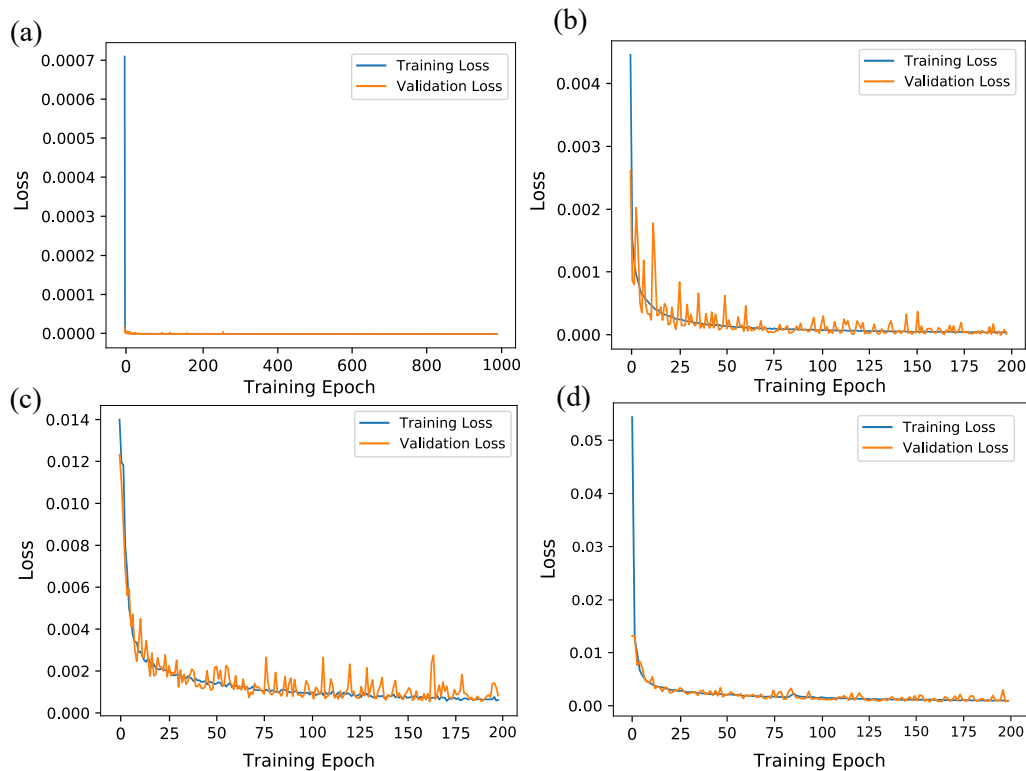
Figure 6. Training and validation loss for the deep LSTM neural network with multi-timestep predictive horizon. (a) 1 timestep predictive horizon. (b) 50 timesteps predictive horizon. (c) 100 timesteps predictive horizon. (d) 200 timesteps predictive horizon.

($I \in [0.79, 2.3)$ nA), and "Regular Bursting" ($I \in [0.24, 0.79)$) nA) and compared the predicted state trajectories with the Hodgkin-Huxley model.

3.1. Regular Spiking

In this section, we demonstrate the efficacy of our trained deep LSTM neural network over the range of external current between 2.3 nA and 3 nA in predicting the regular spiking dynamics shown by the biophysiological Hodgkin-Huxley model of CA1 pyramidal neuron in response to the external current $I \geq 2.3$ nA. In this range, we observe firing rates between approximately 165 Hz and 187 Hz. For clarity, we here show our results only for the membrane potential traces. We provide the complete set of simulation results on the LSTM network performance in predicting the dynamics of all the 9 states of the Hodgkin-Huxley model in Appendix B.1 (see Figures A1, A2, A3, A4, and A5).

Figure 7 shows a comparison of the membrane potential traces simulated using the Hodgkin-Huxley model and the 4 different predictive horizons of the LSTM network (i.e., 1 timestep, 50 timesteps, 100 timesteps, and 200 timesteps, which we represent as $N_p = 1, 50, 100, 200$) for the external stimulating current $I = 3.0$ nA. Note that all the simulations are performed using the same initial condition as provided in Appendix A. Since our LSTM network uses the initial sequence of outputs of appropriate predictive horizon (i.e., $N_p = 1, 50, 100, 200$) from the Hodgkin-Huxley model to make future time predictions, the LSTM network predictions (shown by dashed red line) start after 0.1 ms, 5 ms, 10 ms, and 20 ms in Figure 7a, Figure 7b, Figure 7c, and Figure 7d respectively.

As shown in Figure 7, the iterative prediction of the membrane potential traces by the LSTM network didn't differ significantly over a short time horizon (up to 200 ms) for $N_p = 1, 50, 100, 200$, but it significantly improved afterward with the increased predictive horizon of the LSTM network (i.e., $N_p = 1$ to $N_p = 200$). In particular, the LSTM network performance significantly improved in predicting the timing of the occurrence of spikes, but the magnitude of the membrane potential traces during spikes degraded as we increased N_p from 1 to 200. For clarity, we also computed
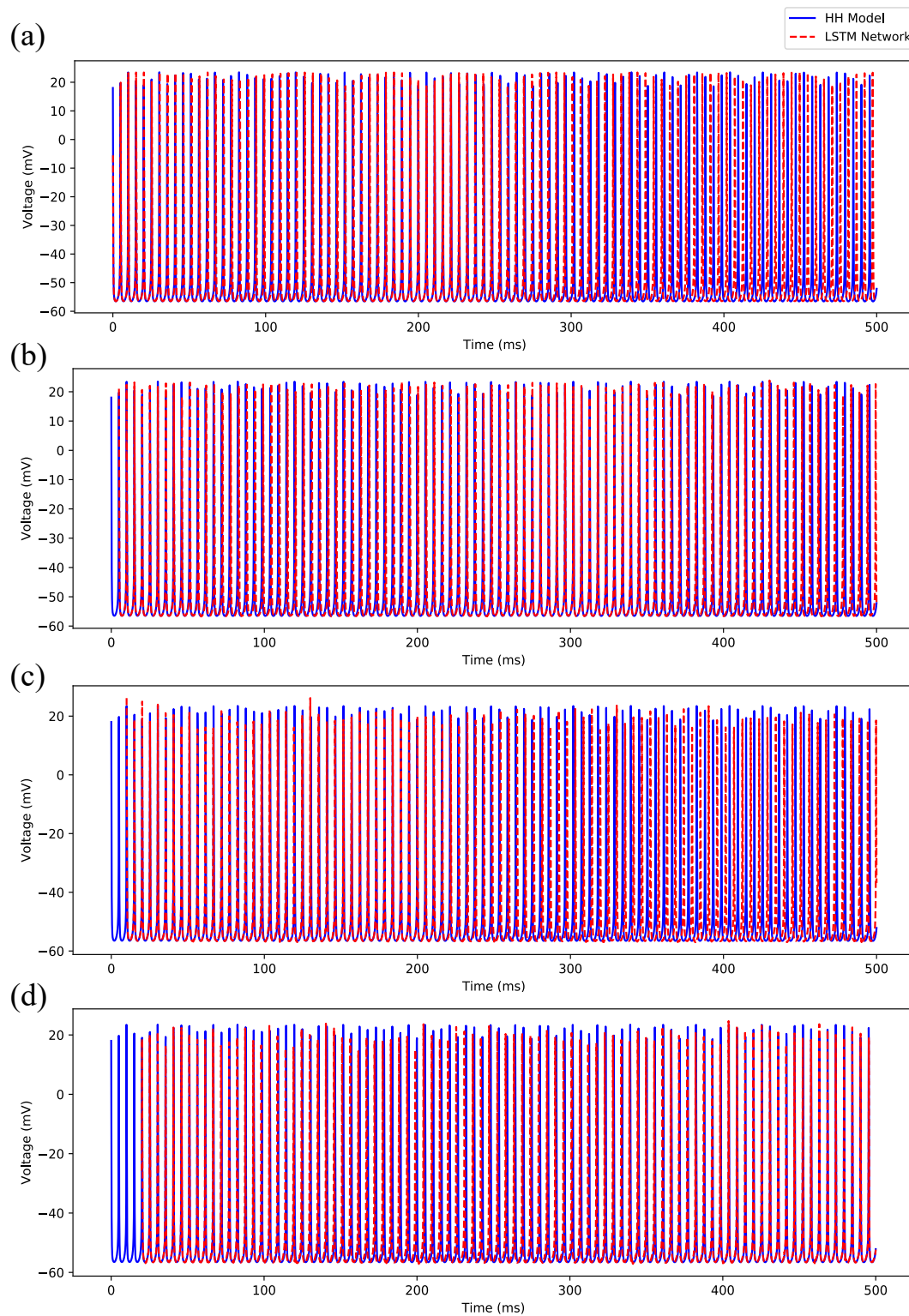
Figure 7. Comparison of predicted membrane potential traces by the deep LSTM neural network ("LSTM Network") to the regular spiking pattern exhibited by the Hodgkin-Huxley model ("HH Model") in response to the external stimulating current $I = 3.0$ nA. (a) Prediction using 1 timestep predictive LSTM network ($N_p = 1$). (b) Prediction using 50 timesteps predictive LSTM network ($N_p = 50$). (c) Prediction using 100 timesteps predictive LSTM network ($N_p = 100$). (d) Prediction using 200 timesteps predictive LSTM network ($N_p = 200$).

the time-averaged root mean squared error (RMSE) of the membrane potential traces between the Hodgkin-Huxley model and the LSTM network for $N_p = 1, 50, 100, 200$ over 500 ms of simulation time. Figure 8(a) shows that the time-averaged RMSE decreased consistently with the increased predictive horizon of the LSTM network. Overall, these results show that our LSTM network with a longer predictive horizon prefers to capture temporal correlations more accurately over the amplitude while LSTM network with a shorter predictive horizon prefers to capture the amplitude more accurately over the temporal correlations.
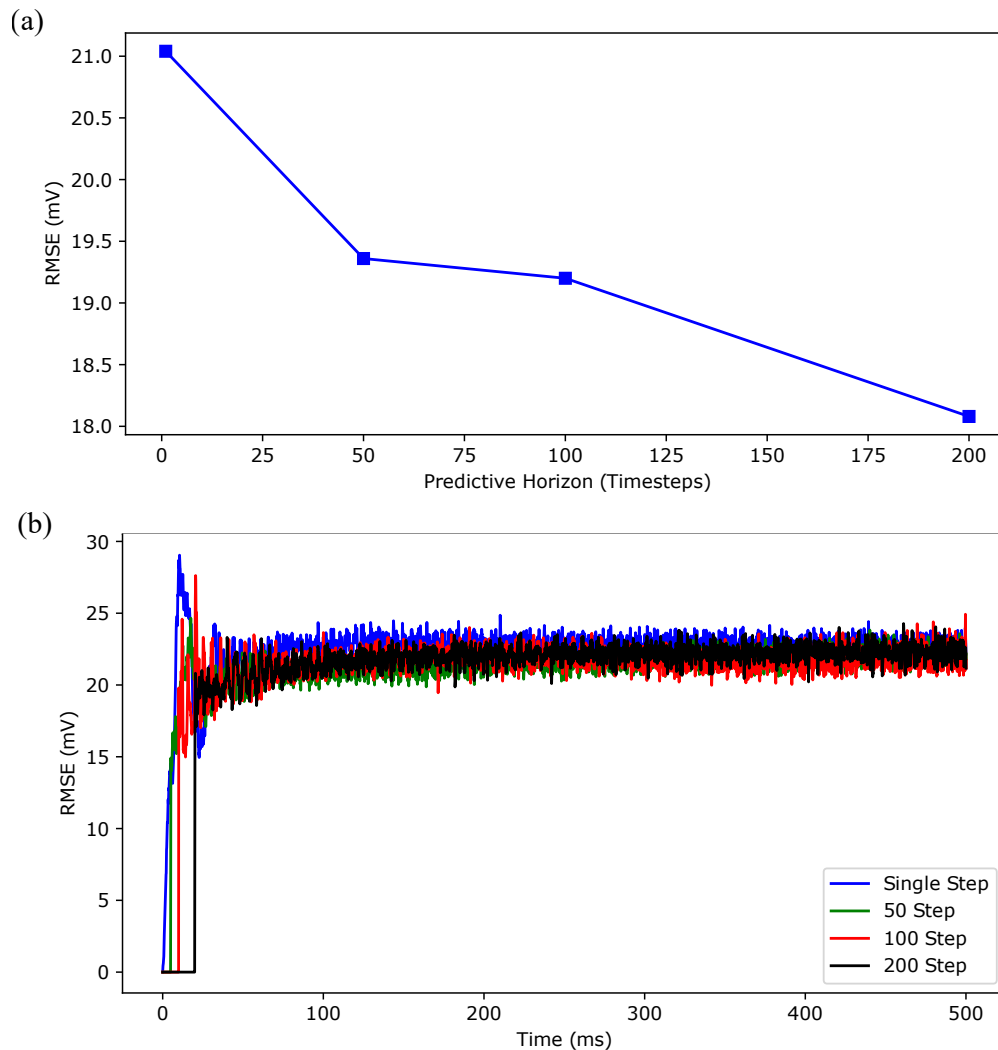
Figure 8. The effect of the length of predictive horizon of the deep LSTM neural network on the accuracy of regular spiking patterns prediction. (a) shows the time-averaged root mean squared error (RMSE) versus predictive horizon of the LSTM network ($N_p = 1, 50, 100, 200$) for the simulation results shown in Figure 7. (b) shows the RMSE versus simulation time for 5000 independent realizations, drawn from the predicted membrane potential trajectories of 50 randomly selected stimulating currents from a Uniform distribution $\mathcal{U}(2.3, 3.0)$ and 100 random initial conditions for each stimulating current.

To systematically evaluate whether the designed LSTM networks provide reasonable predictions of the membrane potential traces of the regular spiking dynamics across the range of external input currents between 2.3 nA and 3.0 nA, we performed simulations for 50 random stimulating currents drawn from a Uniform distribution $\mathcal{U}(2.3, 3.0)$. For each stimulating current, we chose 100 initial conditions drawn randomly from the maximum and minimum range of the Hodgkin-Huxley state variables (Note that the network was not trained over this wide range of initial conditions). Figure 8(b) shows the LSTM network performance, represented in terms of the root mean squared error vs time

over 5000 realizations, for $N_p = 1, 50, 100, 200$. As shown in this figure, the root mean squared error decreased with the increased predictive horizon of the LSTM network for all time, which is consistent with the result shown in Figure 8(a).

In conclusion, these results suggest that our deep LSTM neural network with a longer predictive horizon feature can predict the regular (periodic) spiking patterns exhibited by hippocampal CA1 pyramidal neurons with high accuracy over a long-time horizon.

3.2. Irregular Bursting

In this section, we demonstrate the efficacy of our trained deep LSTM neural network over the range of external current between 0.79 nA and 2.3 nA in predicting the irregular bursting dynamics shown by the biophysiological Hodgkin-Huxley model of CA1 pyramidal neuron in response to the external current $I \in [0.79, 2.3)$ nA. In this range, we observe firing rates between approximately 53 Hz and 164 Hz. For clarity, we here show our results only for the membrane potential traces. We provide the complete set of simulation results on the LSTM network performance in predicting the dynamics of all the 9 states of the Hodgkin-Huxley model in Appendix B.2 (see Figures A6, A7, A8, A9, and A10).

