Classifying Music from Thought

Akshay Sadanandan [1] **Arjun Rao** [1] **Sayali Shelke** [1] **Varun Eranki** [1]
{ asadanandan, arao, sbshelke, veranki} @wpi.edu

Abstract

Deep Learning methods have shown to produce state-of-the-art results when applied to problems in various fields of study. In this paper, we investigate the lesser explored intersection of deep learning and cognitive neuroscience with the task of classifying music from EEG recordings taken from the OpenMIIR dataset. We implement several novel techniques and obtain better results than previous state-of-the-art model for the classification task.

1. Introduction

Music is now a part of our everyday lives. People create tunes that are are unwinding or inspiring and the initial step to this is envisioning the tune in the head to locate the correct rhythm. The downside of the present music recovery frameworks is that it requires the individual to emulate the tune through singing, humming or by recollecting a bit of the verses. Music imagery looks profoundly feasible through means of brain computer interfaces by the utilization of EEG signals. (Schaefer, 2011) says "music is especially suitable to use here as (externally or internally generated) stimulus material, since it unfolds over time, and EEG is especially precise in measuring the timing of a response." We can utilize EEG signals to recover transient qualities, for example, musical rhythm and tempo of every stimuli. Deep learning has become a mainstream strategy as of late, developing models that are state of the art in the fields of speech recognition, computer vision, natural language processing,etc.. As amazing as this sounds, there has been less utilization of deep learning in the field of brain computer interfaces. Over the last couple of years, there certainly has been some progress in the field of BCI and deep learning, one example being Elon Musk opening neuralink to enable better use of machine learning in this field. There are certain challenges with regards to applying BCI with deep learning. Some of them are the noise that is present while gathering the data, absence of abundant data, high dimensionality of data and and comprehend the significance of whatever information has been procured. EEG accounts for good temporal resolution which we think is going to be an important factor when it comes to reading the music imagery information.

We worked on OpenMIIR dataset for our classification problem and have explored how deep learning can take us forward in this field. Music imagery information retrieval (MIIR) is an emerging field of research at the intersection of cognitive neuroscience and music information retrieval. The goal is to identify music pieces from brain signals recorded while they were either perceived or imagined. MIIR systems may one day be able to recognize a song just as we think of it. Being able to reliably distinguish between just two different imagined music pieces would already allow us to build a music-based brain computer interface (BCI) for patients that are only able to communicate by modulating their own neural activity. To perform an in-depth analysis of the signal,we performed deep learning techniques such as CNN's, LSTM's and VAE's and novel techniques like Recurrence Plots to compensate for the lack of data since this involves starting at a point close to the ball-park range of the model.

2. Dataset

OpenMIIR is a public domain dataset of electroencephalography (EEG) recordings taken during music perception and imagination.The study recorded signals at a frequency of 512 Hz and save in FIF file format. This data was acquired during an ongoing study that so far comprised 10 subjects listening to and imagining 12 short music fragments – each 7s-16s long – taken from well-known pieces. These stimuli were selected from different genres and systematically span several musical dimensions such as meter, tempo and the presence of lyrics. There are 4 conditions for each trial and for each subject. That brings the total to 4 * 5 * 12 = 240 events for each subject and there are 9 subjects in total. Pre-processing was done with the help of the MNE-python toolkit (Gramfort et al., 2013), Pylearn2 to remove the bad channels and remove the Electroculography signals. The most common channels used in EEG studies is between 0.5-30 Hz and a band pass filter was used to extract signal content in this range. All the recording were taken with the help of a BioSemi Active-Two system using 64+2 EEG channels at 512 Hz.

The conditions were presented in the following ways:

1. Stimulus perception with cue clicks
2. Stimulus imagination with cue clicks
3. Stimulus imagination without cue clicks
4. Stimulus imagination without cue clicks, with additional feedback from participants after each trial

ID	Name	Length
1	Chim Chim Cheree (lyrics)	13.3s
2	Take Me Out to the Ballgame (lyrics)	7.7s
3	Jingle Bells (lyrics)	9.7s
4	Mary Had a Little Lamb (lyrics)	11.6s
11	Chim Chim Cheree	13.5s
12	Take Me Out to the Ballgame	7.7s
13	Jingle Bells	9.0s
14	Mary Had a Little Lamb	12.2s
21	Emperor Waltz	8.3s
22	Hedwig's Theme (Harry Potter)	16.0s
23	Imperial March (Star Wars Theme)	9.2s
24	Eine Kleine Nachtmusik	6.9s

Table 1. Information about the stimuli

3. Related Work

Many works have been conducted to improve EEG classification accuracy and a great variety of hand-designed features have been proposed. With the rapid development of deep learning in recent years, many excellent networks have been presented by researchers. In recent years, many public works have discussed deep learning applications in bioinformatics research.(Stober et al., 2015) made the first open dataset of eeg recordings during music listening and imagination which was meant to allow MIR researchers, who usually don't have brain recording devices, to access brain data.

