Predicting Local Field Potentials with Recurrent Neural Networks*

Louis Kim[1], Jacob Harer[1], Akshay Rangamani[2], James Moran[1], Philip D. Parks[1], Alik Widge[3], Emad Eskandar[3], Darin Dougherty[3], Sang (Peter) Chin[1,4]

***Abstract*— We present a Recurrent Neural Network using LSTM (Long Short Term Memory) that is capable of modeling and predicting Local Field Potentials. We train and test the network on real data recorded from epilepsy patients. We construct networks that predict multi-channel LFPs for 1, 10, and 100 milliseconds forward in time. Our results show that prediction using LSTM outperforms regression when predicting 10 and 100 millisecond forward in time.**

I. Introduction

Local Field Potentials (LFPs) are recorded by implanting an electrode in the brain and measuring the voltage caused by the activations of the surrounding neurons within a few microns of an electrode. Because the resolution of LFP includes many neurons it cannot measure high frequency data such as single neuron spikes but often contains information in lower frequency bands below 500Hz. Interest in Local Field Potentials is rising because of their lower power costs to record, allowing continuous cortical recordings over long periods of time (years). LFPs have been part of studies that have investigated sensory processing [1, 2], motor planning [3], etc. There is also increasing evidence that manipulating specific features of the LFP can improve the symptoms of neuro-psychiatric disorders [4, 5]. LFPs show oscillatory behavior at 1-50Hz frequencies (periods of 20-1000ms), implying that the neural generators have a fluctuating state even without control inputs. Control of such oscillating systems often benefits from a model-predictive framework. By modeling the expected future state of the system with and without a given control input (brain stimulation), the output (LFP) can be more accurately controlled to the desired endpoint. We therefore sought to create accurate models of LFP dynamics as a framework for predictive control.

Local Field Potentials have been modeled using a variety of methods in the past. They have been modeled using a recurrent network of inhibitory and excitatory point-like integrate and fire neurons in [6]. A biophysical inverse current source density analysis is described in [7] to estimate the current sources that contribute to the LFPs. The biophysical method in [8] is used to model the spatial reach of LFPs and the factors it depends on. All three methods are based on biophysical models of LFPs and are motivated by discovering underlying properties such as source neurons, correlations between their firing, etc. Moreover, [6] and [8] evaluate their models on simulated data. Our approach is focused on statistical learning approach to modeling and predicting LFPs. Statistical learning approach was used in [9] to model LFPs using a linear autoregressive model. We made first attempt at using artificial neural networks to model LFPs [10]. In this work, we expand on our recent work to demonstrate that recurrent neural networks can successfully learn the patterns of LFPs relatively well and outperform regression model in predicting LFPs 10ms and 100ms forward in time.

Table 1. Summary of datasets used.

	PATIENT 1	PATIENT 2	PATIENT 3
RECORDING DURATION	~12 minutes	~35 minutes	~39 minutes
RECORDING LOCATIONS	Dorsal anterior cingulate, dorsal posterior cingulate, ventral anterior cingulate (*Patient 1 and 2 only*), medial orbitofrontal (*Patient 2 and 3 only*), dorsolateral prefrontal cortex (DLPFC), ventrolateral prefrontal cortex (VLPFC), amygdala, hippocampus, superficial anterior temporal		
NUMBER OF RECORDING CHANNELS	82	98	89
STIMULATION LOCATION	Amygdala	Amygdala	Dorsal posterior cingulate
NUMBER OF STIMULATION PERIODS	69	260	265
DURATION OF A STIMULATION	400ms	400ms	400ms

II. Data

The dataset we are using was collected using a Blackrock neural signal amplifier, connected to PMT epilepsy monitoring electrodes. Each high-density electrode consists of multiple probe points (recording channels) which measure LFPs.