Figure 9 shows a comparison of the membrane potential traces simulated using the Hodgkin-Huxley model and the 4 different predictive horizons of the LSTM network (i.e., $N_p = 1, 50, 100, 200$) for the external stimulating current $I = 1.5$ nA. Note that all the simulations are performed using the initial condition used for $I = 3.0$ nA in Figure 7. Since our LSTM network uses the initial sequence of outputs of appropriate prediction horizon (i.e., $N_p = 1, 50, 100, 200$) from the Hodgkin-Huxley model to make future time predictions, the LSTM network predictions (shown by dashed red line) start after 0.1 ms, 5 ms, 10 ms, and 20 ms in Figure 9a, Figure 9b, Figure 9c, and Figure 9d respectively.

As shown in Figure 9, the LSTM performance significantly improved in predicting the timing of the occurrence of spikes up to 100 ms with the increased predictive horizon of the LSTM network from $N_p = 1$ to $N_p = 200$, but the performance degraded in capturing the magnitude of the membrane potentials during spiking with an increased value of N_p. Although the time-averaged root mean squared error of the membrane potential traces between the Hodgkin-Huxley model and the LSTM network for $N_p = 1, 50, 100, 200$ showed an improved performance with the increased value of N_p (see Figure 10(a)), none of the LSTM networks showed a reasonable prediction of the timing of the occurrence of spikes in this regime beyond 100 ms of the time-horizon.

To systematically evaluate whether the designed LSTM networks provide reasonable predictions of the membrane potential traces of the regular spiking dynamics across the range of external input currents between 0.79 nA and 2.3 nA, we performed simulations for 50 random stimulating currents drawn from a Uniform distribution $\mathcal{U}(0.79, 2.3)$. For each stimulating current, we chose 100 initial conditions drawn randomly from the maximum and minimum range of the Hodgkin-Huxley state variables (Note that the network was not trained over this wide range of initial conditions). Figure 10(b) shows the LSTM network performance, represented in terms of the root mean squared error vs time over 5000 realizations, for $N_p = 1, 50, 100, 200$. As shown in this figure, the root mean squared error decreased with the increased predictive horizon of the LSTM network for all time, which is consistent with the result shown in Figure 10(a).

In conclusion, these results suggest that our deep LSTM neural network with a longer predictive horizon feature can predict the irregular bursting patterns exhibited by hippocampal CA1 pyramidal neurons with high accuracy over only a short-time horizon.

3.3. Regular Bursting

In this section, we demonstrate the efficacy of our trained deep LSTM neural network over the range of external current between 0.24 nA and 0.79 nA in predicting the regular bursting dynamics shown by the biophysiological Hodgkin-Huxley model of CA1 pyramidal neuron in response to the external current $I \in [0.24, 0.79)$ nA. In this range, we observe firing rates between approximately 8 Hz and
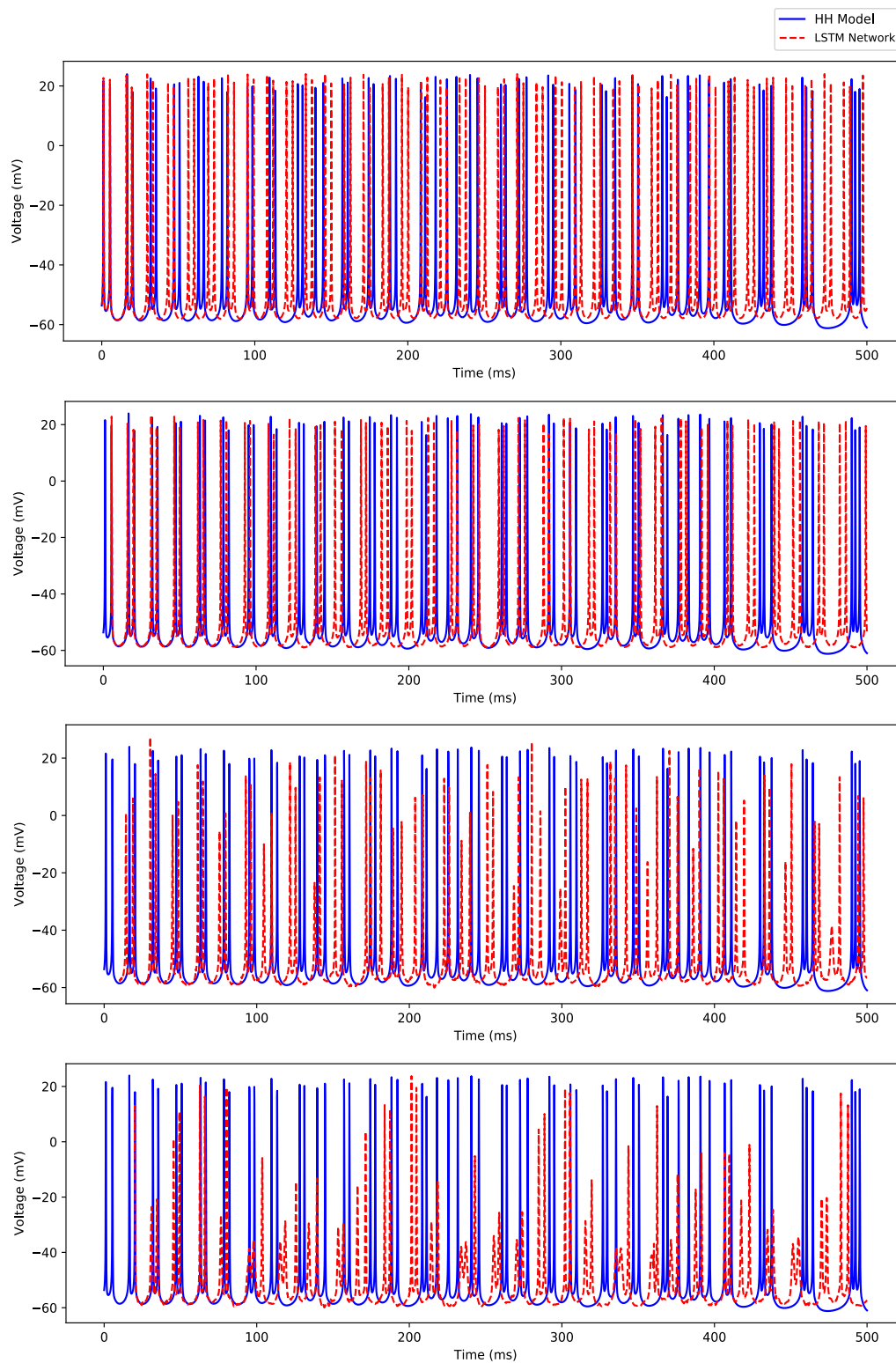
Figure 9. Comparison of predicted membrane potential traces by the deep LSTM neural network ("LSTM Network") to the irregular bursting spiking patterns exhibited by the Hodgkin-Huxley model ("HH Model") in response to the external stimulating current $I = 1.5$ nA. (a) Prediction using 1 timestep predictive LSTM network ($N_p = 1$). (b) Prediction using 50 timesteps predictive LSTM network ($N_p = 50$). (c) Prediction using 100 timesteps predictive LSTM network ($N_p = 100$). (d) Prediction using 200 timesteps predictive LSTM network ($N_p = 200$).
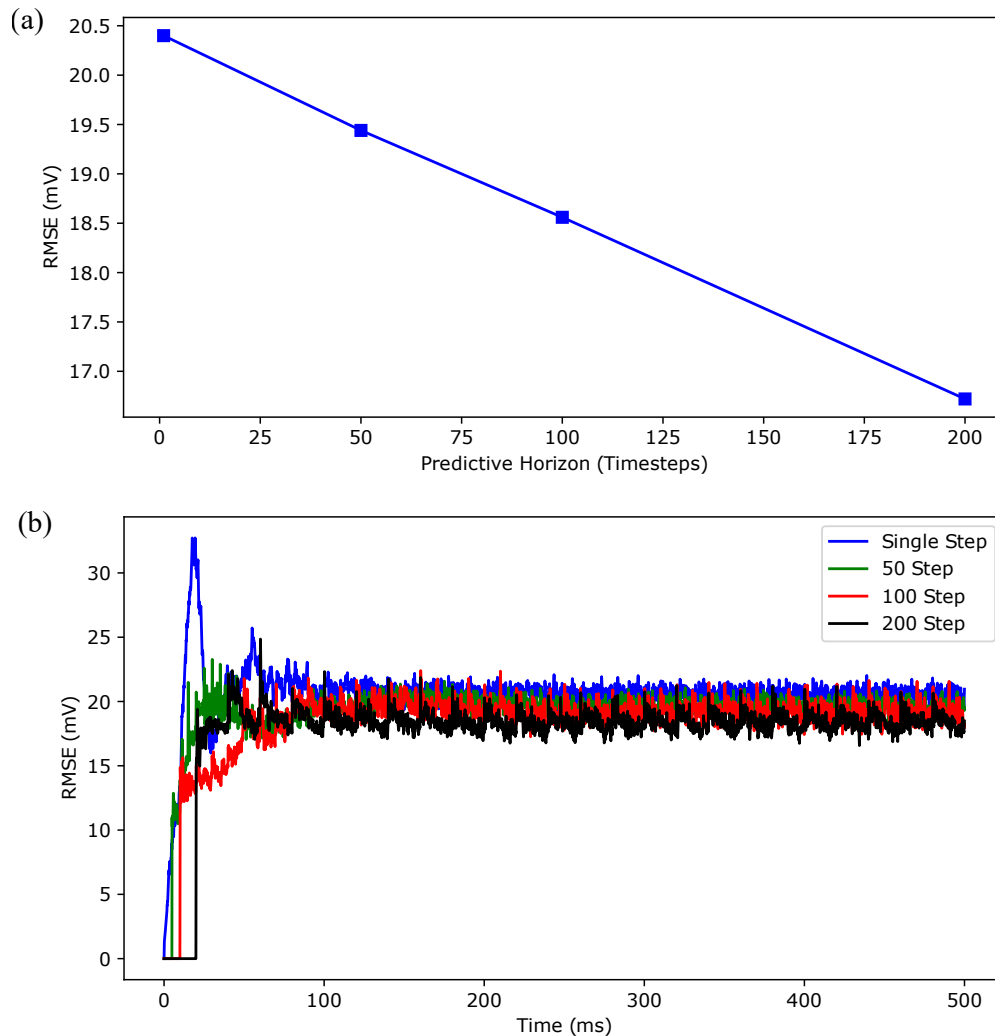
Figure 10. The effect of the prediction horizon of the deep LSTM neural network on the accuracy of irregular bursting dynamics prediction. (a) shows the time-averaged root mean squared error (RMSE) versus predictive horizon of the LSTM network ($N_p = 1, 50, 100, 200$) for the simulation results shown in Figure 9. (b) shows the RMSE versus simulation time for 5000 independent realizations, drawn from the predicted membrane potential trajectories of 50 randomly selected stimulating currents from a Uniform distribution $\mathcal{U}(0.79, 2.3)$ and 100 random initial conditions for each stimulating current.

52 Hz. For clarity, we here show our results only for the membrane potential traces. We provide the complete set of simulation results on the LSTM network performance in predicting the dynamics of all the 9 states of the Hodgkin-Huxley model in Appendix B.3 (see Figures A11, A12, A13, A14, and A15).

Figure 11 shows a comparison of the membrane potential traces simulated using the Hodgkin-Huxley model and the 4 different predictive horizons of the LSTM network (i.e., $N_p = 1, 50, 100, 200$) for the external stimulating current $I = 0.5$ nA. Note that all the simulations are performed using the initial condition used for $I = 3.0$ nA in Figure 7. Since our LSTM network uses the initial sequence of outputs of appropriate prediction horizon (i.e., $N_p = 1, 50, 100, 200$) from the Hodgkin-Huxley model to make future time predictions, the LSTM network predictions (shown by dashed red line) start after 0.1 ms, 5 ms, 10 ms, and 20 ms in Figure 11a, Figure 11b, Figure 11c, and Figure 11d respectively.

By analyzing the results shown in Figure 11, we found that the LSTM network performance in predicting the timing of spikes during bursts as well as tracking the subthreshold potential improved significantly from $N_p = 1$ to $N_p = 200$, but the performance substantially degraded in capturing the magnitude of the membrane potentials during spiking. In conclusion, the 200 timesteps prediction
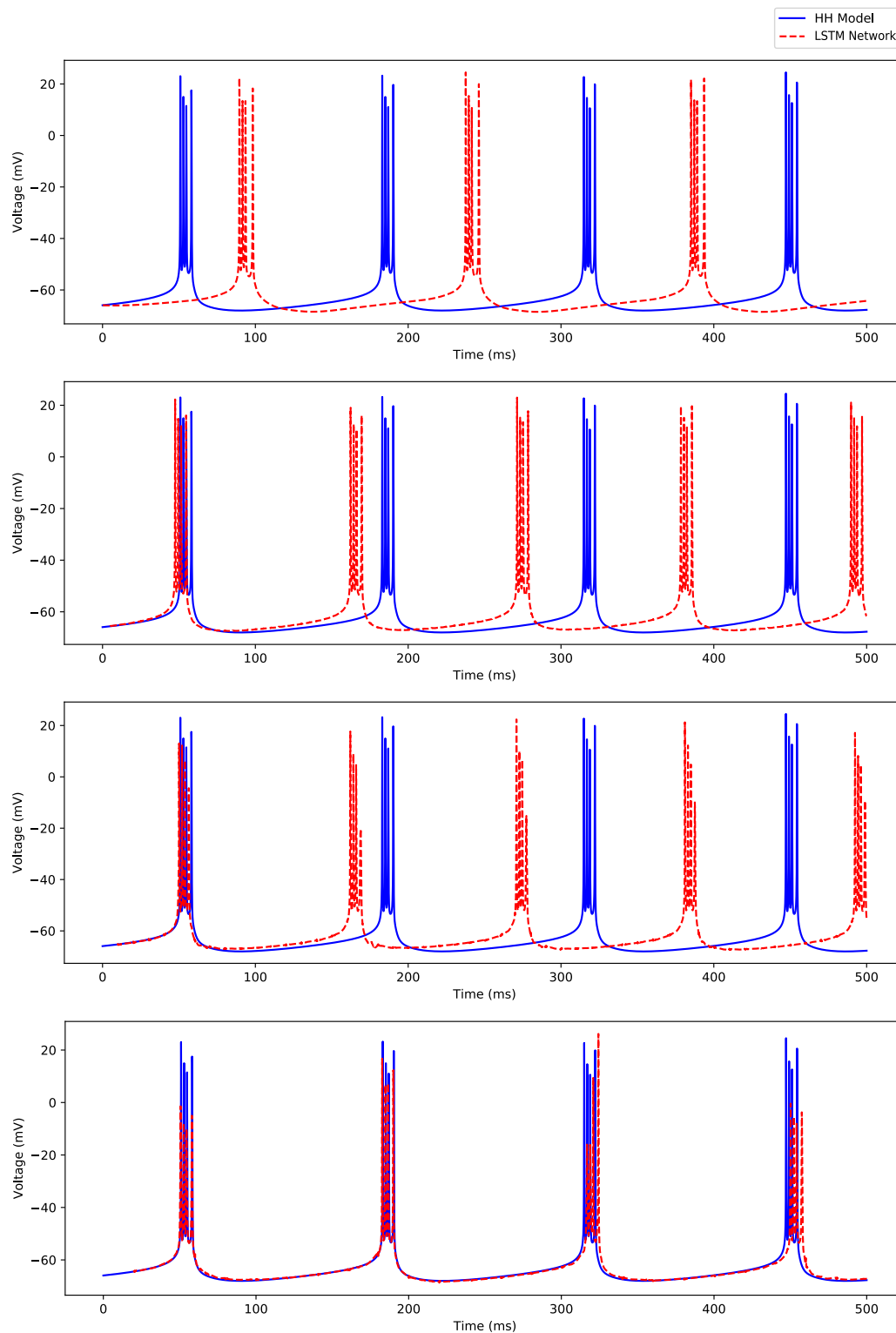
Figure 11. Comparison of predicted membrane potential traces by the LSTM network ("NN Prediction") to the irregular bursting spiking patterns exhibited by the Hodgkin-Huxley model ("HH Model") in response to the external stimulating current $I = 0.5$ nA. (a) Prediction using 1 timestep predictive LSTM network ($N_p = 1$). (b) Prediction using 50 timesteps predictive LSTM network ($N_p = 50$). (c) Prediction using 100 timesteps predictive LSTM network ($N_p = 100$). (d) Prediction using 200 timesteps predictive LSTM network ($N_p = 200$).

horizon based LSTM network (see Figure 11(d)) predicts the temporal dynamics with reasonable accuracy over the first 300 ms of prediction.

