Convolutional neural network models for real-time seizure forecasting

Jeremy Angel Matthew Wootten

May 30, 2019

Abstract

1% of people around the world have epilepsy, a debilitating condition which prevents them from functioning in everyday life. Unpredictability causes much of the seizure-associated morbidity, so successful prediction of seizures would alleviate suffering. We propose a regression algorithm for predicting the time until the next seizure, rather than existing models which focus on detecting the preictal (pre-seizure) period. Specifically, our algorithm aims to predict, given a five-second electroencephalogram (EEG) recording that occurs before a seizure, how much time remains before that next seizure. While attempting to predict the time until next seizure largely succeeds with models that are provided samples from the same preictal period, this fails to generalize to unseen preictal periods. Results from a wide variety of non-deep-learning models, such as linear regression, multilayer perceptrons, and gradient boosted trees suggest that forecasting is impractical from a single slice. We, therefore, tested convolutional neural networks, reasoning that the filter design would allow them to recognize the same patterns in different features. We also used a custom loss function, in order to de-emphasize the importance of predictions very far from the seizures. As a control, we used an LSTM, given that this is the basis of current methods. Overall, the models all had relatively poor performance, though all models appeared to break down around the same point, suggesting a fundamental limit to prediction. We are continuing to explore hyperparameter tuning, loss function tweaking, and survival analysis to improve our model.

Introduction

1% of people around the world have epilepsy, a debilitating condition which prevents them from functioning in everyday life, constantly at risk of losing consciousness and control at any time (Fiest et al., 2016). Camfield and Camfield, 2010 studied a group of 15-30 year old patients with idiopathic generalized epilepsy (no brain damage outside of epilepsy) and tonic-clonic seizures (seizures that cause whole-body convulsions). They found that 40% could not graduate high school, 33% were unemployed, and 27% had a psychiatric diagnosis. The form of epilepsy these patients had is considered mild; many cases are much worse.

Seizures occur when a defective cluster of neurons sends an unusually strong signal, causing a positive feedback loop where the neuron continuously sends out this same signal. This signal then spreads across the brain, affecting a large area resulting in a seizure. Thus, if we look at brain activity for signs of future abnormality, we may be able to predict an upcoming seizure. (Stringer, 2017)

Many existing seizure predictors use electroencephalograms (EEGs). EEGs are recordings of electrical activity in the brain: specifically, the differences in voltage between pairs of electrodes. Each pair of electrodes measures one channel. Such electrodes can be either intracranial (implanted within the head) or extracranial (attached to the scalp). While extracranial EEGs are much more practical, intracranial EEGs are more sensitive (Kuruvilla & Flink, 2003). An EEG recording is made from a relatively small number of channels, each of which is a voltage signal varying over time. By analyzing these, algorithms in the past have been able to predict seizures. Typically, these algorithms begin by picking out a number of features, little bits of information that previous research has shown are related to seizure occurrence.

However, the precise interactions between all of these features are fantastically complex. Not all patients experience seizures in the same way, so we need some mechanism to combine all the different features into a single prediction. Machine learning fits this specification well. At its core, machine learning (ML) encompasses the set of algorithms that work not by explicit specification, but by relying on some sort of training data to infer a pattern, which may be too complex for a human to interpret. These machine learning algorithms are able to find patterns regardless of what data they are looking at while explicitly specified algorithms are designed for a single problem. Many existing seizure prediction algorithms have been based on machine-learning methods, and the state-of-the-art algorithms are nearly all ML-based.

Artificial neural networks (referred to here as ANNs or just "neural networks"), one example of a machine learning algorithm, are a loose mathematical model of how the brain functions. They have been successfully used to model complex features in images and signals. An ANN is a network of artificial neurons, connected by artificial synapses. The neurons and synapses each have associated values: neurons have biases and synapses have weights. Each neuron takes the values from the neurons connected to it from previous layers, multiplies it by the weights associated with the synapses, adds the bias, and applies some nonlinear function called an *activation function*. The only exceptions to this rule are the input neurons, which have values set to predictor variables. Some neurons serve as outputs, which take on values that encode the prediction. Neurons which are neither inputs nor outputs are called "hidden" neurons; they have no expected value. These networks become useful when they are trained, a procedure that gives example inputs and outputs so that the network "learns" to correctly classify outputs (Schmidhuber, 2015).

