Predicting seizures using spiking artificial neural networks

By Jeremy Angel and Matthew Wootten

Background/Past Research

Artificial neural networks have two broad properties: their *topology* (how the neurons are connected) and the behavior of the neurons. These two components make up the neural network *architecture*.

Before any data flows into the network, it undergoes preprocessing. There are a number of possible methods we may use to transform the raw data into a form suitable for input into the neural net. One trivial approach is simply setting cutoffs on the incoming signal, and firing whenever the value passes the cutoff. However, this is probably not the route we will take, because we have not encountered research where this approach is used, and so are very uncertain of its properties. Another option, used by Ghosh-Dastidar and Adeli (2009), extracts a set of nine parameters of the waveform and uses those as inputs. This approach makes input simplest, but may remove some inherent spike structure from the waveform. Lastly, we could use the conventional input for audio (another waveform format), and input many instantaneous samples for a short period (maybe a couple seconds) a large number of times, and rely on the convolutional portion of the network to recognize spikes. This has the advantage of being widely used, but the disadvantage of being designed primarily for non-spiking networks.

The next stop for the data is the neural network itself, composed of (possibly) convolutional layers followed by feed-forward fully-connected layers. The convolutional layers have one neuron for each pair/triple/*n*-tuple of input values, which help to combine adjacent values together and thus detect patterns among small sections rather than individual neurons, and to reduce the size of the input. It then passes on to fully connected layers, where each neuron is connected to every other neuron in the next layer. The neurons themselves are *continuous* and *spiking*: they have (nearly) continuous input, which the neuron accepts until it reaches a threshold and fires. These neurons are also known as *integrate-and-fire neurons*, because of how the accumulation is modeled. An activation is then applied: we expect to use a standard function such as sigmoid, tanh, or ReLU.

Lastly comes the post-processing. This is a very simple step - simply choosing the output format (likely a neuron with a number of input spikes corresponding to the chance of a seizure in the future), and them computing a statistic such as precision/recall from the predictions.

We will train the networks using backpropagation, specifically the variant specified by Lee, Delbruck, and Pfeiffer (2016). This is very similar to standard backpropagation.

Overview/Purpose

We have identified nine parameters of the neural network: the neuron class (spiking/non spiking), the spike continuity (continuous/discrete), the layer types (convolutional/feed-forward-only), the training algorithm, the input combination operation, the activation function, the preprocessing, and the properties of each spike transmission / spike count (single spike/multiple spikes). We plan to hold each of these constant except for spike count, though this may possibly change to better accommodate collaboration.

We will be attempting to predict seizures before they occur. We are confident that the prediction task has not been attempted before using spiking neural networks, though the somewhat related detection (i.e. after it has begun) task has succeeded. We hope that this will lead to a safer life for epileptics, as they can foresee seizures and take corrective action quickly.

Objectives

This research project has four primary objectives: building the neural network, training it, determining and running the preprocessing, and running the network through the actual EEG data. Each primary objective consists of two to three secondary objectives, except for running through the data which only involves just that. Each secondary objective falls under the broad category of describing, building or testing.

Each primary objective is a piece in the overall machine learning process using the convolutional spiking neural network that will be used to predict seizures.

Building the neural network will consist of three secondary objectives: writing out a mathematical description of the network to get a deeper understanding of what is going on in the network, building the network by either constructing it manually or by editing some framework for spiking neural networks if they are shown to fit properly such as the Brian simulator, and testing the network using a simple, small dataset and a training algorithm that is quick to implement. The Brian simulator is a spiking neural network simulator that has been used for machine learning in somewhat similar problems such as image classification.

Training the neural network will have two parts: building the training algorithm and testing it. The testing will be done on the same simple, small dataset used to test the neural network and will be used to weed out any bugs. This will help to screen the training algorithm after it has been built before it is used for running through the spiking neural network.

The next primary objective of determining and running the preprocessing consists of determining what preprocessing to use, building it, and testing it on a small piece of the EEG data. First, the preprocessing that would be best to use on the EEG data for the spiking neural network will have to be determined. Then, either a preprocessing routine will have to be built or found to use to complete the initial inputs for the neural network. In either case, the preprocessing routine will the be tested on a small piece of the EEG data to weed out any bugs before use with the full data.

The final primary objective will be to run the full EEG data through the neural network previously constructed. This may involve some trial and error, especially depending on optimization and how much computing power is available. Ideally, after this final objective, results will be obtained from the EEG data.

Timetable (highly approximate)

Task	**Estimated time**
Background, experimental design, etc	3 weeks
Building neural network Describing Building Testing	6 weeks 2 week 3 weeks 1 week
Building training algorithm Building Testing	4 weeks 3 weeks 1 week
Preprocessing Describing Building Testing	6 weeks 2 weeks 3 weeks 1 week
Subsamples of real data	1 week
Final analysis Optimization/refactoring Executing	???? ?? ??