Figure 12(a) shows the time-averaged root mean squared error of the membrane potential traces between the Hodgkin-Huxley model and the LSTM network for $N_p = 1, 50, 100, 200$. As noted in this figure, the root mean squared error decreased substantially between 100 timesteps and 200 timesteps prediction horizon compared to the regimes of regular spiking (Figure 8(a)) and irregular bursting (Figure 10(a)), which indicates that a longer predictive horizon based LSTM network is necessary to capture the two different timescales (i.e., short intraburst spiking intervals and long interburst subthreshold intervals) presented in these dynamics.
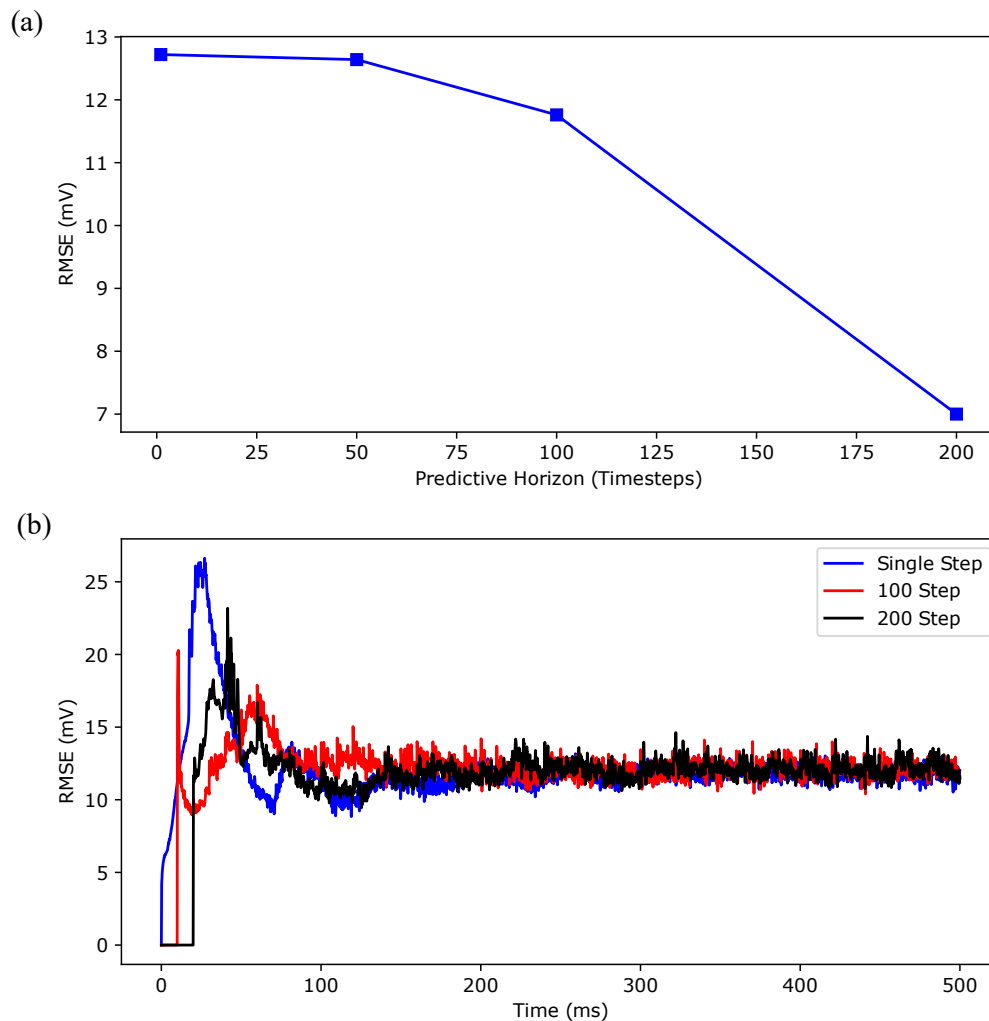
Figure 12. The effect of the prediction horizon of the multi-timestep LSTM network on the accuracy of regular bursting dynamics prediction. (a) shows the time-averaged root mean squared error (RMSE) versus predictive horizon of the LSTM network ($N_p = 1, 50, 100, 200$) for the simulation results shown in Figure 11. (b) shows the RMSE versus simulation time for 5000 independent realizations, drawn from the predicted membrane potential trajectories of 50 randomly selected stimulating currents from a Uniform distribution $\mathcal{U}(0.24, 0.79)$ and 100 random initial conditions for each stimulating current.

Figure 12(b) shows the LSTM networks performances, represented in terms of the root mean squared error vs time over 5000 realizations, for $N_p = 1, 100,$ and 200 timestep prediction horizon LSTM network. As shown in this figure, the root mean squared error decreased with the increased predictive horizon of the LSTM network for all time, which is consistent with the result shown in Figure 12(a). Note that we have excluded the simulation result for $N_p = 50$ as we found out in our detailed analysis that the trained LSTM network for $N_p = 50$ led to instability in predicting spiking

responses for some of the initial condition values in this regime. The reason for this may be that the network may not have seen these initial conditions during the training.

4. Discussion

In this paper, we developed and presented a new data-driven long short-term memory (LSTM) based neural network (NN) architecture to predict the dynamical spiking patterns of single neurons. Compared to other LSTM-based NN architectures for forecasting dynamical systems behavior reported in the literature, our architecture incorporated a single dense feedforward output layer with an activation function and a reverse-order sequence-to-sequence mapping approach into traditional LSTM based neural networks to enable truly multi-timestep stable predictions of the dynamics over a long time-horizon. We demonstrated the efficacy of our architecture in predicting the multi-time scale dynamics of hippocampal CA1 pyramidal neurons and compared the predictions from our model with the ground truth synthetic data obtained from an experimentally validated biophysiological model of CA1 pyramidal neuron in the Hodgkin-Huxley formalism. Through simulations, we showed that (1) the presented architecture can learn multi-timescale dynamics, and (2) the predictive accuracy of the network increases with the increase in the predictive horizon of the LSTM network.

Our results for irregular bursting regime showed the limitation of the designed deep LSTM neural network architecture in making an accurate prediction of the timing of the occurrence of spikes over a long-time horizon compared to regular spiking and regular bursting regimes. A possible reason for this may be the architecture itself or the dataset used for training these networks, which requires further investigations by training the networks on the dataset explicitly generated from this regime. In addition, it has been shown that this regime of bursting exhibits chaotic dynamics [46], which may provide further explanation for the network's struggle to accurately predict this bursting behavior, as the system exhibits a high sensitivity to the initial conditions. Hybrid approaches that combine LSTM networks with mean stochastic models (MSM) have been explored for predictively modeling chaotic dynamical systems in [24]. However, this LSTM-MSM approach is limited to iterative single step prediction, and the application of this technique falls beyond the scope of this paper.

Another limitation of our presented approach in modeling neuronal dynamics as currently constructed is the inclusion of the full state vector in both training and predictive evaluation. In experiment, it may be infeasible to have the entire state vector of the neuron measured for any given time. This should provide a valuable direction for future research, as partially observed systems or neuronal spike train recordings are much more feasible to measure in vivo or vitro, and merit further consideration in combination with this approach.

In all dynamical regimes, our results showed a degraded performance of the deep LSTM neural network in predicting the amplitude of membrane potentials during the timing of the occurrence of spikes with the increased predictive horizon of the LSTM network. This issue may be related to the equally weighted norm-2 loss function used for training the networks. A further investigation is required by considering different loss functions, such as norm-1 or weighted norm-2, which we consider as our future work.

In addition, we make a note on the computational requirement of our presented approach. The computational cost of inference with an artificial neural network can effectively be boiled down to the number of multiplications and additions needed to complete a forward pass of information. The inference complexity for an LSTM is roughly $\mathcal{O}(d_i \cdot d \cdot h + d \cdot h^2)$, as described in [24], where d_i is the dimensionality of each input, d is the number of inputs, and h is the dimensionality of the hidden states. Using this, we estimated the inference complexity of our LTSM network, represented by $\mathcal{O}(I)$, for the prediction horizon of 1, 50, 100, and 200. Using a single Nvidia GTX 1080Ti, we have also calculated the average computation time required for each of the predictive horizons used to predict 500ms of state values from 1,000 examples. We report these values in Table 1.

Table 1. Comparison of computational requirement for the iterative approach presented in this paper.

N_P ($\Delta t = 0.1$ms)	Prediction Time(ms)	Iterations	On-Line Computation Time (s)	$\mathcal{O}(I)$
1	500	5000	8.896	5.3×10^9
50	500	100	3.778	1.6×10^8
100	500	50	3.679	1.1×10^8
200	500	25	3.565	8.0×10^7

Although the data-driven approach developed in this paper showed the ability of the designed LSTM-based neural network in learning multi-timescale dynamics, we note that the network struggles to accurately capture the dynamics of some state variables where the magnitude of the state variable is comparable to the numerical precision of our simulations. This can particularly be seen in A2, A3, A4, and A14, where the network is not able to reconstruct the dynamics of the state variable q_{sAHP} with a reasonable accuracy. One possible way to alleviate this issue may be to increase the tolerance of the numerical errors in simulations which may increase the overall computational cost during training.

In conclusion, our results showed that a longer predictive horizon-based LSTM network can provide a more accurate prediction of multi-time scale dynamics, but at the expense of extensive offline training cost.

Author Contributions: Conceptualization, B.P. and G.K.; Methodology, B.P. and G.K.; Formal Analysis, B.P. and G.K.; Investigation, B.P.; Resources, B.P. and G.K.; Writing – Original Draft Preparation, B.P.; Writing – Review & Editing, G.K.; Visualization, B.P. and G.K.; Supervision, G.K.; Project Administration, G.K.; Funding Acquisition, G.K.

Funding: We gratefully acknowledge the startup funding support from the University of Idaho College of Engineering and Department of Chemical & Materials Engineering.

Conflicts of Interest:

The authors declare no conflict of interest.

References

1. Salmelin, R.; Hari, R.; Lounasmaa, O.; Sams, M. Dynamics of brain activation during picture naming. *Nature* **1994**, *368*, 463.
2. Fox, M.D.; Snyder, A.Z.; Vincent, J.L.; Corbetta, M.; Van Essen, D.C.; Raichle, M.E. The human brain is intrinsically organized into dynamic, anticorrelated functional networks. *Proceedings of the National Academy of Sciences* **2005**, *102*, 9673–9678.
3. Kiebel, S.J.; Daunizeau, J.; Friston, K.J. A hierarchy of time-scales and the brain. *PLoS computational biology* **2008**, *4*, e1000209.
4. Gerstner, W.; Kistler, W.M.; Naud, R.; Paninski, L. *Neuronal dynamics: From single neurons to networks and models of cognition*; Cambridge University Press, 2014.
5. Siettos, C.; Starke, J. Multiscale modeling of brain dynamics: from single neurons and networks to mathematical tools. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine* **2016**, *8*, 438–458.
6. Breakspear, M. Dynamic models of large-scale brain activity. *Nature neuroscience* **2017**, *20*, 340.
7. Herz, A.V.; Gollisch, T.; Machens, C.K.; Jaeger, D. Modeling single-neuron dynamics and computations: a balance of detail and abstraction. *science* **2006**, *314*, 80–85.
8. Gerstner, W.; Naud, R. How good are neuron models? *Science* **2009**, *326*, 379–380.
9. Chen, S.; Billings, S. Neural networks for nonlinear dynamic system modelling and identification. *International journal of control* **1992**, *56*, 319–346.
10. Purwar, S.; Kar, I.; Jha, A. Nonlinear system identification using neural networks. *IETE journal of research* **2007**, *53*, 35–42.
11. Kuschewski, J.G.; Hui, S.; Zak, S.H. Application of feedforward neural networks to dynamical system identification and control. *IEEE Transactions on Control Systems Technology* **1993**, *1*, 37–49.
12. Pan, S.; Duraisamy, K. Long-time predictive modeling of nonlinear dynamical systems using neural networks. *Complexity* **2018**, *2018*.

13. Gupta, P.; Sinha, N.K. Modeling robot dynamics using dynamic neural networks. *IFAC Proceedings Volumes* **1997**, *30*, 755–759.

14. Patra, J.C.; Pal, R.N.; Chatterji, B.; Panda, G. Identification of nonlinear dynamic systems using functional link artificial neural networks. *IEEE transactions on systems, man, and cybernetics, part b (cybernetics)* **1999**, *29*, 254–262.

15. Nagabandi, A.; Kahn, G.; Fearing, R.S.; Levine, S. Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning. 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2018, pp. 7559–7566.

16. Jaeger, H.; Haas, H. Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *science* **2004**, *304*, 78–80.

17. Bailer-Jones, C.A.; MacKay, D.J.; Withers, P.J. A recurrent neural network for modelling dynamical systems. *network: computation in neural systems* **1998**, *9*, 531–547.

18. Lenz, I.; Knepper, R.A.; Saxena, A. DeepMPC: Learning deep latent features for model predictive control. Robotics: Science and Systems. Rome, Italy, 2015.

19. Pascanu, R.; Mikolov, T.; Bengio, Y. On the difficulty of training recurrent neural networks. International conference on machine learning, 2013, pp. 1310–1318.

20. Mohajerin, N.; Waslander, S.L. Multistep Prediction of Dynamic Systems With Recurrent Neural Networks. *IEEE transactions on neural networks and learning systems* **2019**.

21. Lin, L.; Gong, S.; Li, T.; Peeta, S. Deep learning-based human-driven vehicle trajectory prediction and its application for platoon control of connected and autonomous vehicles. The Autonomous Vehicles Symposium, 2018, Vol. 2018.

22. Gonzalez, J.; Yu, W. Non-linear system modeling using LSTM neural networks. *IFAC-PapersOnLine* **2018**, *51*, 485–489.

23. Wang, Y. A new concept using LSTM neural networks for dynamic system identification. 2017 American Control Conference (ACC). IEEE, 2017, pp. 5324–5329.

24. Vlachas, P.R.; Byeon, W.; Wan, Z.Y.; Sapsis, T.P.; Koumoutsakos, P. Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* **2018**, *474*, 20170844.

25. Zenke, F.; Ganguli, S. Superspike: Supervised learning in multilayer spiking neural networks. *Neural computation* **2018**, *30*, 1514–1541.

26. Huh, D.; Sejnowski, T.J. Gradient descent for spiking neural networks. Advances in Neural Information Processing Systems, 2018, pp. 1433–1443.

27. Pandarinath, C.; O'Shea, D.J.; Collins, J.; Jozefowicz, R.; Stavisky, S.D.; Kao, J.C.; Trautmann, E.M.; Kaufman, M.T.; Ryu, S.I.; Hochberg, L.R.; others. Inferring single-trial neural population dynamics using sequential auto-encoders. *Nature methods* **2018**, p. 1.

28. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to sequence learning with neural networks. Advances in neural information processing systems, 2014, pp. 3104–3112.

29. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural computation* **1997**, *9*, 1735–1780.

