
Classification of EEG with Recurrent Neural Networks

Alex S. Greaves
Stanford University
agreaves@cs.stanford.edu

1 Abstract

3-D perception is a task that is growing in popularity in television and entertainment. Algorithms and innovations that mimic 3-D perception are of great importance to those in this industry, and as such they need a metric for how well a particular innovation is working. Electroencephalogram (EEG) recordings are an accurate and objective method of evaluating brain activity, and so the primary task is to use EEG recordings score different methods of mimicking 3-D perception. As a first step in doing this we must find the best features and methods to classify EEG recorded when participants are viewing regular 2D stimuli, and actual 3D stimuli. Hence, in this paper, we explore methods to address the following goal:

Can we use EEG signals to accurately classify whether someone is viewing a 2D or 3D image?

Using over 5,000 training examples, we investigated the effectiveness of multiple models in achieving this task, with an emphasis on neural networks, and in particular, Recurrent Neural Networks.

2 Introduction

In recent years, EEG classification has become an increasingly important problem in various fields. In the field of medicine, EEG detection could be incredibly promising for seizure or stroke detection in patients that are susceptible to such conditions, and a great deal of research has already been put into solving this problem. Other medical applications include manufacturing transportation devices for patients with limited motor abilities to control using simply their thoughts or extremely subtle facial movements. EEG would pick up on both of these and an efficient and accurate classifier could lead to the successful creation of such a device that would change the lives of patients with such a disability.

Yet other applications exist in the fields of psychology and neuroscience, where EEG classification can give insight into the inner workings of the human brain. For this project we will explore this particular application for the purpose of classifying human response to visual stimuli. In particular, the paradigm involves presenting three conditions of visual stimulus to the subject: (1) the same undoctored image presented to both eyes separately, (2) the same image but with binocular disparity between each eye to create a 3D effect, and (3) the image enhanced with an algorithm to increase its perceived depth presented to each eye with no binocular disparity.

The main goals for this project is to discriminate between EEG recorded during 2D vs 3D stimuli. From the classifier's features we can extract which regions of the brain and which time points during the recording were the most informative in distinguishing between these two classes. In synchrony this will tell us what regions of the brain respond strongly to 3D stimuli at what time after the initial onset of the image. This paper will focus on the methods and results of the discrimination task. A secondary goal, and the subject of future work, is to then use these spatio-temporal cues to compare the EEG recordings of the undoctored vs depth-enhanced images to gauge how well the algorithm does at evoking a 3D-like response in the human brain.

Conventional approaches to EEG classification primarily focus on classifying frequency information of record-

ings without deep learning, extracting this information using the Fourier or other transforms. However, recent literature has indicated that there is promise in using neural networks for EEG classification. In particular, due to the temporal nature of these recordings, a primary candidate for successful classification has been a Recurrent Neural Network, where at each time step the network retains information from previous time steps. This is the approach we will be taking for this project.

3 Approach

The data contains recordings from 12 human subjects, each of whom were recorded for approximately 20 minutes, corresponding to roughly 200 presentations of the stimulus for each class. Each stimulus was presented for 1.65 seconds, and at a sampling rate of 256Hz this corresponds to 420 time points per sample. At each time point the surface voltage is recorded from a 128 electrode set. With 2500 samples per class, the full size of the data per class is (2500 samples x 420 time points x 128 electrodes). The data is preprocessed to remove facial movements and detrended in order to reduce noise. In addition, due to the nature of visual response in the brain, frontal electrodes are excluded from the data set as they contain very little information regarding the stimulus. Hence, the final shape of the data per class before it gets to the classifier is (2500 samples x 420 time points x 88 electrodes).