We use a category of artificial neural networks known as convolutional neural networks (CNNs). These take advantage of spatial or temporal structure in the data. Instead of having a layer where every neuron is connected to every other, convolutional networks have a series of *filters*, smaller networks which are applied to adjacent slices of each input. This takes advantage of spatial or temporal shift invariance, a property which these recordings have (since the five second cuts have), to reduce the overall size of the network.

For networks to be useful, they must be trained. This entails feeding the network existing pairs of input and output. This is done through a method called back-propagation, based on gradient descent. It works by using automatic differentiation to calculate the gradient of the loss function with respect to the weights and then by nudging the inputs in the direction of greatest decrease of the loss by an amount parameterized by a learning rate.

EEG-based seizure prediction has a nearly two decade history. Prediction systems have been used for two purposes: for their direct utility (in hopes of a better comprehensive seizure predictor) and as a means for testing hypotheses either about seizure biology or about machine learning techniques. We are addressing the problem from the first point of view, since neither our machine learning techniques nor features are independently unique; accordingly, we will primarily consider previous unconstrained attempts. The first system to use neural networks was by Petrosian, Prokhorov, Homan, Dasheiff, and Wunsch, 2000; they used small (10-15 hidden neuron) recurrent networks with both raw and frequency-decomposed EEG waveforms. While this model did correctly identify most seizures, it had a high false-positive rate, and frequently triggered too early. Quyen et al., 2001 used a non-machine learning technique, nonlinear analysis, to predict seizures on average 7 minutes in advance; however, Clercq, Lemmerling, Huffel, and Paesschen, 2003 failed to replicate their findings, suggesting sensitivity to small differences in the measurements.

Later models improved to the point where the differences in efficacy were connected to the inherent tradeoff between false negatives and false positives. D'Alessandro et al., 2005 used a mix of genetic algorithm-based feature selection and shallow neural networks to achieve a 0% false negative rate, at the cost of 1.1 false positives per hour. Netoff, Park, and Parhi, 2009 took the opposite approach, using an SVM to achieve a 0% false positive rate while only catching about 80% of seizures. The newest models are starting to close this gap, however. Tsiouris et al., 2018 use an LSTM to get both a false positive and true positive rate over 99% two hours in advance.

Evidence suggests that the theoretical maximum prediction horizon can likely be extended even farther than this. Mula and Monaco, 2011 notes that a third of epilepsy patients with partial seizures (a common type) report preictal symptoms, often hours or days in advance. More precisely, Litt et al., 2001 showed that clinicians could identify abnormalities in the energy of EEGs seven hours in advance. To our knowledge, no existing algorithm has yet made predictions this far in advance.

Method

Data and pre-processing

We take data from the CHB-MIT Scalp EEG Database (Shoeb, 2009) on the PhysioNet platform (Goldberger et al., 2000). This dataset includes extracranial EEGs for 23 pediatric epilepsy patients containing 185 useful seizure recordings.

We chose a feature-based, rather than pure deep-learning model. Effective deep learning models require a massive set of labeled data (generally millions of training samples) and extraordinary computational power for training. Using domain knowledge to choose features can reduce the burden on the learning algorithm to find appropriate ones. The systems that establish the current state of the art in the field for longer-term prediction, Tsiouris et al. (2018), uses a feature-based model. Until enough annotated seizure data becomes available, feature-based models will likely outperform others.

Broad feature-sets seem to perform better than narrow ones. A quick survey of Acharya, Hagiwara, and Adeli (2018) finds that those with complex feature sets tend to outperform the simpler ones (though some bias may be introduced by the tendency to use experimental features by themselves). As such, we used a variant on the broad feature-set from Tsiouris et al. (2018) and adapted it to a new neural network architecture.