30. Glorot, X.; Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. Proceedings of the thirteenth international conference on artificial intelligence and statistics, 2010, pp. 249–256.

31. McKiernan, E.C.; Marrone, D.F. CA1 pyramidal cells have diverse biophysical properties, affected by development, experience, and aging. *PeerJ* **2017**, *5*, e3836.

32. Golomb, D.; Yue, C.; Yaari, Y. Contribution of persistent Na+ current and M-type K+ current to somatic bursting in CA1 pyramidal cells: combined experimental and modeling study. *Journal of neurophysiology* **2006**, *96*, 1912–1926.

33. Nowacki, J.; Osinga, H.M.; Brown, J.T.; Randall, A.D.; Tsaneva-Atanasova, K. A unified model of CA1/3 pyramidal cells: An investigation into excitability. *Progress in biophysics and molecular biology* **2011**, *105*, 34–48.

34. Ferguson, K.A.; Huh, C.Y.; Amilhon, B.; Williams, S.; Skinner, F.K. Simple, biologically-constrained CA1 pyramidal cell models using an intact, whole hippocampus context. *F1000Research* **2014**, *3*.

35. Poirazi, P.; Brannon, T.; Mel, B.W. Arithmetic of subthreshold synaptic summation in a model CA1 pyramidal cell. *Neuron* **2003**, *37*, 977–987.

36. Royeck, M.; Horstmann, M.T.; Remy, S.; Reitze, M.; Yaari, Y.; Beck, H. Role of axonal NaV1. 6 sodium channels in action potential initiation of CA1 pyramidal neurons. *Journal of neurophysiology* **2008**, *100*, 2361–2380.

37. Katz, Y.; Menon, V.; Nicholson, D.A.; Geinisman, Y.; Kath, W.L.; Spruston, N. Synapse distribution suggests a two-stage model of dendritic integration in CA1 pyramidal neurons. *Neuron* **2009**, *63*, 171–177.

38. Bianchi, D.; Marasco, A.; Limongiello, A.; Marchetti, C.; Marie, H.; Tirozzi, B.; Migliore, M. On the mechanisms underlying the depolarization block in the spiking dynamics of CA1 pyramidal neurons. *Journal of computational neuroscience* **2012**, *33*, 207–225.

39. Marasco, A.; Limongiello, A.; Migliore, M. Fast and accurate low-dimensional reduction of biophysically detailed neuron models. *Scientific reports* **2012**, *2*, 1–7.

40. Kim, Y.; Hsu, C.L.; Cembrowski, M.S.; Mensh, B.D.; Spruston, N. Dendritic sodium spikes are required for long-term potentiation at distal synapses on hippocampal pyramidal neurons. *Elife* **2015**, *4*, (e06414) 1–30.

41. Bezaire, M.J.; Raikov, I.; Burk, K.; Vyas, D.; Soltesz, I. Interneuronal mechanisms of hippocampal theta oscillations in a full-scale model of the rodent CA1 circuit. *Elife* **2016**, *5*, (e18566) 1–106.

42. Werbos, P.J. Generalization of backpropagation with application to a recurrent gas market model. *Neural networks* **1988**, *1*, 339–356.

43. Werbos, P.J.; others. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE* **1990**, *78*, 1550–1560.

44. Mozer, M.C. A focused backpropagation algorithm for temporal. *Backpropagation: Theory, architectures, and applications* **1995**, *137*.

45. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* **2014**.

46. Naud, R.; Marcille, N.; Clopath, C.; Gerstner, W.   Firing patterns in the adaptive exponential integrate-and-fire model. *Biological cybernetics* **2008**, *99*, 335.

Appendix A Hodgkin-Huxley Model of CA1 Pyramidal Neuron Dynamics

We used the following Hodgkin-Huxley model of CA1 pyramidal neuron from [32] to demonstrate the efficacy of our data-driven modeling approach presented in this paper:

$$C\frac{dV}{dt} = -g_L(V - V_L) - I_{Na} - I_{NaP} - I_{Kdr} - I_A - I_M - I_{Ca} - I_C - I_{sAHP} + I_{app}, \tag{A1}$$

where the ionic currents I_{Na}, I_{NaP}, I_{Kdr}, I_A, I_M, I_{sAHP}, I_C, and I_{Ca} are given by

$$I_{Na} = g_{Na}m_\infty^3(V)h_{Na}(V - V_{Na}), \tag{A2a}$$

$$I_{NaP} = g_{NaP}p_\infty(V)(V - V_{Na}), \tag{A2b}$$

$$I_{Kdr} = g_{Kdr}n_{Kdr}^4(V - V_K), \tag{A2c}$$

$$I_A = g_A a_\infty^3(V)b_{Kdr}(V - V_K), \tag{A2d}$$

$$I_M = g_M z_M(V - V_K), \tag{A2e}$$

$$I_{Ca} = g_{Ca}r_{Ca}^2(V - V_{Ca}), \tag{A2f}$$

$$I_C = g_C d_\infty([Ca^{2+}]_i)c_C(V - V_K), \tag{A2g}$$

$$I_{sAHP} = g_{sAHP}q_{sAHP}(V - V_K). \tag{A2h}$$

Here, V is the membrane potential in mV, C is the membrane capacitance, V_L is the reversal potential of the leak current, g_L is the conductance of the leak current, and I_{app} is the externally applied stimulating current. The ionic currents I_{Na}, I_{NaP}, I_{Kdr}, I_A, I_M, I_{sAHP}, I_C, and I_{Ca} represent the transient sodium current, persistent sodium current, delayed rectifier potassium current, A-type potassium current, muscarinic-sensitive potassium current, slow calcium-activated potassium current, fast calcium-activated potassium current, and high threshold calcium current respectively. g_i, $i \in$

$\{Na, NaP, Kdr, A, M, Ca, C, sAHP\}$ represents the conductance of the ion channel i. $V_i, i \in \{Na, K, Ca\}$ is the reversal potential of the ion channel i.

The dynamics of the transient activation/deactivation variables of the ionic and calcium currents, i.e., $h_{Na}, n_{Kdr}, b_{Kdr}, z_M, r_{Ca}, c_C, q_{sAHP}$, and $[Ca^{2+}]_i$, are given by:

$$\frac{dh_{Na}}{dt} = \phi \frac{h_\infty(V) - h_{Na}}{\tau_{h_{Na}}(V)}, \tag{A3a}$$

$$\frac{dn_{Kdr}}{dt} = \phi \frac{n_\infty(V) - n_{Kdr}}{\tau_{n_{Kdr}}(V)}, \tag{A3b}$$

$$\frac{db_{Kdr}}{dt} = \frac{b_\infty(V) - b_{Kdr}}{\tau_{b_{Kdr}}}, \tag{A3c}$$

$$\frac{dz_M}{dt} = \frac{z_\infty(V) - z_M}{\tau_z}, \tag{A3d}$$

$$\frac{dr_{Ca}}{dt} = \frac{r_\infty(V) - r_{Ca}}{\tau_{r_{Ca}}}, \tag{A3e}$$

$$\frac{dc_C}{dt} = \frac{c_\infty(V) - c_C}{\tau_{c_C}}, \tag{A3f}$$

$$\frac{dq_{sAHP}}{dt} = \frac{q_\infty(V) - q_{sAHP}}{\tau_{q_{sAHP}}}, \tag{A3g}$$

$$\frac{d[Ca^{2+}]_i}{dt} = -\nu I_{Ca} - \frac{[Ca^{2+}]_i}{\tau_{Ca}}. \tag{A3h}$$

Here, $m_\infty(V)$, $h_\infty(V)$, $n_\infty(V)$, $p_\infty(V)$, $a_\infty(V)$, $b_\infty(V)$, $z_\infty(V)$, $r_\infty(V)$, $c_\infty(V)$, $q_\infty([Ca^{2+}]_i)$, and $d_\infty([Ca^{2+}]_i)$ are the steady-state activation/deactivation functions. ϕ is a scaling parameter. $\tau_{h_{Na}}(V)$, $\tau_{n_{Kdr}}(V)$, $\tau_{b_{Kdr}}$, $\tau_{r_{Ca}}$, τ_{c_C}, and $\tau_{q_{sAHP}}$ are the time constants. The steady-state activation/deactivation functions are given by:

$$m_\infty(V) = \frac{1}{1 + e^{-(V - \theta_m)/\sigma_m}}, \tag{A4a}$$

$$n_\infty(V) = \frac{1}{1 + e^{-(V - \theta_n)/\sigma_n}}, \tag{A4b}$$

$$h_\infty(V) = \frac{1}{1 + e^{-(V - \theta_h)/\sigma_h}}, \tag{A4c}$$

$$p_\infty(V) = \frac{1}{1 + e^{-(V - \theta_p)/\sigma_p}}, \tag{A4d}$$

$$b_\infty(V) = \frac{1}{1 + e^{-(V - \theta_b)/\sigma_b}}, \tag{A4e}$$

$$z_\infty(V) = \frac{1}{1 + e^{-(V - \theta_z)/\sigma_z}}, \tag{A4f}$$

$$a_\infty(V) = \frac{1}{1 + e^{-(V - \theta_a)/\sigma_a}}, \tag{A4g}$$

$$r_\infty(V) = \frac{1}{1 + e^{-(V - \theta_r)/\sigma_r}}, \tag{A4h}$$

$$c_\infty(V) = \frac{1}{1 + e^{-(V - \theta_c)/\sigma_c}}, \tag{A4i}$$

$$d_\infty([Ca^{2+}]_i) = \frac{1}{(1 + a_c/[Ca^{2+}]_i)}, \tag{A4j}$$

$$q_\infty([Ca^{2+}]_i) = \frac{1}{1 + (a_q^4/[Ca^{2+}]_i^4)}. \tag{A4k}$$

Here, $a_c, a_q, \theta_i, \sigma_i$ for $i \in \{m, n, h, p, b, z, a, r, c\}$ are the model parameters. The voltage dependent time constants $\tau_{h_{Na}}(V)$ and $\tau_{n_{Kdr}}(V)$ are given by

$$\tau_{h_{Na}}(V) = 1 + \frac{7.5}{1 + e^{-(V-\theta_{ht})/\sigma_{ht}}}, \tag{A5a}$$

$$\tau_{n_{Kdr}}(V) = 1 + \frac{5}{1 + e^{-(V-\theta_{nt})/\sigma_{nt}}}, \tag{A5b}$$

s

where $\theta_{ht}, \theta_{nt}, \sigma_{ht}$, and σ_{nt} are model parameters.

Throughout this paper, we used the following numerical values for the unknown model parameters [32]: $C = 1\mu F/cm^2$, $g_L = 0.05$ mS/cm^2, $V_L = -70$ mV, $\nu = 0.13$ cm^2/(ms $\times$ μA), $g_{Na} = 35$ mS/cm^2, $V_{Na} = 55$ mV, $g_{NaP} = 0.4$ mS/cm^2, $g_{Kdr} = 6.0$ mS/cm^2, $V_K = $ -90 mV, $g_A = 1.4$ mS/cm^2, $g_M = 0.5$ mS/cm^2, $g_{Ca} = 0.08$ mS/cm^2, $g_C = 10$ mS/cm^2, $V_{Ca} = 120$ mV, and $g_{sAHP} = 5$ mS/cm^2, $\theta_m = $ -30 mV, $\sigma_m = 9.5$ mV, $\theta_h = $ -45 mV, $\sigma_h = $ -7 mV, $\theta_{ht} = $ -40.5 mV, $\sigma_{ht} = $ -6 mV, $\phi = 10$, $\theta_P = $ -47 mV, $\sigma_P = 3$ mV, $\theta_n = $ -35 mV, $\sigma_n = 10$ mV, $\theta_{nt} = $ -27 mV, $\sigma_{nt} = $ -15 mV, $\theta_a = $ -50 mV, $\sigma_a = 20$ mV, $\theta_b = $ -80 mV, $\sigma_b = $ -6 mV, $\theta_z = $ -39 mV, $\sigma_z = 5$ mV, $\theta_r = $ -20 mV, $\sigma_r = 10$ mV, $\tau_r = 1$ ms, $\theta_c = $ -30 mV, $\sigma_c = 7$ mV, $\theta_c = 2$ ms, $a_c = 6$, $\tau_q = 450$ ms, and $a_q = 2$.

Unless otherwise stated, we used the following initial conditions to simulate the Hodgkin-Huxley model for generating the synthetic data: $V_0 = $ -71.81327mV, $h_{Na0} = 0.98786$, $n_{Kdr0} = 0.02457$, $b_{KA0} = 0.203517$, $u_{KM0} = 0.00141$, $r_{Ca0} = 0.005507$, $[Ca]_{i0} = 0.000787$, $c_{C0} = 0.002486$, $q_{Ca0} = 0.0$.

Appendix B Simulation Results on Full State Predictions of Hodgkin-Huxley Model

In Section 3, we showed our simulation results only for the membrane potential traces. Here, we provide the simulation results for all the 9 states of the Hodgkin-Huxley model of CA1 pyramidal neuron (HHCA1Py) predicted by the deep LSTM neural network over a long-time horizon and show the comparison between these predictions and the simulated dynamics from HHCA1Py.

Appendix B.1 Regular Spiking

In this section, we show the simulation results on predicting the dynamics of all the 9 states of HHCA1Py over a long-time horizon using the deep LSTM neural network for the regular periodic spiking regime ($I \in [2.3, 3.0]$ nA). Figures A1, A2, A3, and A4 show the comparison between the state's dynamics simulated using the Hodgkin-Huxley model and the deep LSTM neural network model developed for 1 timestep, 50 timesteps, 100 timesteps, and 200 timesteps (equivalently, $N_p = 1, 50, 100, 200$) predictive horizon, respectively.

As shown in these figures, the performance of the deep LSTM neural network model in predicting state dynamics significantly improved with the increased predictive horizon of the LSTM network (i.e., $N_p = 1$ to $N_p = 200$) for all the states except q_{sAHP} for which we found that the magnitude is comparable to the numerical precision of the performed simulations. Figure A5 shows the root mean squared error between the states of HHCA1Py and the deep LSTM neural network model as a function of simulation time over 5000 random realizations, for $N_p = 1, 50, 100, 200$. These results show that the root mean squared error decreases from $N_p = 1$ to $N_p = 200$.