At this point we have the option of doing one of two things with the data: (1) extract raw signal features, or (2) extract frequency features using the Fourier Transform. While the latter approach typically works best for EEG data, for this particular stimulus preliminary results indicate that features derived from frequency information yield very little information for discriminating between these two stimuli. Hence, throughout the rest of the paper we extract raw signal features. The process is as follows. We first define two hyper-parameters, the window length (W) and time step size (S). For step i of the $N = \lfloor \frac{420}{S} \rfloor$ steps (where $t_i = iS$), and for each electrode, we extract the mean signal strength of that window. Letting $V_j^{(k)}$ be the signal strength of electrode k at time-point j , the i -th features for electrode k ($F_i^{(k)}$) is:

$$F_i^{(k)} = \frac{1}{S} \sum_{j=t_i}^{t_{i+1}} V_j^{(k)}$$

In this way, we reduce the dimensionality of the data to be 2500 samples x N steps x 88 electrodes. Previously in the quarter, we have taken approaches to classifying this data without the aid of deep learning with moderately successful results. In general, EEG is difficult to classify due to its relatively low signal-to-noise ratio. This task is even more difficult for shorter recordings and in visual studies it is not uncommon to get below 60% classification accuracy even with the best classifiers. For this data set, prior to this project the state of the art was 67% cross-validation accuracy, which we can use as our success metric because the data set is balanced. In order to achieve this, we first perform PCA on each of the steps to reduce our dimensionality from 88 electrodes to M components (another hyper-parameter), and then flatten the last two dimensions to form the final pre-classifier data set with dimensionality 2500 samples x $N * M$ features. The best (non-deep-learning) classification method on this newly formed data set (using the best hyper-parameters) is Elastic Net regression, which is regularized linear regression that penalizes both the $L1$ and $L2$ norms.

In this project, we attempted three different models of neural networks to attempt to find a better classifier.

Model 1: Simple Multi-layer Perceptron

This is the simplest model we implemented, where the input was the same, flattened input as in our previous analysis. We tried both one and two hidden layer networks. For each hidden layer, we explored the use of both a fully connected and dropout layer, the latter of which was used to reduce over-fitting. A softmax layer was used to produce output probabilities.

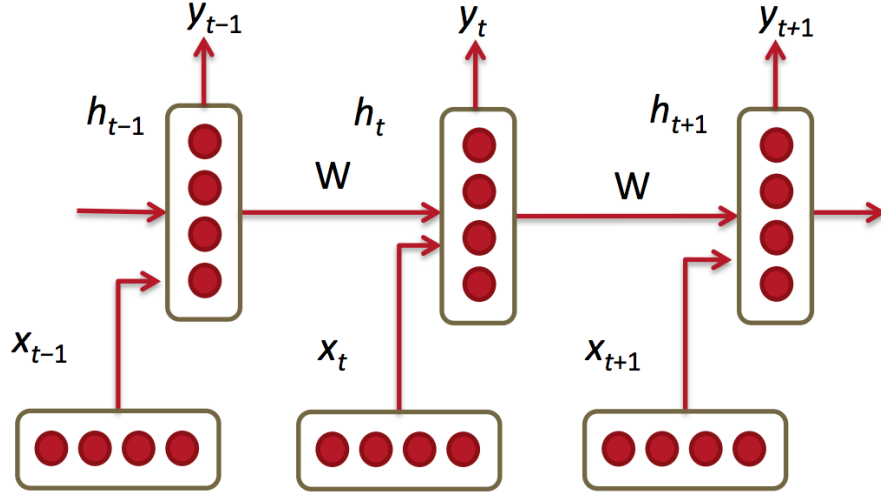


Figure 1: Basic structure of Elman Recurrent Neural Network

Model 2: Elman Recurrent Neural Network

The second model we implemented was a simple Elman RNN. As pictured in Figure 1, at each time step i , this network computes the subsequent hidden state from the previous one and the input at that time step. Thus, if $h^{(t-1)}$ is the previous hidden state and $x^{(t)}$ is the input at time-step i , the subsequent hidden state is given by

$$h^{(t)} = \sigma(Hh^{(t-1)} + Wx^{(t)})$$

where if h has m hidden units and x has dimension n , H and W are matrices with dimension $n \times n$ and $n \times m$, respectively. In addition, σ is used here to denote some non-linearity. We use a softmax layer on the last hidden layer to extract probabilities for classification.