Recordings were taken from three different patients. Details are shown in Table 1. Recording time varies between 12 minutes to 40 minutes depending on patient. The locations and number of electrodes used for each patient vary, and as such networks were trained individually for each patient. Because of this variations

*This work was supported by the Defense Advanced Research Projects Agency (DARPA), Biological Technologies Office (BTO), under contract number W911NF-14-2-0045. The opinions presented are those of the authors alone and not of DARPA, Draper, or the Massachusetts General Hospital. Akshay Rangamani and Sang Chin are in part supported by Air Force Office of Scientific Research grant (FA9550-12-1-0136) and National Science Foundation grant (NSF-DMS-1222567).

[1] The Charles Stark Draper Laboratory, Inc, Cambridge, MA 02141 USA. (e-mail: {lkim, jharer, jmoran, pparks, schin}@draper.com).

[2] Department of Electrical & Computer Engineering, Johns Hopkins University, Baltimore, MD 21218 USA. (e-mail: rangamani.akshay@jhu.edu).

[3] Massachusetts General Hospital, Boston, MA 02114 USA. (e-mail: {awidge, eeskandar, ddougherty}@partners.org).

[4] Department of Computer Science, Boston University, Boston, MA 02215 USA. (e-mail: spchin@cs.bu.edu).

in the locations and numbers of electrodes between patients, networks trained on one patient cannot be applied to other patients. In practice, this means networks would have to be trained on the target patient before use. The same type of electrodes and recording technology were used for all patients. During recording patients were asked not to perform any task. The electrodes were sampled at a frequency of 1kHz, since we are limited in temporal resolution beyond that. Table 1 summarizes the datasets used.

The majority of the electrode sites were used to measure the LFPs at their location. However, two electrodes in each dataset were used to provide stimulation instead. Stimulation consists of applying an AC voltage between the two electrodes with a set frequency, duration and amplitude. The frequency and amplitude of stimulation were varied during the course of the recording in order to measure the effect of different stimulation parameters.

In order to prevent overfitting, models were trained on a first 300 seconds sequence of data and tested on a second 300 seconds sequence. Both sequences contain several stimulations and periods of time between stimulations.

III. Long Short Term Memory Networks

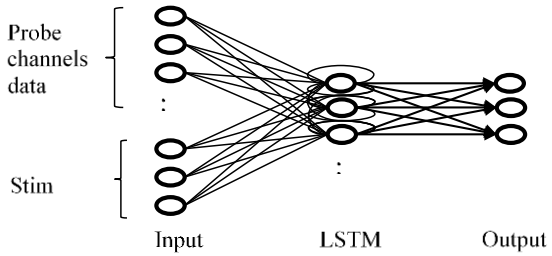
The networks used in this paper are recurrent neural networks (RNN's) built using Long Short Term Memory (LSTM) [11] units. RNN's expand upon traditional feed forward neural networks by also allowing connections back down the network. These backwards connections form dependencies where the current state of the network is dependent upon the previous state of the network. For simplicity in training this is usually limited to connections to the same layer, meaning a layer's current state at time *T* is dependent both upon the input to it at time *T* and upon its previous state at time *T-1*.

In addition to this standard feedback path which is included for all RNN's, LSTM layers also include a second state known as the cells state. Information in this state is directly passed from one time step to the next. The LSTM controls the value of this state through the use of an input gate, forget Gate, and output gate. The input gate regulates when the input affects the cell state, the forget gate regulates if the previous cell state value should affect the current, and the output gate controls if the cell state affects the output of the Neuron. The behavior of each gate is learned during training along with the standard weights. The cell state gives LSTM neurons the ability to store information for arbitrarily long periods of time and as such make them useful in cases with dependencies on data far in the past.

IV. Prediction

We predicted the value of the sampled signal 1, 10, and 100ms forward in time. At each time step *T* the input is a vector containing the sampled values at time *T*. The desired output is then the value of the sample at time *T + the desired prediction distance*. In addition to the LFP signals across all channels, input to the prediction models also included a vector consisting of a bit which indicates if a stimulation is occurring followed by the amplitude and duration of the stimulation. The LFP measurements and stimulation vector were concatenated together to form the input.

Figure 1. Multi-channel network architecture.