Appendix B.2 Irregular Bursting

In this section, we show the simulation results on predicting the dynamics of all the 9 states of HHCA1Py over a long-time horizon using the deep LSTM neural network for the irregular bursting regime ($I \in [0.79, 2.3]$ nA). Figures A6, A7, A8, and A9 show the comparison between the state's dynamics simulated using the Hodgkin-Huxley model and the deep LSTM neural network
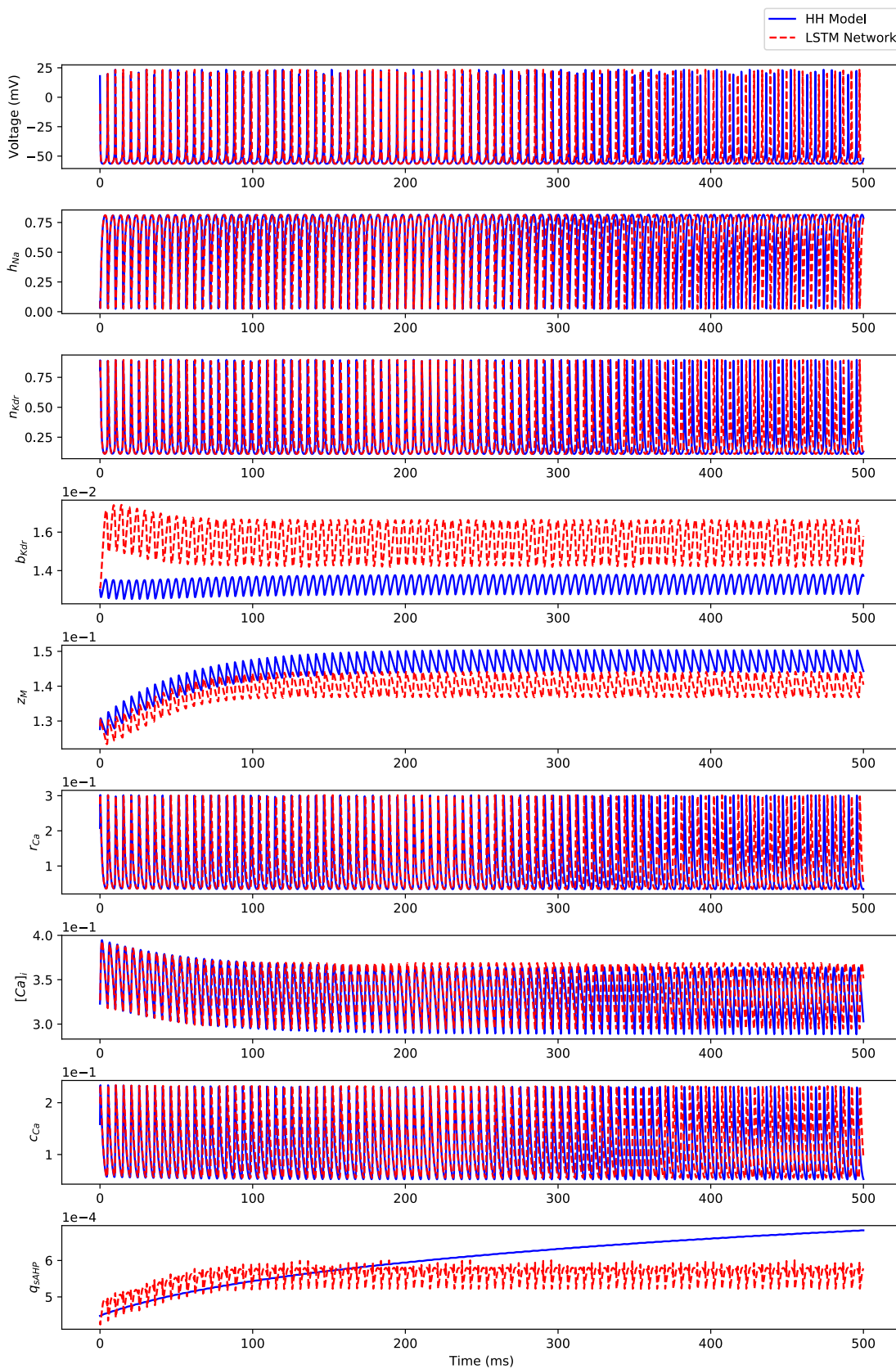
Figure A1. Comparison between the Hodgkin-Huxley model ("HH Model") states' dynamics and the iterative predictions of states' dynamics using the 1 timestep predictive horizon-based deep LSTM neural network ("LSTM Network") in response to $I = 3.0$ nA.
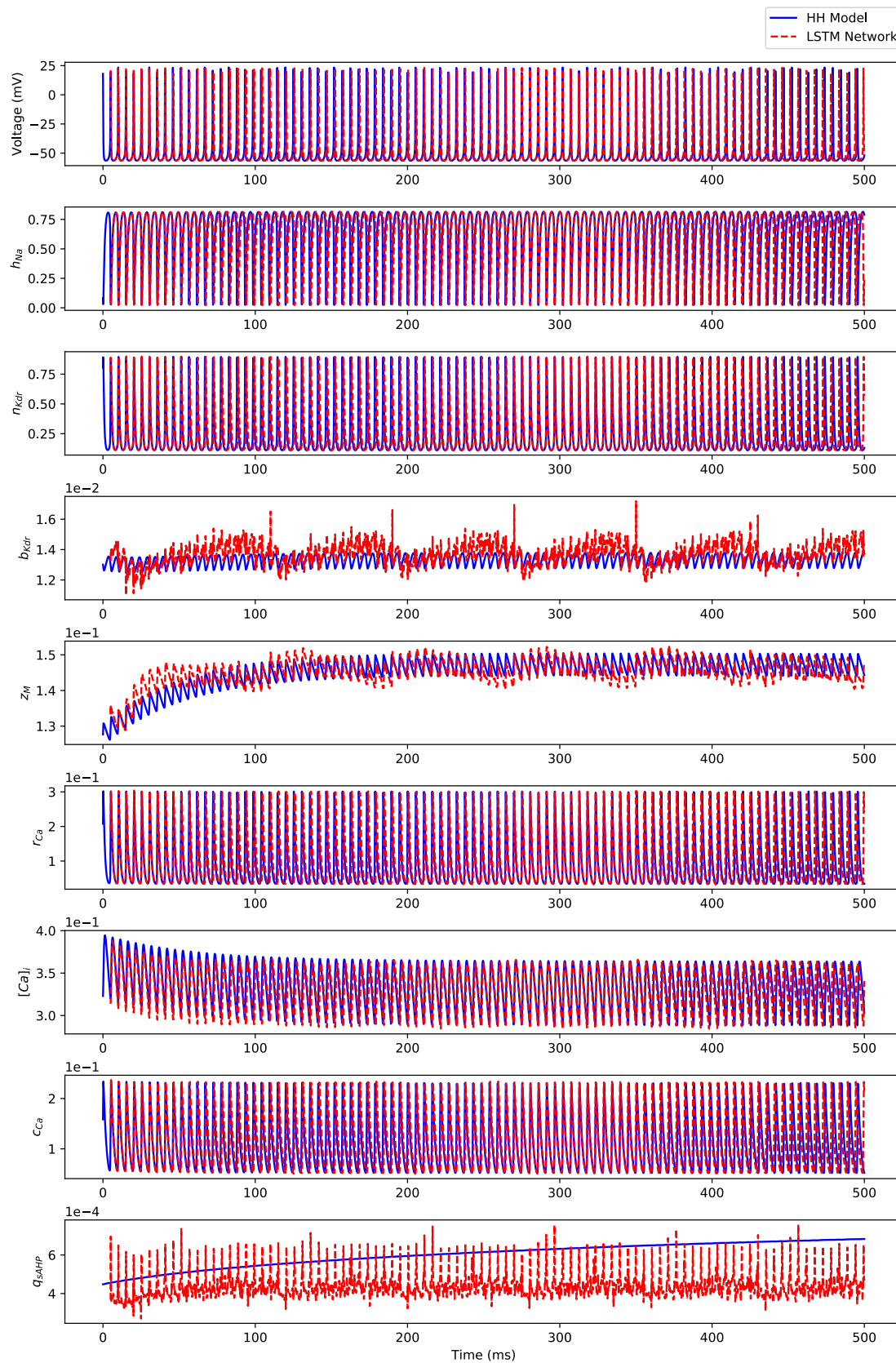
Figure A2. Comparison between the Hodgkin-Huxley model ("HH Model") states' dynamics and the iterative predictions of states' dynamics using the 50 timesteps predictive horizon-based deep LSTM neural network ("LSTM Network") in response to $I = 3.0$ nA.
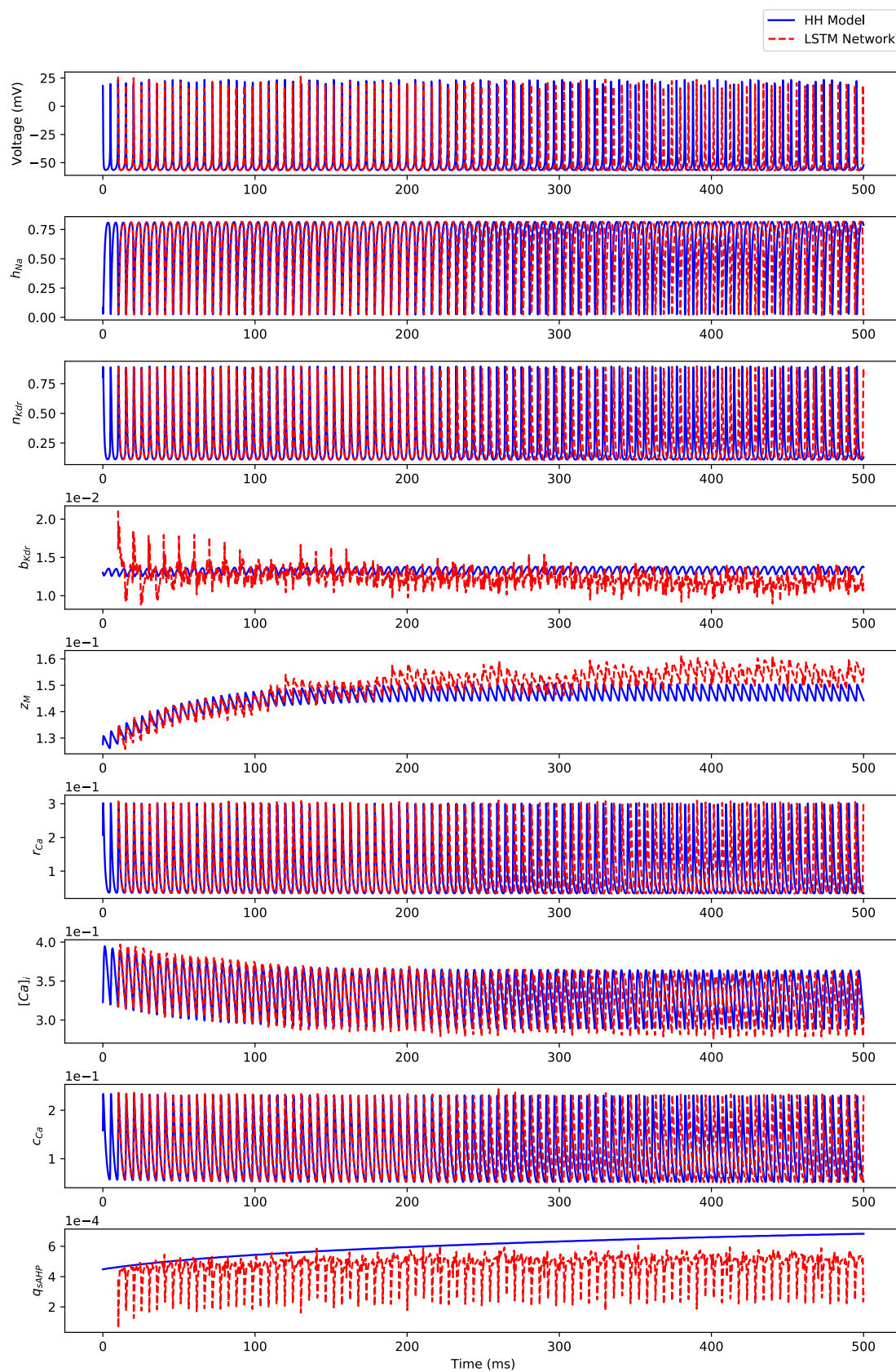
Figure A3. Comparison between the Hodgkin-Huxley model ("HH Model") states' dynamics and the iterative predictions of states' dynamics using the 100 timesteps predictive horizon-based deep LSTM neural network ("LSTM Network") in response to $I = 3.0$ nA.
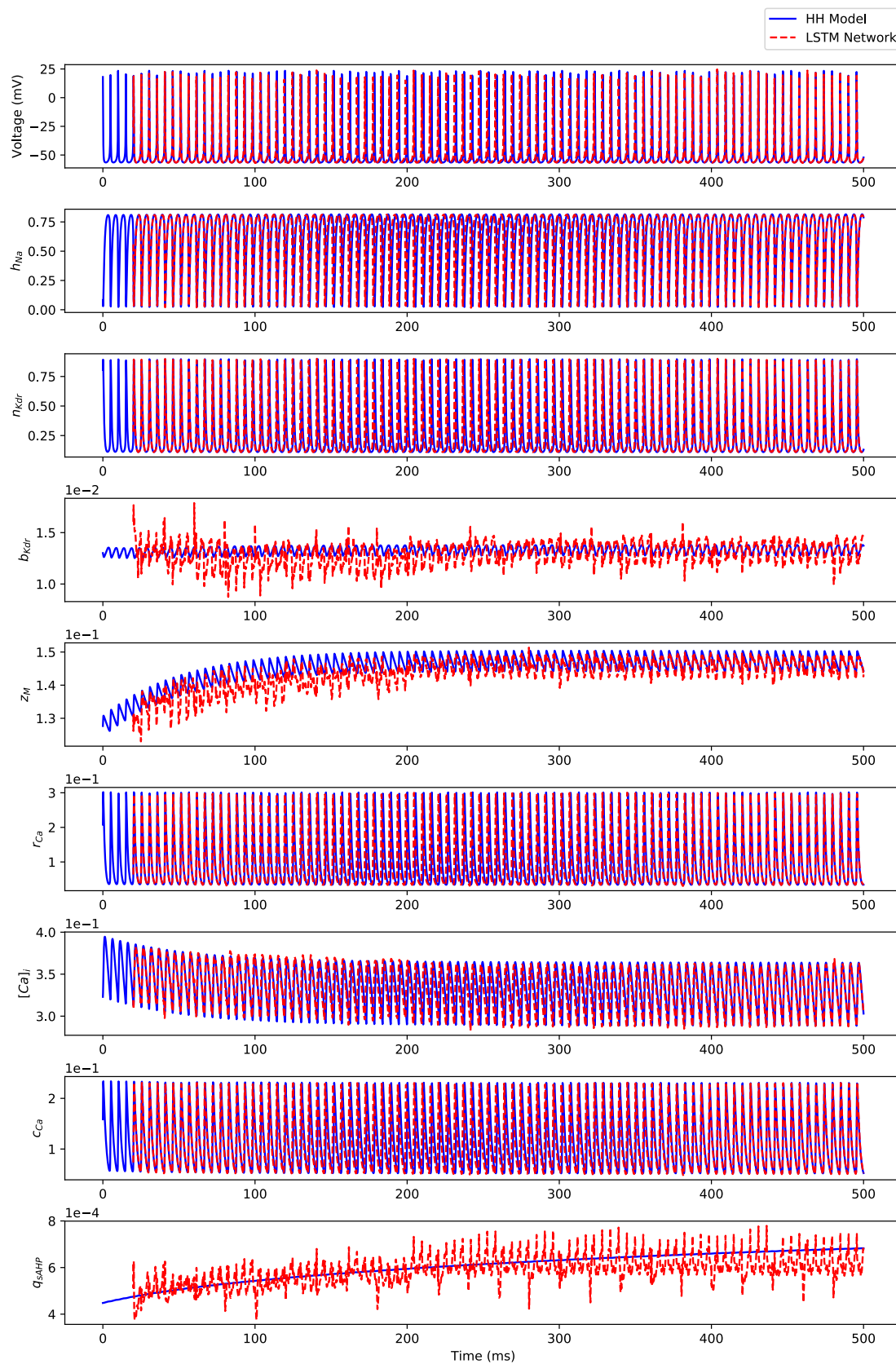
Figure A4. Comparison between the Hodgkin-Huxley model ("HH Model") states' dynamics and the iterative predictions of states' dynamics using the 200 timesteps predictive horizon-based deep LSTM neural network ("LSTM Network") in response to $I = 3.0$ nA.
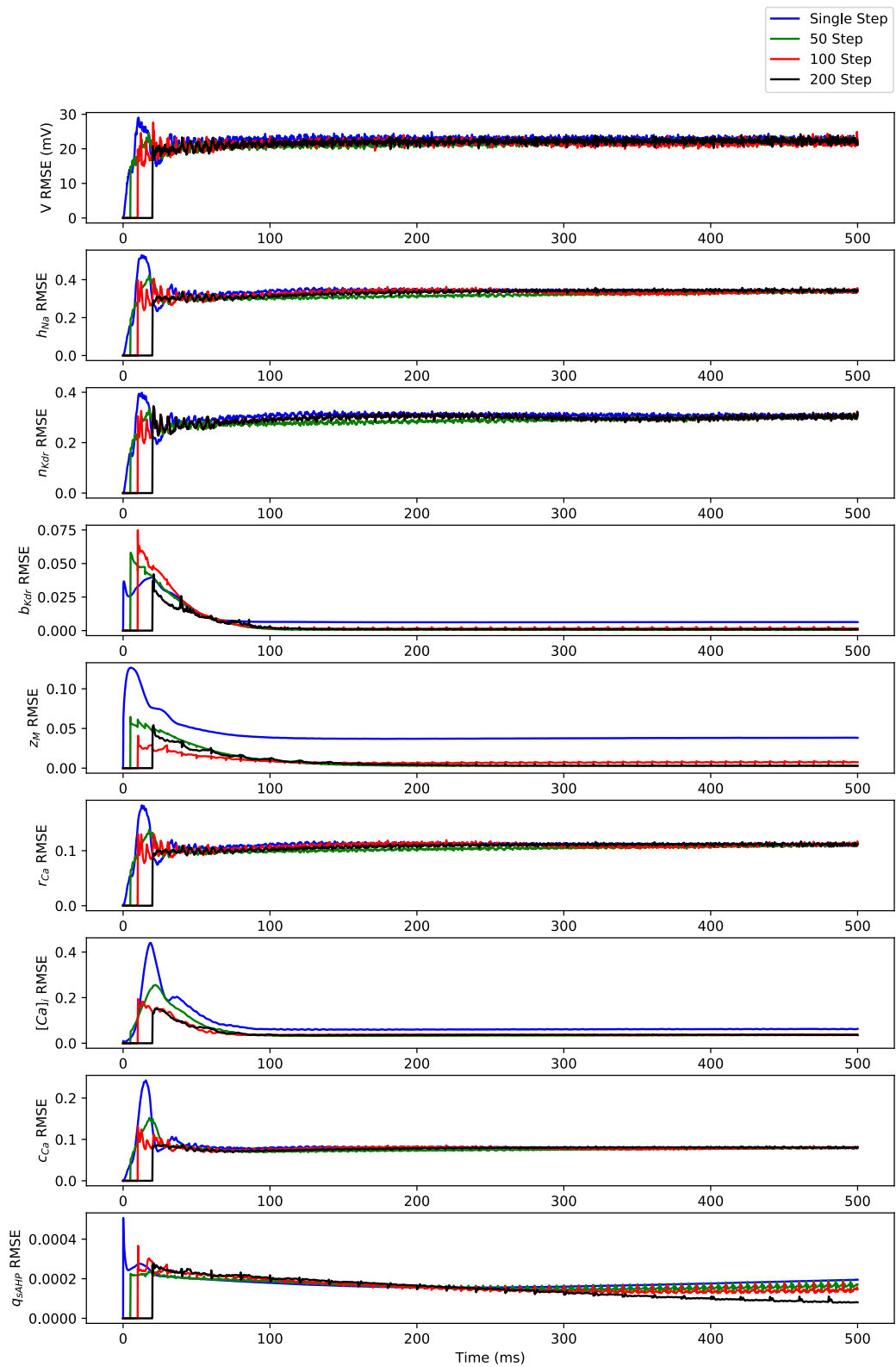
Figure A5. The root mean squared error (RMSE) versus simulation time for 5000 independent realizations, drawn from the predicted membrane potential trajectories of 50 randomly selected stimulating currents from a Uniform distribution $\mathcal{U}(2.3, 3.0)$ and 100 random initial conditions for each stimulating current.