The features include:

- Statistical properties (mean, variance, standard deviation, skewness, and kurtosis)
- Count of zero crossings
- Amplitude range
- Unsigned area bounded by signal
- Total signal energy
- Signal energy percentages in delta, theta, alpha, beta, and gamma bands
- Discrete wavelet decomposition coefficients
- Maximum cross-correlation
- Decorrelation time
- Graph properties: clustering coefficient, betweenness centrality, eccentricity, diameter, radius, and characteristic path

The only difference from Tsiouris et al. (2018) was the lack of eccentricity properties; we did after profiling revealed that approximately 75% of feature calculation time was dedicated to computing eccentricities. We first divide the annotated recording into five-second non-overlapping segments. Segments that are ictal or less than five minutes post-ictal are excluded. Each remaining segment is then tagged with the time from the beginning of the segment to the seizure onset. Each feature is then computed upon every segment, yielding a 629 item feature vector for each five-second time slice.

Time prediction model

We used a two-dimensional convolutional neural network to regress. We arranged time along the first dimension and features along the second dimension. Since there is no meaningful relationship between the consecutive features (they are heterogeneous), we only convolve over the time dimension. The convolutional filters thus all have dimensions of the form $1 \times n$.

This problem does not admit a traditional loss function. If we used the most common loss function for regression problems, sum of squared errors, undue weight would be given to long-term errors. For example, if the network predicted 1 day when the actual time was 1 day plus 2 hours, the error would be 1600 times as large as the error if the network predicted 4 minutes when the actual time was 1 minute. However, the second is much more important for practical purposes. We therefore designed a loss function that prioritized correctness in the short-term and favored underestimates over overestimates (since an overestimate would make the user miss a potential seizure). We settled on the following function:

$$E(\hat{\theta}, \theta) = \log\left(1 + \left(1 - \frac{\hat{\theta}}{\theta}\right)^2\right) \log\left(e + \frac{\theta}{\hat{\theta}}\right) \quad (1)$$

There is no firm theoretical basis for the terms in this function; we chose them for the sake of creating a function with the correct properties. A principled attempt to find a utility function for this problem would likely improve practical accuracy, as well as allow for a better measurement metric.

Performance evaluation

Assessing performance on a regression problem of this kind poses several unique challenges. The same criteria that are useful in choosing a loss function are important here, too: it must properly weight small predictions higher and far predictions lower, as well as prefer underestimates to overestimates. Taking into account these factors is more important for an evaluation metric than a loss function, since evaluation metrics are less likely than loss functions to optimize for quirks of the method rather than the actual goal. However, since evaluation metrics, unlike loss functions, must be portable across widely varying implementations, it is critically important that they be as justifiable as possible. Since this is, to the best of our knowledge, the first regression attempt at this field, it would be prudent to refrain from setting a precedent for using a poorly-justified metric. Thus, despite its flaws in context, we used the standard coefficient of determination R^2 computed between the predicted and actual times.

Software

All our code is in Python version 3.7. We used `pyEDFlib` (Nahrstaedt, Lee-Messer, & van Beelen, 2009–) to read the CHB-MIT files. For feature calculation, we used NetworkX (Hagberg, Schult, & Swart, 2008), PyWavelets (Lee et al., 2019), NumPy (van der Walt, Colbert, & Varoquaux, 2011), and SciPy (Jones, Oliphant, Peterson, et al., 2001–). We used scikit-learn for our linear regression and MLP models (Pedregosa et al., 2011), `xgboost` for our gradient-boosted trees regressor (Chen & Guestrin, 2016), and PyTorch (Paszke et al., 2017) with the `skorch` wrapper (Benjamin Bossan & Tietz, 2017–) for the neural networks. Plots in this paper were generated with `matplotlib` (Hunter, 2007). Code for this project is available at https://github.com/matthewlw/seizure-forecasting.

Results

We first used more traditional machine learning methods as active controls. Ordinary least squares linear regression (depicted in figure 1), the best-performing of these models, had an R^2 of 0.128. The two other models in this category performed worse. The multilayer perceptron with hidden-layer sizes of 60 and 10 and all other parameters consistent with the default parameters of the `MLPClassifier` of `scikit-learn` 0.20.2 had $R^2 = -0.608$ (figure 2), and the gradient-boosted trees regressor (all parameters defaults in `xgboost` 0.81) had $R^2 = -0.499$ (figure 3).