Three different prediction models were used, *Previous*, *Regression*, and *LSTM*. The *Previous* model simply uses the single most recent data sample as its prediction. The *Regression* model uses an autoregressive approach which takes a concatenation of the previous 10 data samples in order to predict future LFP value. We used an order 10 regression model as a reasonable benchmark since [9] used an order 7 autoregressive model. The *LSTM* prediction model used LSTM units as discussed above, and is further explained in the following section. For each of the 1, 10, and 100ms forward predictions, mean absolute error was computed and averaged over all three patient datasets as a metric to compare the performance of three prediction models.

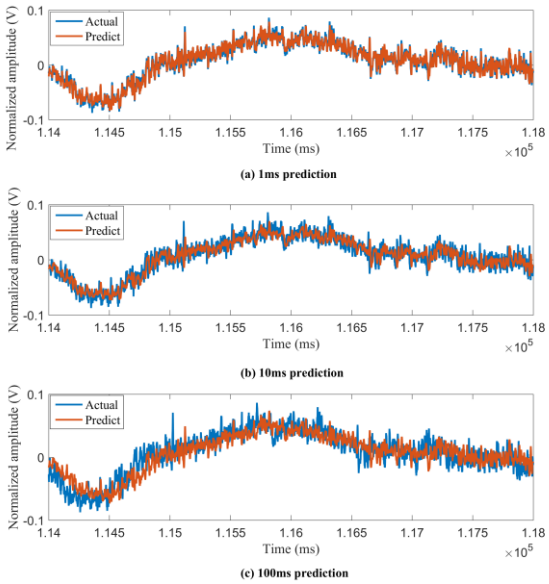
A. LSTM Network Architecture and Training

The LSTM networks we used were constructed and trained using the deep learning framework, Caffe [12]. We trained two different networks, which differ only by their number of LSTM layers. The first network follows a standard 3 layer neural network approach with input, hidden, and output layers. In this case the hidden layer contains LSTM neurons. The network is fully connected, meaning each LSTM layer has a forward connection to every input and a recurrent connection to every LSTM neuron. The output layer contains a neuron for each probe point and each is fully connected to the LSTM hidden layer. Fig. 1 shows the structure of this network architecture.

The second network is the same architecture as the first except it contains two hidden layers. The second layer is fully connected to the first LSTM layer and the output layer connects directly to it. The addition of more hidden layers generally allows for the network to learn more complex representations of the data.

For training, the weights of the networks were initialized randomly using a Gaussian distribution with a standard deviation of 0.01. Training was done by stochastic gradient descent with momentum of 0.9. Initial learning rate was set to 0.1, and decreased every 20,000 iterations by a factor of 10. The network was trained using mini-batches of 200 time steps at a time. The networks hidden layers each contain 2000 LSTM neurons. Ideal number of neurons for the hidden layer of the network and the initial learning rate were

Figure 2. LSTM prediction results for Patient 1, Channel 60, predicting (a) 1ms, (b) 10ms, and (c) 100ms forward in time.



chosen based on their sensitivities to the prediction errors shown in Section IV *C*.

B. Results

For each of the networks described above we trained a network to predict 1, 10, and 100ms out for each of the three patients used. We calculated the mean absolute error of the testing data and averaged it across all patients. The single layer LSTM network achieved mean absolute errors of 0.014, 0.026, and 0.060 for 1, 10, and 100ms forward predictions. Its prediction loses accuracy as we predict 10ms and 100ms, but it still maintains the overall shape. This results indicate that the network was able to learn the patterns of the data relatively well. This is backed up visually by Fig. 2 which shows the prediction results for channel 60 of Patient 1.
The high frequency oscillations in the data are largely lost at 100ms, but the overall shape is maintained.

Fig. 4 compares the overall prediction errors across all three patient datasets for the networks described previously. Surprisingly, *Previous* and *Regression* performed better than both *LSTM* networks when predicting 1ms forward. The success of regression here is most likely due to the linear nature of the data when only looking 1 time step into the future. Fig. 3 shows a zoomed in view of a portion of data for 1ms prediction times. Due to the overarching oscillations in the data it makes sense that a linear model would be able to predict the next point with high accuracy. This assertion might also explain the failure of *LSTM* to do better than *Previous*. If the data is linear the LSTM network will be required to approximate the linear model with its nonlinear activations. This may be difficult given the number of LSTM neurons is not orders of magnitude more than the number of outputs.