model developed for 1 timestep, 50 timesteps, 100 timesteps, and 200 timesteps (equivalently, $N_p = 1, 50, 100, 200$) predictive horizon, respectively.

As shown in these figures, the deep LSTM neural network model provides a reasonable prediction of the dynamics of all the states except q_{sAHP} over the initial 100 ms of simulations. Moreover, the prediction improved from $N_p = 1$ to $N_p = 200$, which is consistent with the results for the regular spiking regime (see Figures A1, A2, A3, A4, and A5). We found that the magnitude of q_{sAHP} was comparable to the numerical precision of our simulations, which hindered the capability of the LSTM network in making a reasonable prediction for this state.

Figure A10 shows the root mean squared error between the states of HHCA1Py and the deep LSTM neural network model as a function of simulation time over 5000 random realizations, for $N_p = 1, 50, 100, 200$. As shown here, the root mean squared error decreased with the increased predictive horizon of the LSTM network (i.e., $N_p = 1$ to $N_p = 200$).

Appendix B.3 Regular Bursting

In this section, we show the simulation results on predicting the dynamics of all the 9 states of HHCA1Py over a long-time horizon using the deep LSTM neural network for the regular bursting regime ($I \in [0.24, 0.79)$ nA). Figures A11, A12, A13, and A14 show the comparison between the state's dynamics simulated using the Hodgkin-Huxley model and the deep LSTM neural network model developed for 1 timestep, 50 timesteps, 100 timesteps, and 200 timesteps (equivalently, $N_p = 1, 50, 100, 200$) predictive horizon, respectively.

As shown in these figures, the performance of the deep LSTM neural network model in predicting state dynamics significantly improved between 1 timestep predictive horizon (Figure A11) and 200 timesteps predictive horizon (Figure A14) across all the states except q_{sAHP} for the similar reason we provided for the regular spiking and irregular bursting regimes. More importantly, the LTSM network predicted the temporal correlations with high accuracy over the time-horizon of 300 ms for $N_p = 200$. The extrapolation of these results suggest that increasing the predictive horizon beyond $N_p = 200$ could improve the prediction beyond 300 ms of time-horizon.