EEG data are generally only available in small quantities, they are high dimensional with a poor signal-to-noise ratio, and there is considerable variability between individual subjects and recording sessions. (Stober, 2016) specifically address challenges for feature learning of such data. Cross-trial encoding forces auto-encoders to focus on features that are stable across trials. Similarity constraint encoders learn features that allow to distinguish between classes by demanding that two trials from the same class are more similar to each other than to trials from other classes.

The paper (Stober et al., 2016) uses tempograms to identify beats by tempo. There were no methods to classify which songs were playing but only methods to identify the tempo of a signal. We learned that beats was a good identifier and have tried experiments based on that.

Further applications of deep learning techniques on EEG data comprise (Stober, 2017) pre-training technique for learning discriminative features from electroencephalography (EEG) recordings using similarity constraints encoding.

(Li & Mandt, 2018) is a paper that helps generate time invariant and time variant features as two separate entities. By doing this, they are able to manipulate either feature. Having control over these features allowed for sequences to be modified by these two entities while still maintaining the other entity. Disentangled EEG

Recurrence plots (RP) were introduced as a visualization tool to measure the time constancy of dynamical systems. Natural processes can have distinct recurrent behaviors like periodicities (as seasonal cycles) or irregular cyclicities(Hatami et al., 2018). Also, biological systems possess behavioral patterns and activity dynamics. Such irregular behavior, despite of all the efforts, were not possible to get detected by a human eye or can be missed while extracting features. As CNNs are capable of automatically extracting features at different levels of abstraction we use this technique (Hatami et al., 2018) to convert our EEG signals to RP images and pass it to the CNN.

4. Proposed Methods

4.1. Convolutional Neural Netowrk

CNN is a standard method to apply in case of any deep learning method. It gives us a good baseline for any architecture and so the first method we tried is a CNN.

4.2. Recurrence Plots and CNN

A recurrence is the time when a trajectory returns to the location that it has visited before. Recurrence Plots depicts the collection of pairs of time at which the trajectory is at same place i.e the set of **(i,j)** with $\vec{x}(i) = \vec{x}(j)$.

A joint recurrence plot is an extension of recurrence plots for multivariate time series. A recurrence plot is built for each feature of the multivariate time series, then the set of recurrence plots is reduced to one single recurrence plot using the Hadamard product.

Here, we used Joint recurrence plots to convert the EEG signals from 64 channels to RP images Fig. 1 and pass it to the Convolutional Neural Network.

4.3. Bi-Directional LSTM

However, standard LSTMs (by which we mean LSTMs containing hidden layers of recurrently connected neurons) have limitations of their own. Firstly, since they process inputs in temporal order, their outputs tend to be mostly

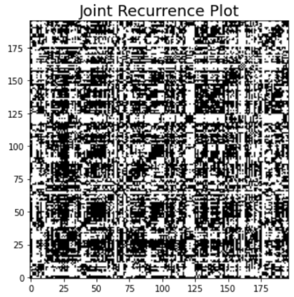

Figure 1. Represents a joint recurrence plot of a EEG signal from 64 channels

based on the previous context. There are ways to introduce future contexts, such as adding a delay between the outputs and the targets; but these do not usually make full use of backward dependencies. An elegant solution to this is provided by bidirectional networks. The typical architecture of a Bi-directional network is shown in fig.2

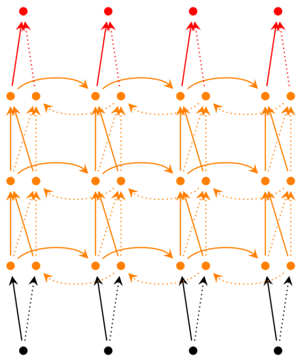

Figure 2. The architecture of a Bi-Directional network.

4.4. LSTM Variational Autoencoder CNN

Since the LSTM model worked well we tried using a LSTM Variational Autoencoder(LSTM VAE) to represent a compressed version of sequence data which in our case is the music stimuli. It has a good chance of working if the brain signal synchronizes with the beat of the music forming another sequence. We are trying to see if these brain signal form a sequence while listening to a song and check how the model performs on the output of this model.

4.5. Disentangled Classification

The output of the LSTM Variational Autoencoder performed better than expected since reconstruction of signals is hard and we achieved a good accuracy on the model. Rather than use the output of the VAE directly for classification, we decided to use the representation for classification. We replace the decoder with a classifier since we want to perform classification. According to the paper, we can separate time variant from time invariant features and we decided to implement this as EEG data contains a lot of additional noise which are time variant like muscle movement, eye blinks etc.. (Stober et al., 2016) shows that beats perform well when it comes to classifcation. We are hoping to find distinct features that are independent of time, one being beat event related potentials(Beat ERP), in the EEG data so that our accuracy is good. ERPs are elicited by specific stimuli presented to the participant.