Machine learning models had more mixed results. The worst performance came from the convolutional and LSTM MSE-loss networks. The convolutional MSE network, as shown in figure 4, had near-constant output of 2000 seconds; the LSTM MSE network is not depicted because it converged to predicting precisely zero for every input vector. However, the custom loss function improved performance in both cases, bringing the CNN and LSTM to R^2 values of 0.105 and 0.242, respectively.

Discussion

Overall performance of all models was poor. Visual inspection of the predicted-actual plots reveals that they are woefully inadequate for any practical use,

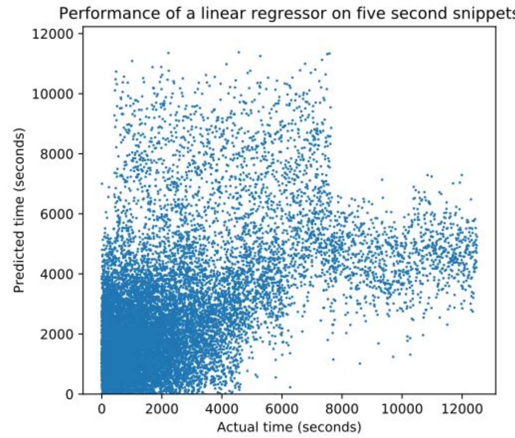

Figure 1: Linear regression predicted-actual plot

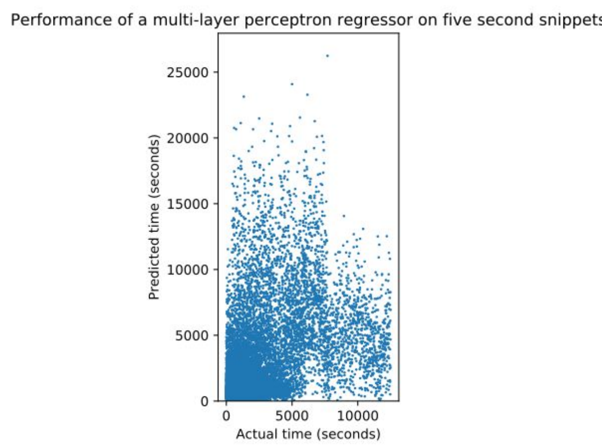

Figure 2: Multilayer perceptron predicted-actual plot

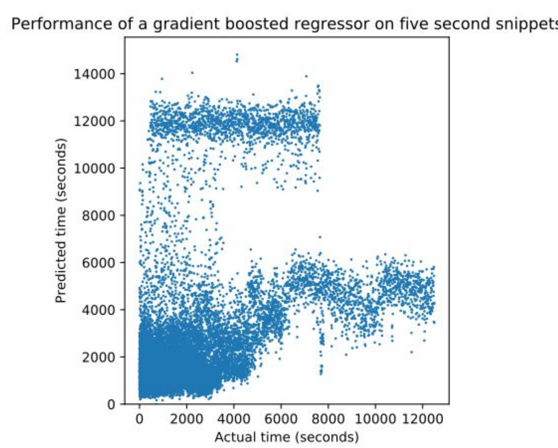

Figure 3: Gradient-boosted trees predicted-actual plot

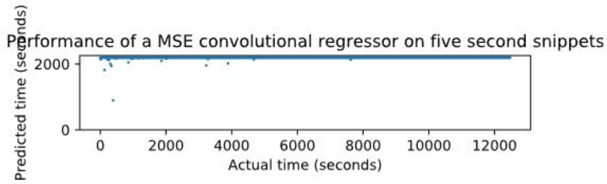

Figure 4: CNN with MSE loss function

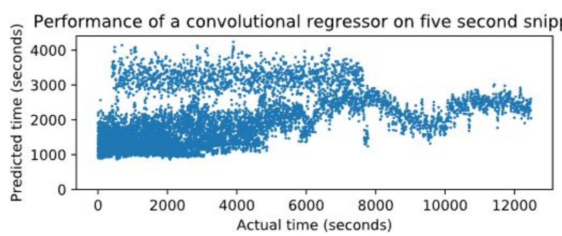

Figure 5: CNN with custom loss function

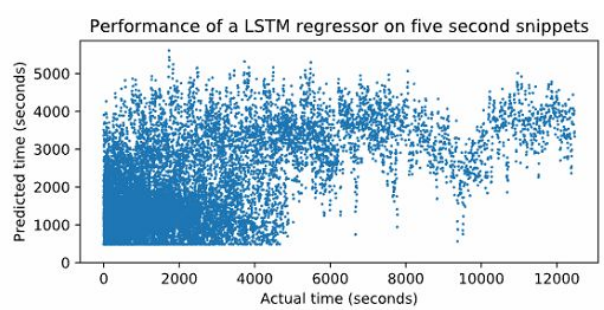

Figure 6: LSTM with custom loss function

especially practical use as a medical device. This goes for all models, not just the traditional ones. However, the work we have done does leave several avenues for further research.

The first potential modification is further tweaking the parameters of the networks. There are a large number of tunable hyperparameters on the neural network models we used, and we explored a very small portion of the parameter space due to time and computational power constraints. Stepping outside of pure convolutional or pure LSTM models could also improve performance by allowing for more specialized tweaks. In particular, the featureset we used would be amenable to a model that could somehow explicitly code for groups of features, since (for example) one would expect to see similar classes of pattern in cross-correlations across all channel pairs.

Zooming even further out, the regression model of seizure forecasting, rather than the classification model, certainly shows promise in a clinical setting. While our research cannot establish the viability of this route, for the reasons described above, a functioning time prediction model would be clinically useful. Further research could investigate theoretical avenues for determining the viability of a model such as this. We noticed that all our model performance plots underwent a shift around 7500 seconds; further research could establish whether this is a quirk of our dataset or a fundamental observation about the nature of epileptic seizures. If it is the latter case, this could establish regression models as a tool for investigating the dynamics of neural activity that causes seizures.