Model 3: Time-dependent Elman Recurrent Neural Network

One fear we had was that in the Elman RNN model, we use the same two matrices at every time step. This is appropriate in Natural Language Processing because any word could appear at any point in the sequence, so in order to be generalizable we must have a consistent transformation for every time point. However, as seen in Figure 2, this is not the case for EEG data. Hence, a more powerful method which involves different matrices H and W for each time point could be more appropriate for this data. Thus, we define the Time-dependent Elman RNN to have the same structure as the network described above, but except that now the subsequent hidden state is computed as

$$h^{(t)} = \sigma(H^{(t)}h^{(t-1)} + W^{(t)}x^{(t)})$$

where H and W are now tensors with first dimension S and last two dimensions the same as above. $H^{(t)}$ and $W^{(t)}$ denote the t -th slice of tensors H and W , respectively (along the first dimension).

4 Experiment

We trained each of these models using batch gradient descent, using cross-entropy loss as our objective function. For this project we utilized Theano, and backpropagation of error was computed with it. For each model, we made an 80-10-10 split of the data set into training, dev, and test sets. We evaluated the success of each model by the classification accuracy of the dev set, and tuned the hyperparameters of the model accordingly. Such hyper-parameters

156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207

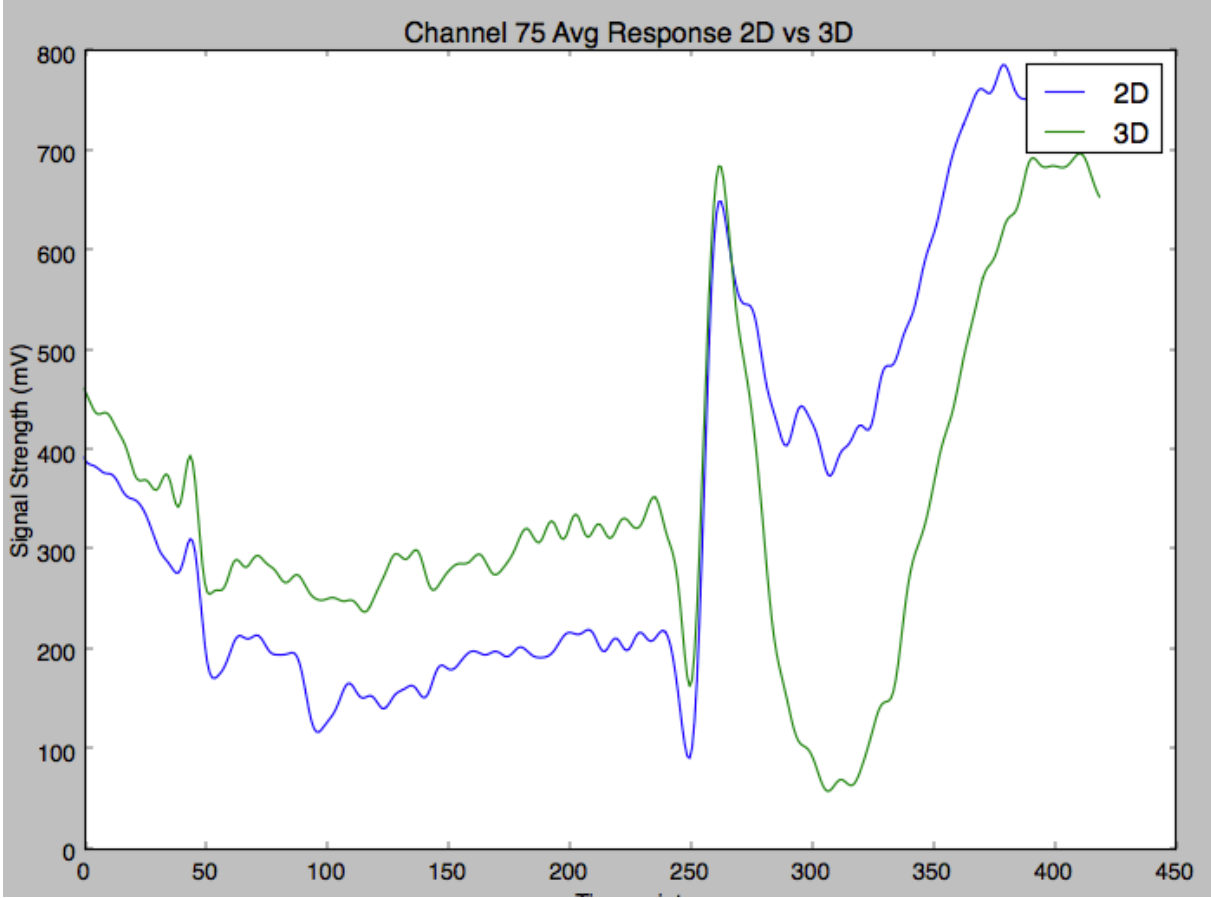


Figure 2: Single electrode raw EEG signal comparing 2D and 3D response

included window length and step size as mentioned above. Others included number of epochs to train, learning rate, dropout probability, regularization strength, and batch size.

The results of these experiments are summarized in Table 1.

Model	Training Acc	Dev Acc	Test Acc
Model 1 (one layer)	0.71	0.69	0.69
Model 1 (two layer)	0.74	0.71	0.72
Model 2	0.62	0.59	0.58
Model 3	0.68	0.62	0.6

Table 1: Best Results from Models 1-3

