Abstract
We present a Recurrent Neural Network using LSTM (Long Short Term Memory) units that is capable of modeling Local Field Potentials. We train and test the network on real data recorded from patients. We construct networks that are capable of modeling LFPs in single and multiple channels, and can also model LFP dynamics after stimulation.

1 Introduction
Local Field Potentials (LFPs) are a type of electrical signal that are recorded from the cortex, and capture the activity of the surrounding neurons within a few microns of an electrode. LFPs are low frequency signals that are usually in the <500Hz band, as opposed to neural spikes that are in the higher end of the spectrum. Interest in Local Field Potentials is rising due to the ability to obtain continuous cortical recordings, and the fact that they capture processes that cannot be observed from the spiking of single neurons. LFPs have been part of studies that have investigated sensory processing [6], motor planning [5], etc.
Attempts have been made at modeling the physics of Local Field Potentials using a Current Source Density analysis [8] among other frameworks. Time Frequency features from LFPs have been used in classifiers for various tasks [7]. However to the best of our knowledge, this is the first attempt at using artificial neural networks, specifically recurrent ones, to model LFPs.

2 Dataset
The dataset we are using was collected from the Defense Advanced Research Projects Agency’s (DARPA) System-Based Neurotechnology for Emerging Therapies (SUBNETS) program. The system “consists of high-density electrodes attached to multiple satellite electronics modules that are distributed around a centralized hub electronics module that provides command and control. Each satellite is connected to a high density electrode, and a cable which connects to the central hub, both of which are hardwired connections.” [1] Each high-density electrode consists of multiple probe points measuring LFP. These electrodes are
embedded in the scalp. Each dataset we used contains around 30 min of recording from patients in an Epilepsy Monitoring Unit (EMU). The LFP was sampled at 1 kHz, since we are only interested in low frequency bands. The majority of the electrode sites where used to measure the LFP at their location. However, two electrodes in each dataset were used to provide stimulation instead. Each stimulation consists of applying a voltage between the two electrodes with a set duration and amplitude.

3 Sparse Networks

In this section we try to learn a representation for the data using a sparse auto-encoder. An autoencoder is an unsupervised learning algorithm that tries to minimize the reconstruction error of the representation [3].

We first focus on parts of the signal with no stimulation being applied. We take a 500ms segments of the data and construct a training set for the autoencoder. The network that was used consisted of 83 input units (corresponding to the number of channels), 10 hidden units, and 83 output units. Here we can see that at first the reconstruction error is relatively large but with larger data dimension, the error decreases dramatically. In the second part of Figure 2 we look at parts of the signal where stimulation is applied. Segments of 2ms duration are concatenated to construct the training set. It is interesting to note that the Euclidean distance seem to spike around the time of stimulation, suggesting the representation learned by the sparse autoencoder is not consistent with LFPs post stimulation.

4 Long Short Term Memory Networks

The networks used in the remainder this paper involve Long Short Term Memory (LSTM) [2] units. LSTM units are a type of recurrent neural network (RNN) architecture, meaning the output of all neurons in each LSTM layer is fed back around to the input of that layer. In addition to this standard feedback path LSTM layers also include an input gate, forget Gate, and output gate control how data is written to the units. During training the behavior of each gate is learned as well as the behavior of the neuron. The addition of these gates makes LSTM layers well suited to learn long term dependencies in a sequence with potentially large time lags between important events.

5 Single Channel Prediction

In this section we focus on data from a single channel of the recorded LFP data. We try to predict the next sample of the signal using the present sample and past information. Since we are trying to model a real valued signal, the output of a single time step of the LSTM network is a real number that is the prediction made by the network for the next sample.

5.1 Network Architecture and Training

The neural network was constructed and trained using the deep learning framework, Caffe [4]. The LFP input layer connects to a 50 wide LSTM layer. The LSTM units are linearly combined to give a single output in the final layer. The network was trained on half the data from a single channel, the other half set aside for
testing. For the single channel case, we focused on the segments of the data without stimulation. The weights of the network were initialized randomly using a Gaussian distribution with a standard deviation of 0.01. Training was accomplished by stochastic gradient descent using a momentum of 0.9. The learning rate was set to 0.1 initially, and decreased every 20000 iterations by a factor of 10. In each iteration, we used a batch of 10 time steps to train the network. Since LSTMs potentially suffer exploding gradients, we enforced a hard constraint on the magnitude of the gradient by clipping it using a threshold.

5.2 Experimental Results
The single LSTM layer neural network was able to learn to predict the LFP signal quite successfully, achieving an average least squares error of 0.001 on the training set and 0.0012 on the test set. The results are shown in the plots below.

In the frequency domain, we can see that our prediction and the target do not entirely match in the higher frequency regions. However in the low frequency regions, our performance is quite good. In LFP analysis we are mostly interested in signals in the lower frequency range, below 100Hz. Our model is in agreement with the target signal in the regions we care about.

6 Multi Channel Prediction
In this section we expand the single channel case to all channels for a given dataset. We expanded both the input and output layers of the network from section 5 to include all 98 channels of input data. The hidden (LSTM) layer of the network was also expanded to 2000 neurons. Additionally, the network was changed to include a vector containing stimulation data. This vector consisted of a bit which indicates if a stimulation is occurring followed by the amplitude and duration of the stimulation. The LFP measurements and stim vector were concatenated together to form the input vector.
6.1 Training
The network was trained on a 100 second sequence of data which contains several stimulations and periods of time between stimulations. It was tested on a different 100 second sequence taken from the same data set. Beyond the larger training and testing data the details of training remained the same as in the single channel case.

6.2 Experimental Results
The network achieved an average least squares error of $1.046 \times 10^{-4}$, although the maximum least squares error was 0.0215. Results are shown in the plots below.

As can be seen from figure 5 the prediction is not always exact. But what is interesting is that although the prediction seems to lose the exact voltages in certain points it still maintains pretty similar shape. This hints that the network was able to learn the patterns of the data relatively well even if not the exact voltages.

7 Conclusion and Future Work
We have demonstrated preliminary results that suggest that Recurrent Neural Networks (RNNs) can be used to model Local Field Potentials. The above results are trained and tested on real world data recorded from patients. In future work we would like to explore the LSTM network in detail, and try to understand how exactly the network models the dynamics of cortical recordings. We would also like to extract time frequency features from LFPs and try to use them to perform classification and other prediction tasks in the RNN-LSTM paradigm.
References