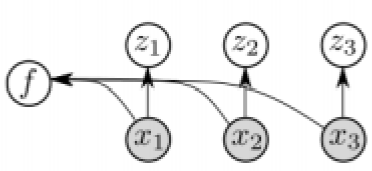

Figure 3. f and z represent the time invariant and time variant features respectively.

5. Experiments

We did outer cross validation with 9 subjects in total, we chose 8 subjects for training and 1 subject as a test subject. We considered two approaches - within subject evaluation and cross subject evaluation to understand the complexity of these approaches and found both these approaches give different results.

5.1. Within-subject Evaluation

5.1.1. Bidirectional LSTM

To understand the complexity of data and to know if data belonging to a subject can be correctly classified if trained on other data from the same subject, we performed a classification task by splitting 3 trials of all subjects to form the train set, 1 trial for validation and 1 trial for testing. The model consisted of 2 Bidirectional LSTM layers with 64 units each, followed by an LSTM layer of 100 units, followed by fully connected layers with 'tanh' and 'softmax' activations respectively as shown in Fig 4. We achieved exceptional results in this area and the model could correctly classify almost the entire test set. We could then understand that it is easy for Deep-Learning models to analyse how a person thinks, learn the patterns of a specific subject's EEG recordings when trained on the same subject's data. By doing this, these models can easily recognize these patterns learned during training and identify the music bits from thoughts.

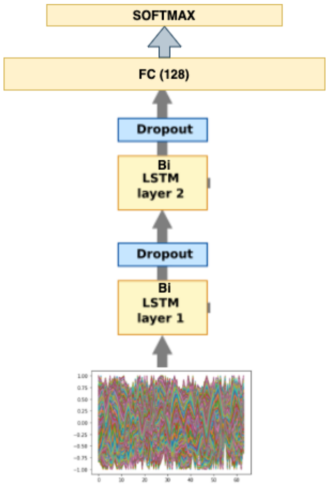

Figure 4. Bi-directinal LSTM pipeline

5.1.2. Recurrence plots and CNN

As explained in Section 3.2, Joint Recurrence plots are used to convert the 64 channel EEG signals into images. Down-sampling of the time-steps was done as it was huge(3518, shortest length) Polyphase filter was used for down-sampling the signal. The resulting RP image was of size 196 x 196. This image was then passed to a CNN with 32 units output layer, Max pooling layer and ReLU activation, followed by 16 units output layer, Max pooling layer and ReLU activation. Then a dense layer of 98 units was used followed by 12 units softmax layer to classify the images as shown in Fig.5 .

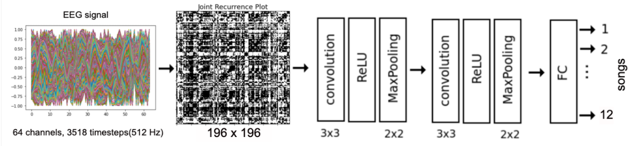

Figure 5. EEG signal is down-sampled and converted to Recurrence plot images and then passed to the CNN. This architecture consists of 2-convolution(32,16), 2-pooling and one FC layer of 98 units and a softmax layer of 12 units.

5.2. Cross-subject Evaluation

5.2.1. Convolutional Neural Netowrk

We use a convolutional neural netowrk with a softmax layer at the end to perform classification. The model consits of 32 layers followed by 64 layers after which we perform max pooling and dropout of 0.25. The layers are then passed to fully connected layers. We set the input length to 4096 to minimize padding and maintaining square length.

5.2.2. Recurrence plots and CNN

Cross subject evaluation is done on the same pipeline explained in Section 5.1.2 and Fig 5

5.2.3. Bidirectional LSTM

We also used a Bidirectional LSTM with two layers of 64 units each and a dropout of 0.7. We increased the input length to 6000 dimensions to account for the loss of information. High dropout was used to avoid overfitting such high dimensionality of data. These layers were followed by fully connected layers with 'tanh' and 'softmax' activations respectively. This model was trained for 15 epochs and the test accuracy was measured after every epoch.

5.2.4. LSTM Variational Autoencoder

Code references: LSTM VAE We set the input length to 4096 to minimize padding and maintaining square length. The model was trained for 600 epochs. The variational autoencoder was chosen for this task with 100 dimensions in the latent dimension space and 32 dimensions in the intermediate space. Finally, once the LSTM VAE is trained, we use a convolutional neural netowrk with a softmax layer at the end to perform classification.