In Figure A10, we show the root mean squared error between the states of HHCA1Py and the deep LSTM neural network model as a function of simulation time over 5000 random realizations, for $N_p = 1, 50, 100, 200$. As shown here, the root mean squared error decreased with the increased predictive horizon of the LSTM network (i.e., $N_p = 1$ to $N_p = 200$), which is consistent with the results of the regular spiking and irregular bursting regimes.
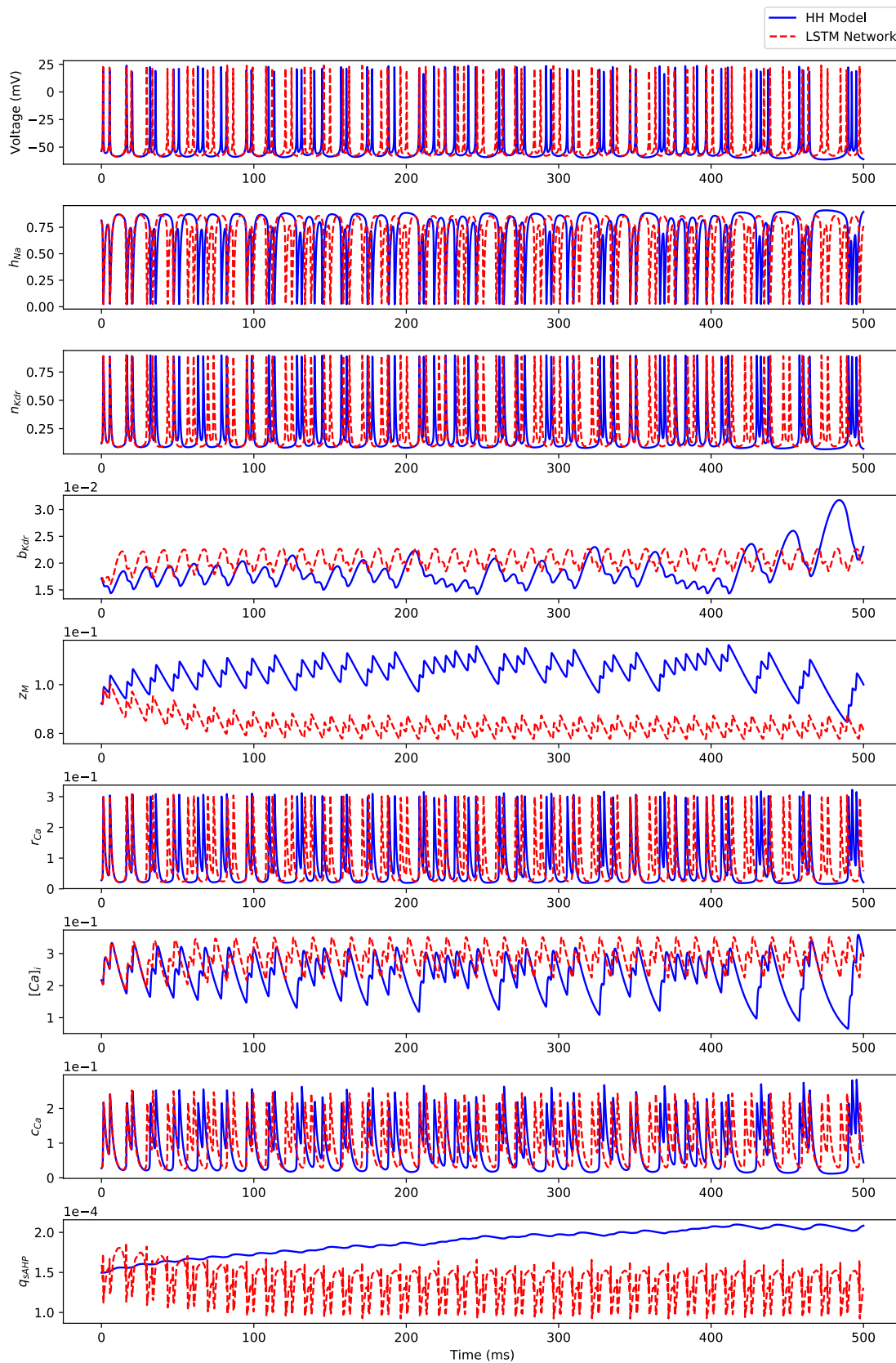
Figure A6. Comparison between the Hodgkin-Huxley model ("HH Model") states' dynamics and the iterative predictions of states' dynamics using the 1 timestep predictive horizon-based deep LSTM neural network ("LSTM Network") in response to $I = 1.5$ nA.
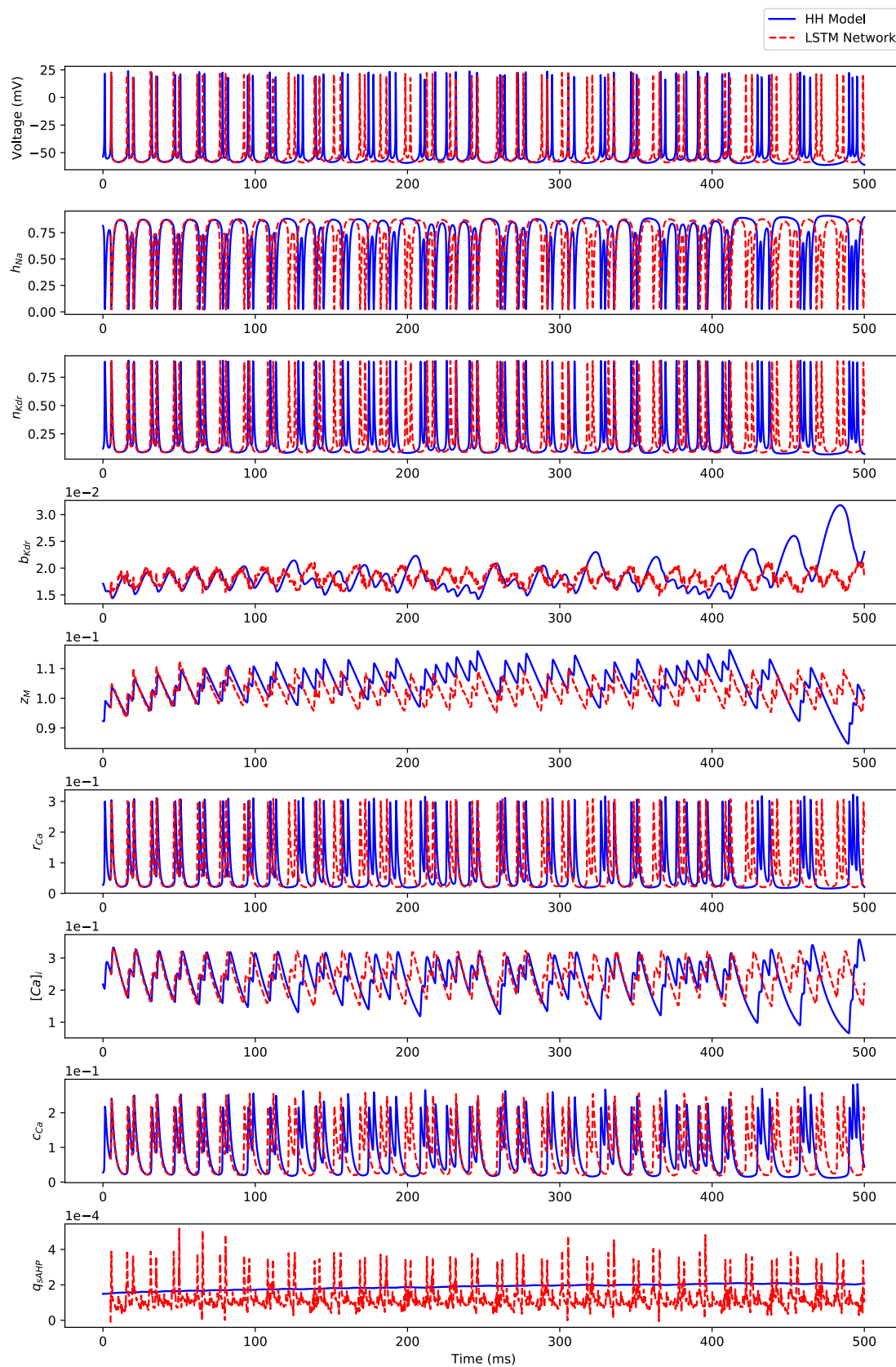
Figure A7. Comparison between the Hodgkin-Huxley model ("HH Model") states' dynamics and the iterative predictions of states' dynamics using the 50 timesteps predictive horizon-based deep LSTM neural network ("LSTM Network") in response to $I = 1.5$ nA.
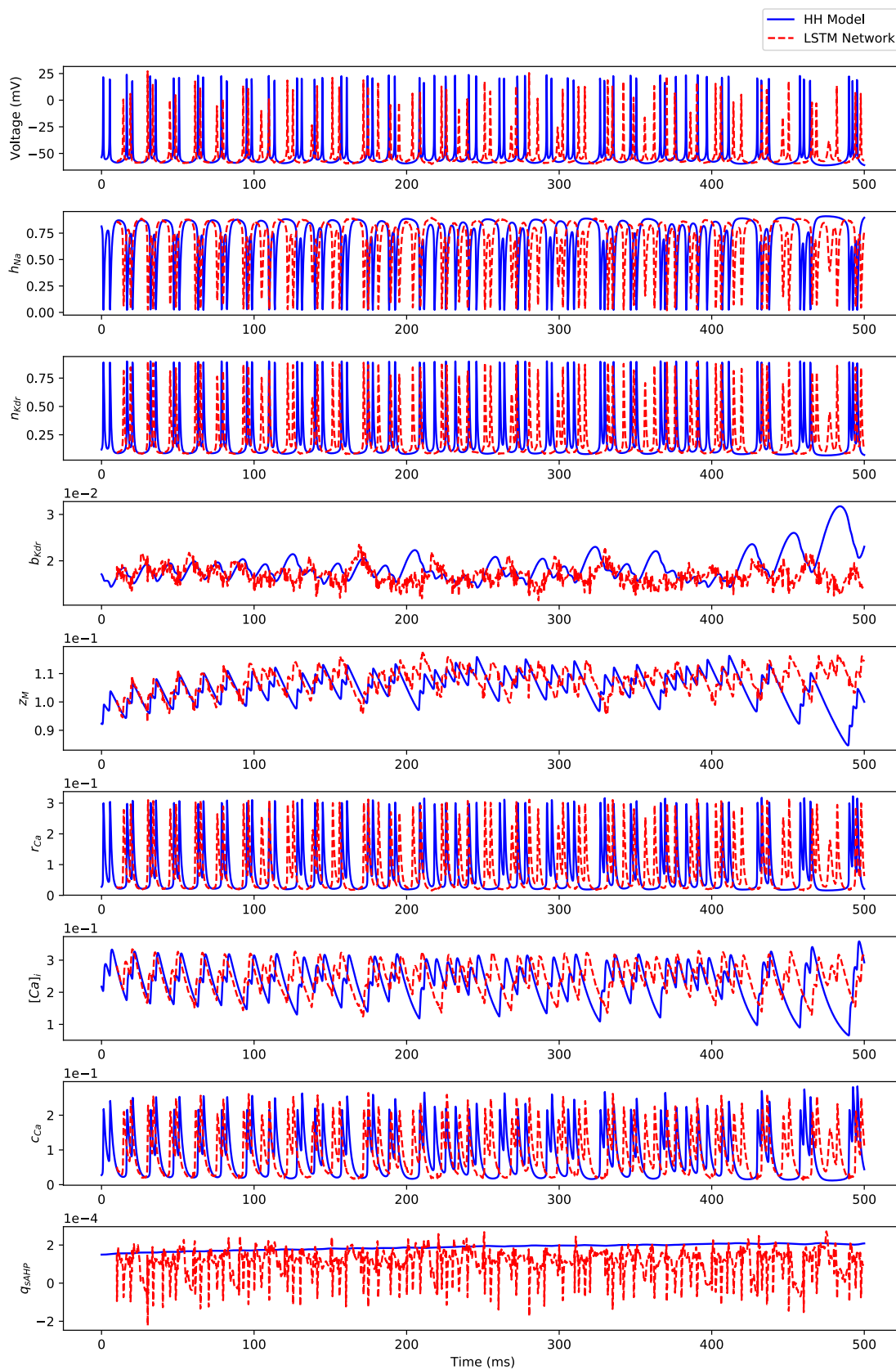
Figure A8. Comparison between the Hodgkin-Huxley model ("HH Model") states' dynamics and the iterative predictions of states' dynamics using the 100 timesteps predictive horizon-based deep LSTM neural network ("LSTM Network") in response to $I = 1.5$ nA.
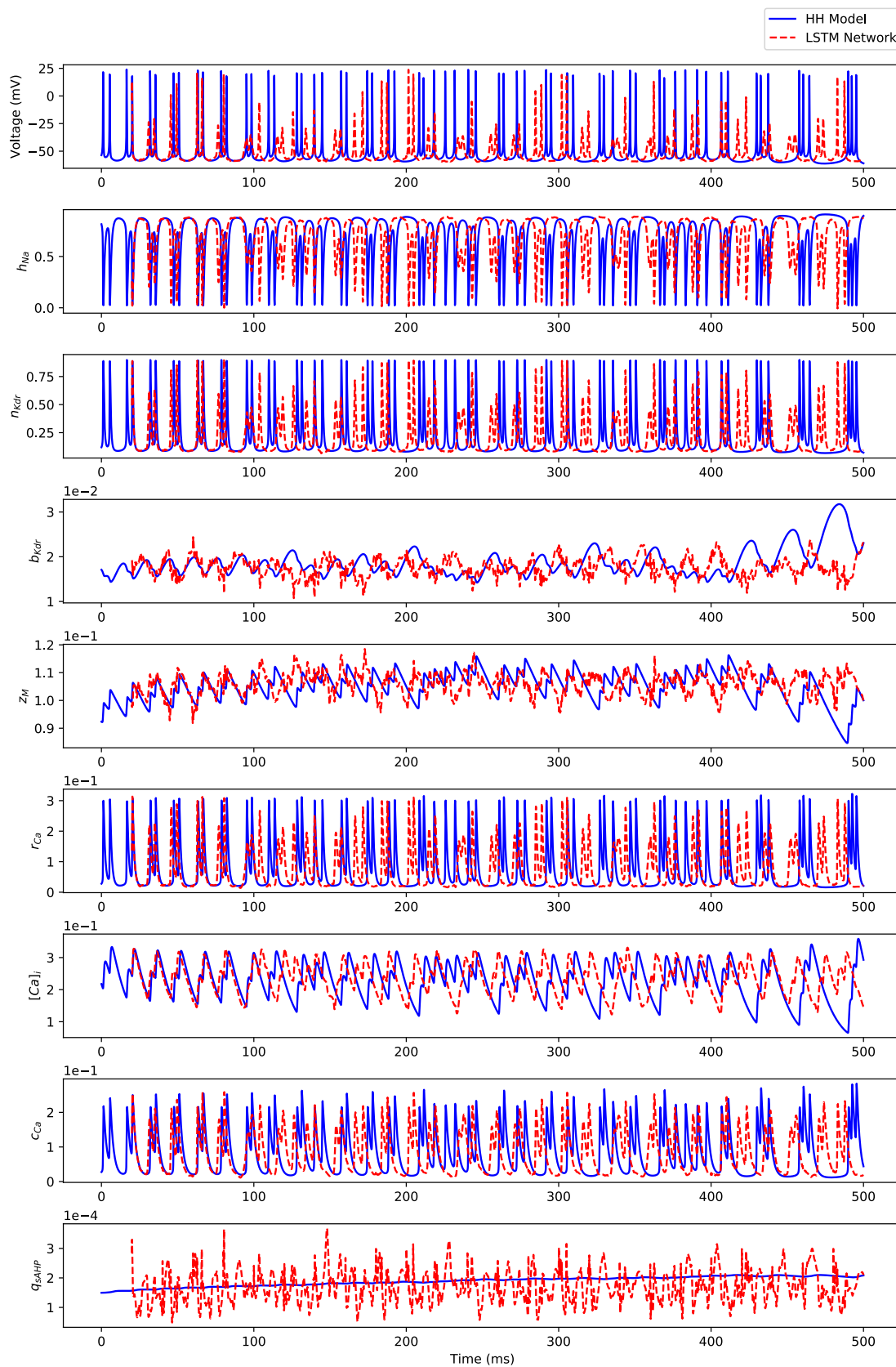
Figure A9. Comparison between the Hodgkin-Huxley model ("HH Model") states' dynamics and the iterative predictions of states' dynamics using the 200 timesteps predictive horizon-based deep LSTM neural network ("LSTM Network") in response to $I = 1.5$ nA.
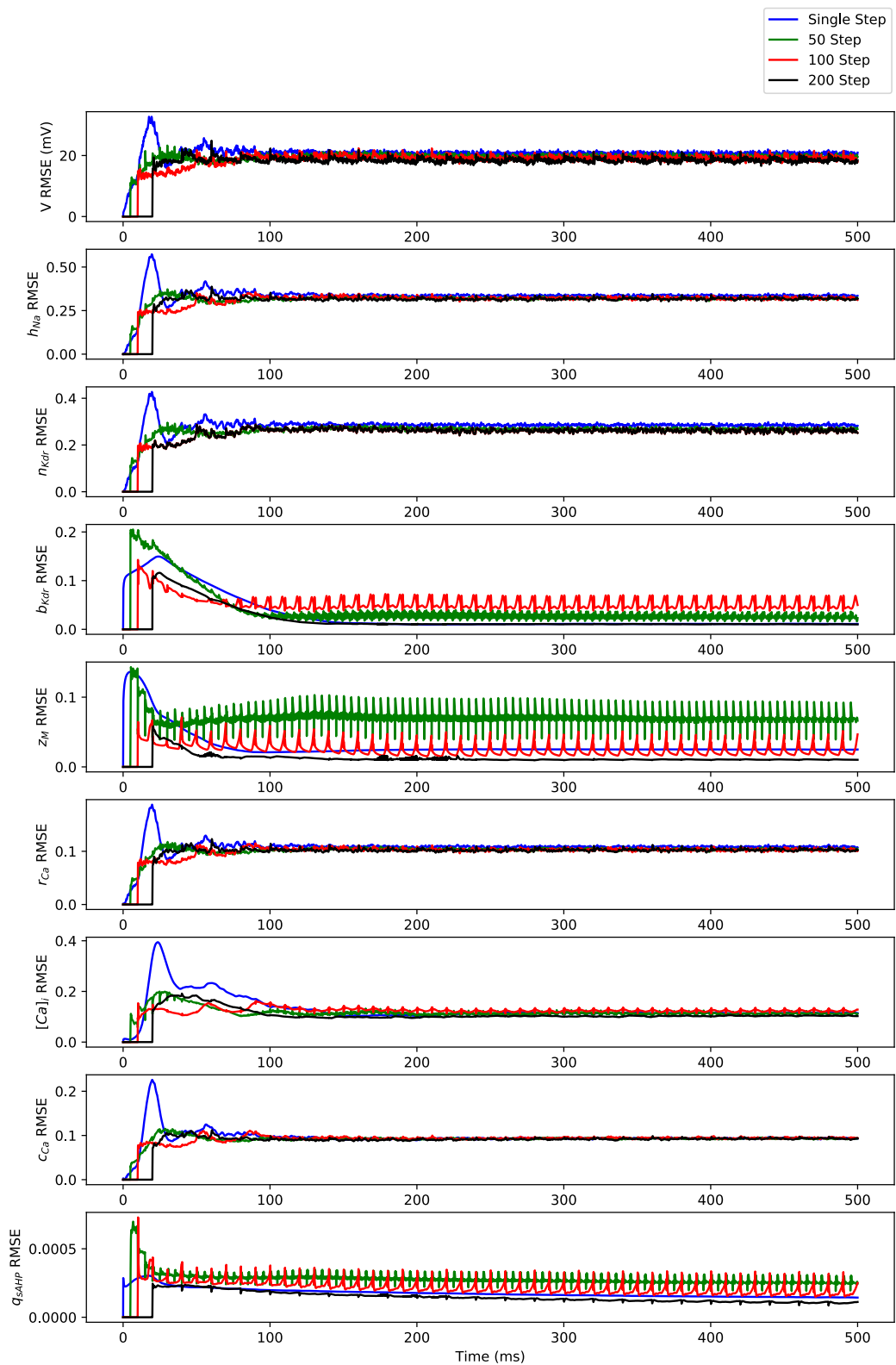