Figure 3. LSTM 1ms prediction results for Patient 1, Channel 60, zoomed in on a 50ms period.

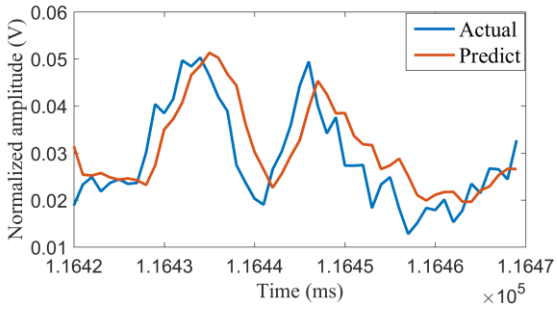
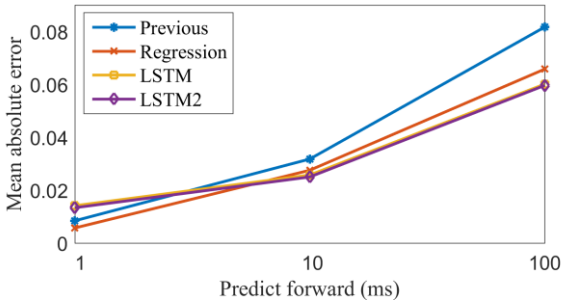


Figure 4. Prediction error (average across three patient datasets) comparison between *Previous*, *Regression*, and *LSTM* models.



The LSTM does outperform the other models when predicting 10ms and 100ms forward. This suggests that there is some dependency on past information beyond 10 time steps which is required for accurate prediction that far into the future. Interestingly though, the additional layer of LSTM neurons seems to make very little difference, yielding about 1% improvement in errors between the two LSTM networks for all 1, 10, and 100ms forward predictions. This implies that the features captured by the first layer cannot be easily improved upon by the addition of a second layer.

Although the LSTM models does outperform the other models it is surprising that they do not do so more. There are many possible explanations for this. It is certainly possible that a more complex network will significantly outperform our current LSTM's. However, the fact that we see insignificant improvement of the two layers LSTM over the single layer seems to suggest that a more complex network does not help this issue. It is also possible that we do not have enough information from the limited number of electrodes we are measuring to predict the activity which governs the brain region we are observing.

C. Sensitivity Analysis

When training neural networks, the hyper parameters such as learning rate and number of neurons often have to be tuned appropriately to minimize the loss of the network. We conducted a sensitivity analysis to analyze the sensitivity of the prediction error to the initial learning rate and the number of neurons. Results are shown in Fig. 5 and Fig. 6. Analysis was done only on patient 1 because of time constraints, since enumerations of these types require

Figure 5. Prediction error sensitivity to learning rate on Patient 1 dataset, predicting 1ms forward with 200 neurons.

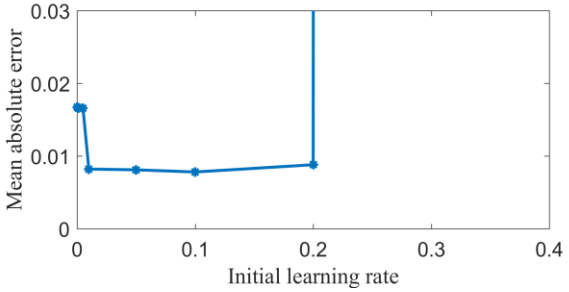
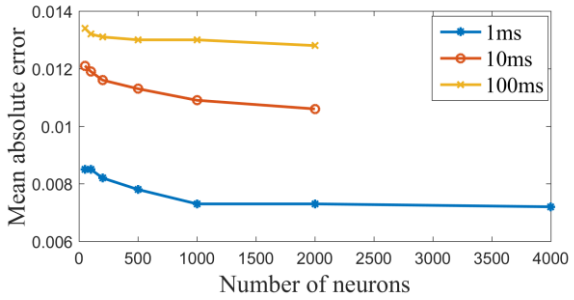


Figure 6. Prediction error sensitivity to number of neurons on Patient 1 dataset with the initial learning rate of 0.1.



retraining the network for each point. We also limited the sensitivity of the prediction error to the number of neurons in the single layer LSTM networks since the addition of a second layer did not seem to provide noticeable improvement. It has also been observed that the number of neurons in the deeper layer itself matters less than the fact that there are more layers [13].