Lastly, we contribute a novel loss function specialized for this type of problem. As we observed significant performance gains even when using a metric that did not take into account the same special factors as our loss function, we expect that it may be useful as a heuristic that improves performance if not as a well-founded piece of model-building. We hope that the concerns we articulated about the utility of current loss functions might inspire future research into a more well-justified loss function that takes those concerns into account.

References

Acharya, U. R., Hagiwara, Y., & Adeli, H. (2018). Automated seizure prediction. *Epilepsy & Behavior*, *88*, 251–261. doi:10.1016/j.yebeh.2018.09.030

Benjamin Bossan, D. N. & Tietz, M. (2017–). Skorch: A scikit-learn compatible neural network library that wraps pytorch. [Online; accessed 2019-03-13]. Retrieved from https://github.com/dnouri/skorch

Camfield, P. & Camfield, C. (2010). Idiopathic generalized epilepsy with generalized tonic-clonic seizures (IGE-GTC): A population-based cohort with > 20 year follow up for medical and social outcome. *Epilepsy & Behavior*, *18*(1-2), 61–63. doi:10.1016/j.yebeh.2010.02.014

Chen, T. & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785–794). KDD '16. San Francisco, California, USA: ACM. doi:10.1145/2939672.2939785

Clercq, W. D., Lemmerling, P., Huffel, S. V., & Paesschen, W. V. (2003). Anticipation of epileptic seizures from standard EEG recordings. *The Lancet*, *361*(9361), 971. doi:10.1016/s0140-6736(03)12780-8

D'Alessandro, M., Vachtsevanos, G., Esteller, R., Echauz, J., Cranstoun, S., Worrell, G., … Litt, B. (2005). A multi-feature and multi-channel univariate selection process for seizure prediction. *Clinical Neurophysiology*, *116*(3), 506–516. doi:10.1016/j.clinph.2004.11.014

Fiest, K. M., Sauro, K. M., Wiebe, S., Patten, S. B., Kwon, C.-S., Dykeman, J., … Jetté, N. (2016). Prevalence and incidence of epilepsy. *Neurology*, *88*(3), 296–303. doi:10.1212/wnl.0000000000003509

Goldberger, A. L., Amaral, L. A. N., Glass, L., Hausdorff, J. M., Ivanov, P. C., Mark, R. G., … Stanley, H. E. (2000). PhysioBank, PhysioToolkit, and PhysioNet. *Circulation*, *101*(23). doi:10.1161/01.cir.101.23.e215

Hagberg, A. A., Schult, D. A., & Swart, P. J. (2008). Exploring network structure, dynamics, and function using networkx. In T. V. Gäel Varoquaux & J. Millman (Eds.), *Proceedings of the 7th python in science conference (scipy2008)* (pp. 11–15).

Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing In Science & Engineering*, *9*(3), 90–95. doi:10.1109/MCSE.2007.55

Jones, E., Oliphant, T., Peterson, P., et al. (2001–). SciPy: Open source scientific tools for Python. [Online; accessed 2019-03-13]. Retrieved from http://www.scipy.org/

Kuruvilla, A. & Flink, R. (2003). Intraoperative electrocorticography in epilepsy surgery: Useful or not? *Seizure*, *12*(8), 577–584. doi:10.1016/s1059-1311(03)00095-5

Lee, G. R., Gommers, R., Wasilewski, F., O'Leary, A., Nahrstaedt, H., Hurtado, D. M., … 0-tree. (2019). Pywavelets/pywt: Pywavelets v1.0.2. doi:10.5281/zenodo.2577784

Litt, B., Esteller, R., Echauz, J., D'Alessandro, M., Shor, R., Henry, T., … Vachtsevanos, G. (2001). Epileptic seizures may begin hours in advance of clinical onset. *Neuron*, *30*(1), 51–64. doi:10.1016/s0896-6273(01)00262-8

Mula, M. & Monaco, F. (2011). Ictal and periictal psychopathology. *Behavioural Neurology*, *24*(1), 21–25. doi:10.3233/BEN-2011-0314

Nahrstaedt, H., Lee-Messer, C., & van Beelen, T. (2009–). Pyedflib: Read and write edf files in python. [Online; accessed 2019-03-13]. Retrieved from https://github.com/holgern/pyedflib

Netoff, T., Park, Y., & Parhi, K. (2009). Seizure prediction using cost-sensitive support vector machine. In *2009 annual international conference of the IEEE engineering in medicine and biology society*. IEEE. doi:10.1109/iembs.2009.5333711

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., … Lerer, A. (2017). Automatic differentiation in pytorch. In *Nips-w*.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830. Retrieved from http://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html

Petrosian, A., Prokhorov, D., Homan, R., Dasheiff, R., & Wunsch, D. (2000). Recurrent neural network based prediction of epileptic seizures in intra- and extracranial EEG. *Neurocomputing*, *30*(1-4), 201–218. doi:10.1016/s0925-2312(99)00126-5

Quyen, M. L. V., Martinerie, J., Navarro, V., Boon, P., D'Havé, M., Adam, C., … Baulac, M. (2001). Anticipation of epileptic seizures from standard EEG recordings. *The Lancet*, *357*(9251), 183–188. doi:10.1016/s0140-6736(00)03591-1

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, *61*, 85–117. doi:10.1016/j.neunet.2014.09.003

Shoeb, A. (2009). *Application of machine learning to epileptic seizure onset detection and treatment* (PhD Thesis, Massachusetts Institute of Technology). Retrieved from http://hdl.handle.net/1721.1/54669

Stringer, J. (2017). Seizure initiation and termination. In *Reference module in neuroscience and biobehavioral psychology*. Elsevier. doi:10.1016/b978-0-12-809324-5.00222-4

Tsiouris, K. M., Pezoulas, V. C., Zervakis, M., Konitsiotis, S., Koutsouris, D. D., & Fotiadis, D. I. (2018). A long short-term memory deep learning network for the prediction of epileptic seizures using EEG signals. *Computers in Biology and Medicine*, *99*, 24–37. doi:10.1016/j.compbiomed.2018.05.019

van der Walt, S., Colbert, S. C., & Varoquaux, G. (2011). The NumPy array: A structure for efficient numerical computation. *Computing in Science & Engineering*, *13*(2), 22–30. doi:10.1109/mcse.2011.37