Figure A10. The root mean squared error (RMSE) versus simulation time for 5000 independent realizations, drawn from the predicted membrane potential trajectories of 50 randomly selected stimulating currents from a Uniform distribution $\mathcal{U}(0.79, 2.3)$ and 100 random initial conditions for each stimulating current.
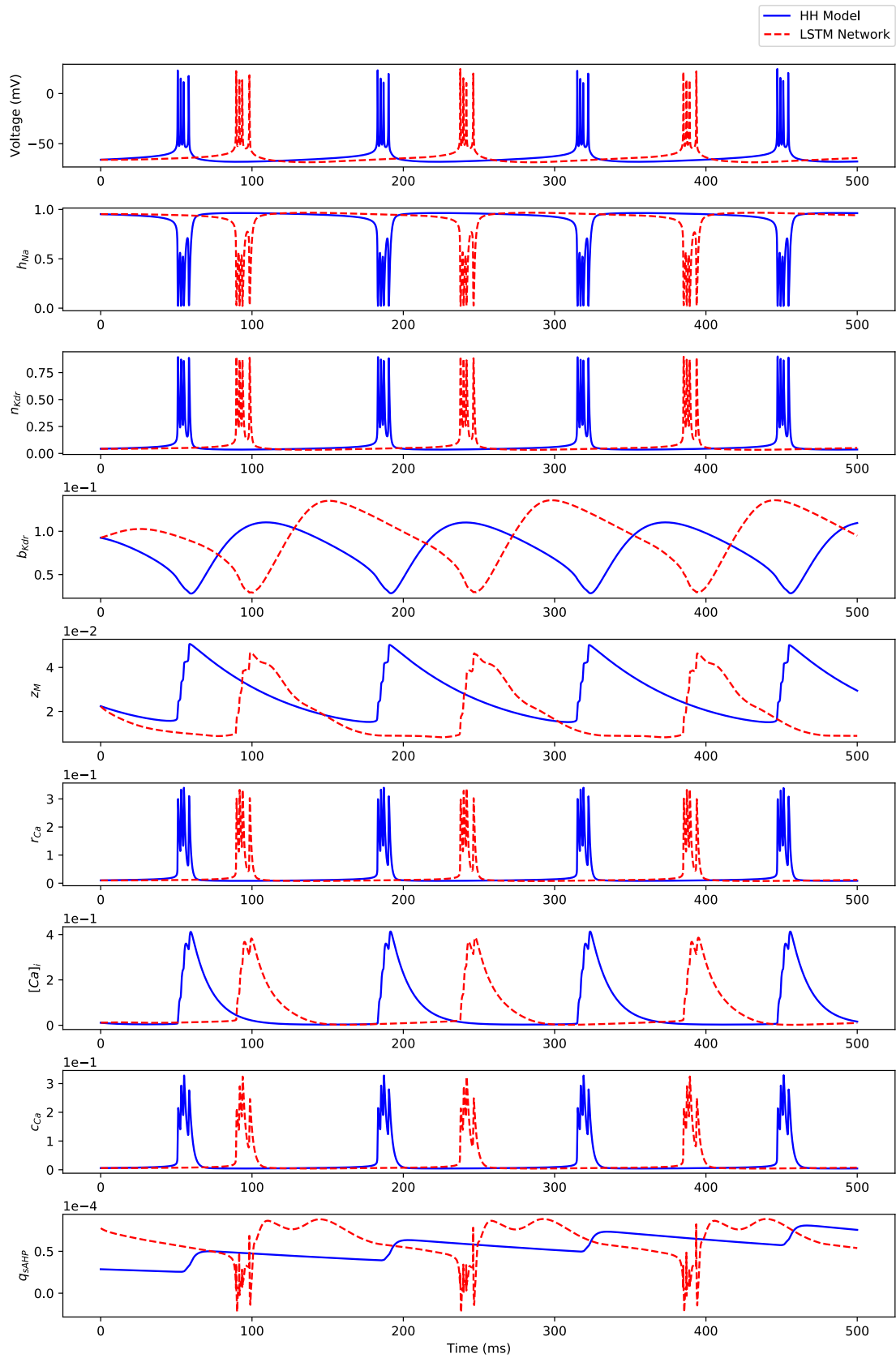
Figure A11. Comparison between the Hodgkin-Huxley model ("HH Model") states' dynamics and the iterative predictions of states' dynamics using the 1 timestep predictive horizon-based deep LSTM neural network ("LSTM Network") in response to $I = 0.5$ nA.
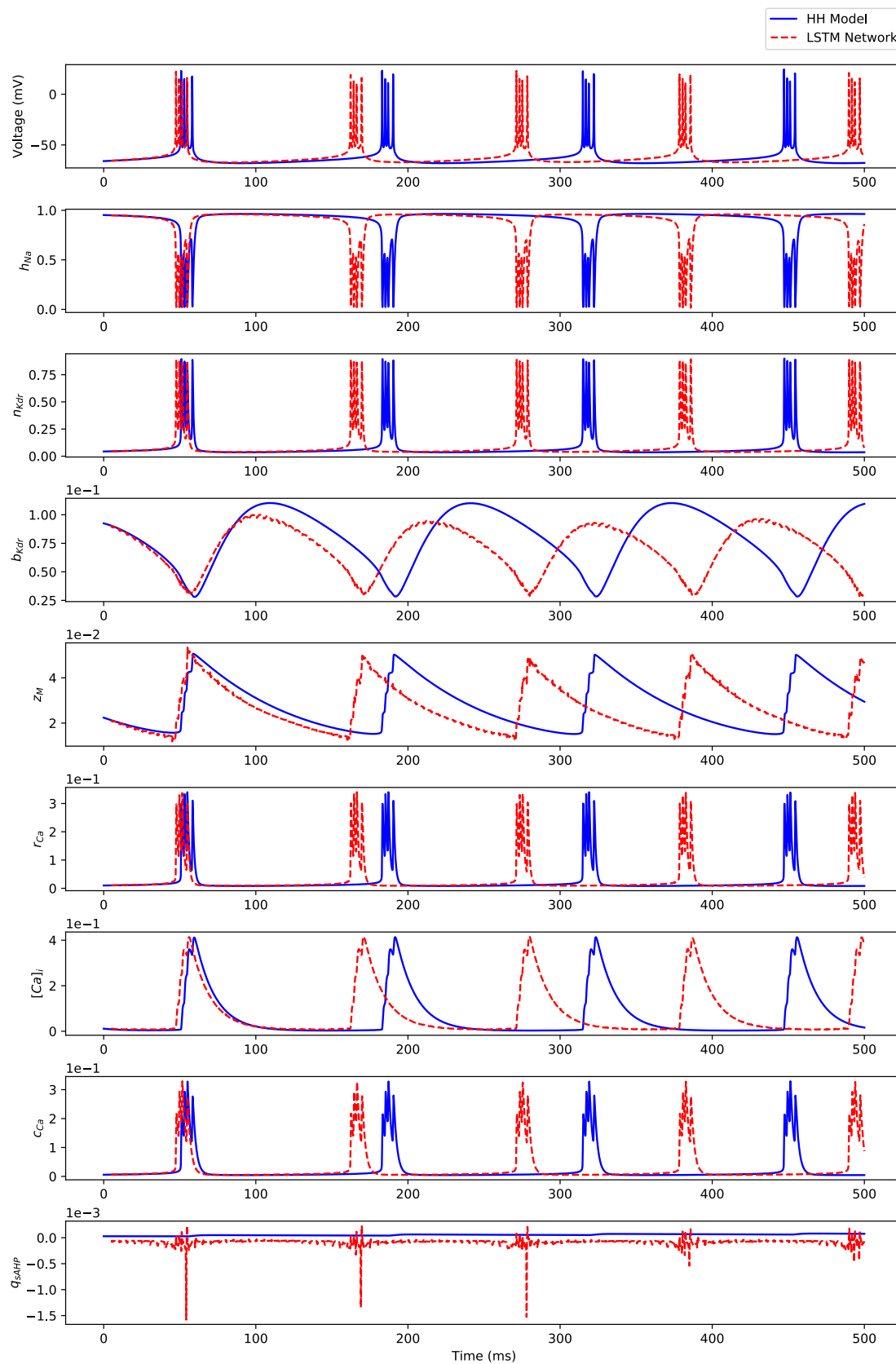
Figure A12. Comparison between the Hodgkin-Huxley model ("HH Model") states' dynamics and the iterative predictions of states' dynamics using the 50 timesteps predictive horizon-based deep LSTM neural network ("LSTM Network") in response to $I = 0.5$ nA.
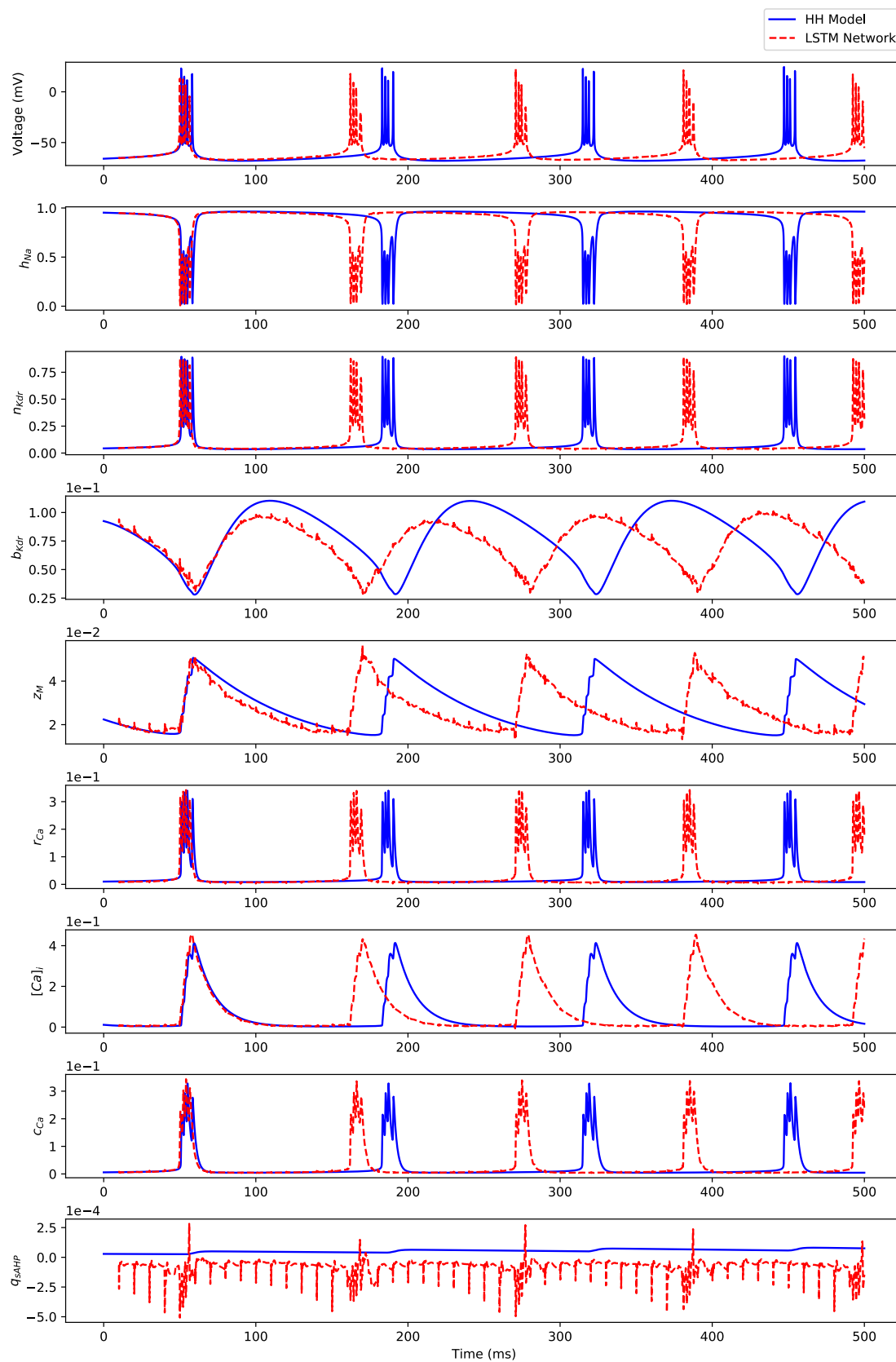
Figure A13. Comparison between the Hodgkin-Huxley model ("HH Model") states' dynamics and the iterative predictions of states' dynamics using the 100 timesteps predictive horizon-based deep LSTM neural network ("LSTM Network") in response to $I = 0.5$ nA.
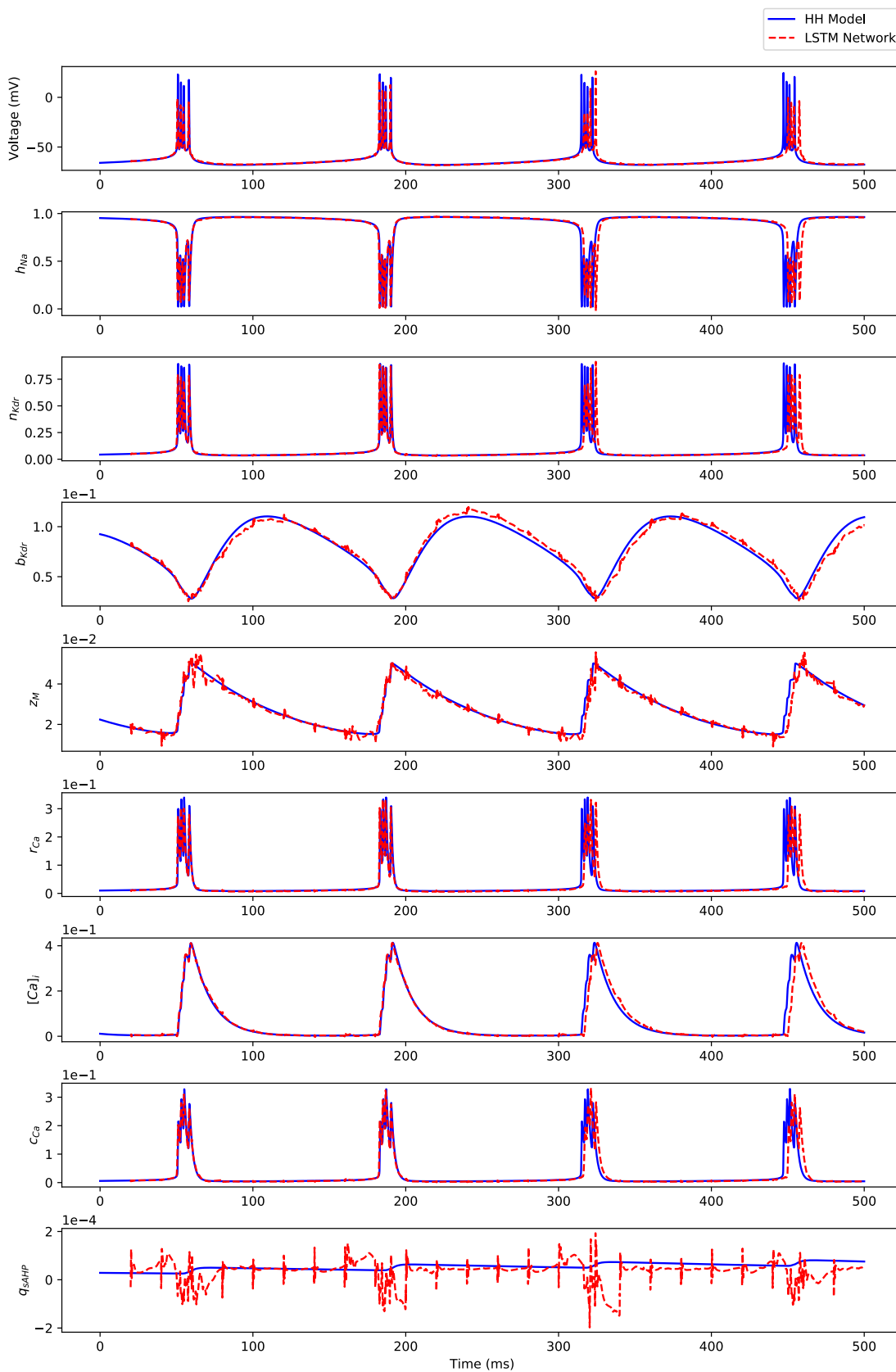
Figure A14. Comparison between the Hodgkin-Huxley model ("HH Model") states' dynamics and the iterative predictions of states' dynamics using the 200 timesteps predictive horizon-based deep LSTM neural network ("LSTM Network") in response to $I = 0.5$ nA.
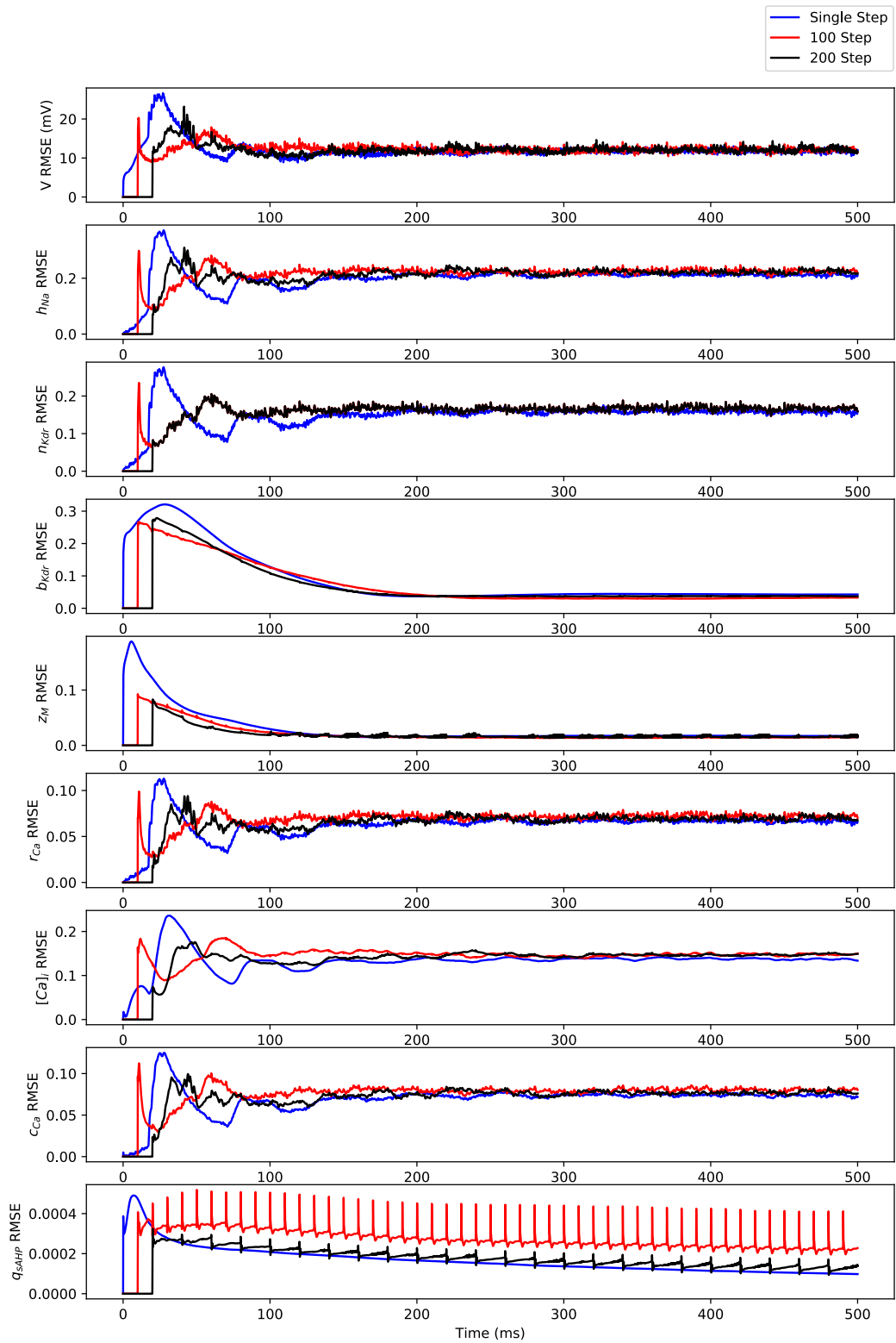
Figure A15. The root mean squared error (RMSE) versus simulation time for 5000 independent realizations, drawn from the predicted membrane potential trajectories of 50 randomly selected stimulating currents from a Uniform distribution $\mathcal{U}(0.24, 0.79)$ and 100 random initial conditions for each stimulating current.