Interestingly, neither recurrent model proved to be more powerful than the non-deep-learning approach developed prior to this project. In particular, the time-dependent Elman RNN was prone to overfitting, likely due to the far greater number of parameters involved in the model, which was an order of magnitude greater than either the regular Elman RNN or the regular feed-forward network. However, the time-dependent model did prove to be more powerful than the regular Elman model, likely due to the nature of the data, as mentioned above.

208 Not surprisingly, the regular feed-forward network did manage to outperform the conventional analysis we de-
209 veloped prior to the project, increasing the accuracy from 0.67 to 0.72 with the best model. Both the one and two
210 hidden layer networks did better than the conventional analysis, with the two hidden layer network proving to be more
211 powerful.
212

213 5 Conclusion

214

215 From these results, we can conclude that it is not straightforward to apply RNNs to EEG data. While it is possible
216 that a more complex RNN would have done better at classification, it seems that a simple feed-forward network will
217 outperform a simple RNN. While EEG data is by nature sequences of vectors, as words are, the relationship from
218 one element in the sequence to the next must be different, to some impactful degree, in EEG from Natural Language
219 Processing. Still, both RNN models managed to get fairly above chance results, and so future work should involve
220 applying more complex recurrent models to EEG data.
221

222 References

223 C. Anderson, E. Forney, D. Hains, and A. Natarajan, "Reliable identification of mental tasks using time-embedded EEG and
224 sequential evidence accumulation," *Journal of Neural Engineering*, vol. 8, no. 2, p. 025023, 2011.

225 D. Coyle, G. Prasad, and T. McGinnity, "Extracting features for a brain-computer interface b self-organizing fuzzy neural network-
226 based time series prediction," in *26th Annual International Conference of The IEEE Engineering in Medicine and Biology Society*.
227 IEEE, 2006, pp. 2183-2186

228 E. Forney, "Electroencephalogram classification by forecasting with recurrent neural networks," Master's thesis, Department of
229 Computer Science, Colorado State University, Fort Collins, CO, 2011.
230

231 [2] Bower, J.M. & Beeman, D. (1995) *The Book of GENESIS: Exploring Realistic Neural Models with the GEneral NEural SIMu-*
232 *lation System*. New York: TELOS/Springer-Verlag.
233