We analyzed the effect of the initial learning rate for a prediction time of 1ms. This was chosen arbitrarily as we expect the results to be similar across all predictions. The results yield an optimal learning rate of 0.1. Upon further inspection they clearly show that an initial learning rate between 0.01 and 0.2 gives reasonable results.

We explored the effect of the number of neurons over all predictions. The error seems to continue to improve as we increase the number of hidden neurons up to a point. This suggests that a large number of neurons is necessary to correctly predict this data set. This makes sense as we would expect data obtained from a neurological source to have a very complex underlying distribution which depends on many factors. That being said, the improvement seems to level off after 2000 neurons, yielding significantly reduced improvement for large increases in layer size.

V. Conclusion and Future Work

We have demonstrated preliminary results that suggest that Recurrent Neural Networks (RNNs) can be used to model and predict Local Field Potentials. In future work, we plan to extend our work by taking the morlet wavelet transform of the data and training on it. The wavelet transform should remove unwanted noise in the signal as well as allowing the network to focus on individual neural frequencies. Additionally, we want to explore additional training methods for recurrent network such as hessian-free optimization in order to better train our networks to take advantages of the long term dependencies inherent in this type of data.

References

[1] A. Belitski et al., "Low-frequency local field potentials and spikes in primary visual cortex convey independent visual information," in *J Neurosci.* 2008; 28(22):5696-709.

[2] F. D. Szymanski et al., "The laminar and temporal structure of stimulus information in the phase of field potentials of auditory cortex," in *J Neurosci*. 2011; 31(44):15787-801.

[3] H. Scherberger et al., "Cortical local field potential encodes movement intentions in the posterior parietal cortex," in *Neuron*. 2005; 46(2):347-54.

[4] A. S. Widge and D. D. Dougherty, "Deep brain stimulation for treatment-refractory mood and obsessive-compulsive disorders," in *Current Behavioral Neuroscience Reports*. 2015; 2(4):187-97.

[5] E. J. Quinn et al., "Beta oscillations in freely moving Parkinson's subjects are attenuated during deep brain stimulation," in *Mov Disord*. 2015; 30(13):1750-58.

[6] A. Mazzoni et al., "Encoding of naturalistic stimuli by local field potential spectra in networks of excitatory and inhibitory neurons," in *PLoS Compu Biol*. 2008; 4(12):e1000239.

[7] K. H. Pettersen et al., "Current-source density estimation based on inversion of electrostatic forward solution: effects of finite extent of neuronal activity and conductivity discontinuities," in *J Neurosci Methods*. 2006; 154(1-2):116-33.

[8] H. Linden, et al., "Modeling the spatial reach of the LFP," in *Neuron*. 2011; 72(5): 859-72.

[9] D. Huberdeau et al., "Analysis of local field potential signals: a systems approach," *Engineering in Medicine and Biology Society (EMBC), 2011 Annual International Conference of the IEEE*, Boston, 2011; pp. 814-7.

[10] A. Rangamani et al., "Modeling local field potentials with recurrent neural networks," *29th Annual Conference on Neural Information Processing Systems (NIPS)*, Montreal, 2015.

[11] S. Hochreiter et al., "Long short-term memory," in *Neural Computation*, 1997; 9(8):1735-80.

[12] Y Jia et al., "Caffe: Convolutional Architecture for Fast Feature Embedding," 2014.

[13] A. Graves, et al., "Speech recognition with deep recurrent neural networks," IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Vancouver, 2013; pp. 6645-49.