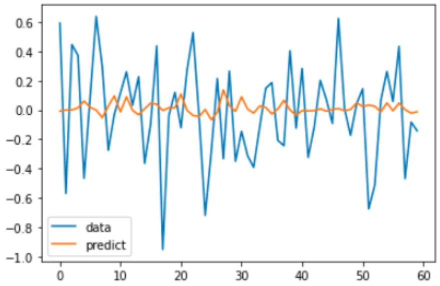

Figure 6. Represents the output of the LSTM VAE

5.2.5. Disentangled Classification

The input data is passed through a convolution layer which is connected to a fully connected layer. This reduces the input length of 3518 to the sequence length mentioned. This is then passed to a bidirectional LSTM at each time step. The paper (Li & Mandt, 2018) concatenates the features from the last time step of the forward unit with the first time step of the backward unit followed by a linear transformation in order to compute the mean and variance of the time invariant features. The time-variant features are also assumed to be gaussian.We use cross entropy loss as the loss function.

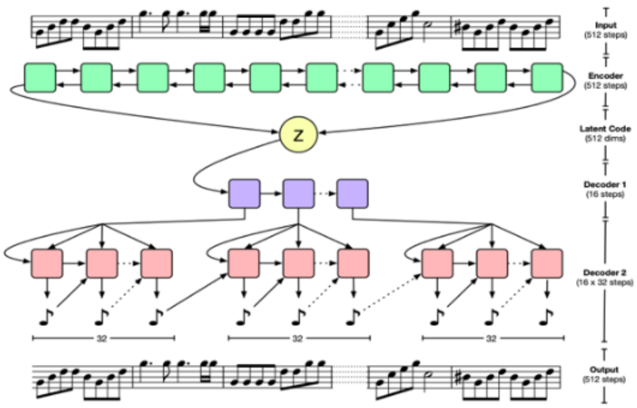

Figure 7. Represents the model of the LSTM VAE

6. Results

6.1. Within-subject Evaluation Results

Method	Condition	Test Accuracy
Bidirectional LSTM	All	99.53
Recurrence Plots + CNN	All	100

Table 1. Within subject test results

6.2. Cross-subject Evaluation Results

Method	Condition	Test Accuracy
SCE, NN (Baseline)	Perception	27.8
CNN	Perception	30.33
Bidirectional LSTM	Perception	31.6
LSTM-VAE-CNN	Perception	26.67
Disentangled VAE	Perception	30.0
Recurrence Plots + CNN	All	23.14
Bidirectional LSTM	All	23.3

Table 2. Cross-subject test results

7. Discussion

Trying to determine which piece of music perceived or imagined by someone using brain signals is a challenging task. Attempting to do so using a small dataset makes the problem even harder. In this paper, we trained classifiers for a 12-class problem on the combined data from 9 subjects with an input dimensionality of 28,160 (at 64 Hz) or 225,280 (at 512 Hz).

All classifiers described in Table 2 performed well and seven exceeded the performance of the previous state-of-the-art model (Stober, 2016) , which had a test accuracy of 27.8%. However, the classifiers did not perform well in the ”within subjects” experiment. Due to the nature of this experiment, the amount of data available to train the model was very low, causing the flexible classification models to over-fit the data. These models cannot be used for the classification task but led us to explore the ”cross-subject” experiment, which led to good results.

The main limitation here, is the scarcity of high quality EEG data. As with most Deep Learning problems, we require a large amount of data to get high classification accuracy. However, there is no such abundance of data when it comes to EEG research.

8. Conclusions and Future Work

We obtained promising results given the size and complexity of the data. We considered the gaussian space to be Gaussian in case of the variational autoencoder, and feel that it is worth attempting the same study assuming it is Laplacian. Given the amount of noise present in the data, it is difficult for a model to go around that and find which part of the signal is most important.

Most deep learning models require a lot of data for good results, the architectures they have provided performed pretty decent for very little data. Data acquisition for BCI is a tedious job and more architectures that perform well with few data points should be encouraged. Of course, a more grand experiment with a lot of participants is going to be a huge task but it will certainly be helpful for the field, especially with adept deep learning techniques around.

Having performed better than chance(8.3) on a consistent basis, we can say that there is most certainly a pattern in the brain signals when listening to music.

References

Gramfort, Alexandre, Luessi, Martin, Larson, Eric, Engemann, Denis, Strohmeier, Daniel, Brodbeck, Christian, Goj, Roman, Jas, Mainak, Brooks, Teon, Parkkonen, Lauri, and Hämäläinen, Matti. Meg and eeg data analysis with mne-python. *Frontiers in Neuroscience*, 7: 267, 2013. ISSN 1662-453X. doi: 10.3389/fnins.2013.00267. URL https://www.frontiersin.org/article/10.3389/fnins.2013.00267.

Hatami, Nima, Gavet, Yann, and Debayle, Johan. Classification of time-series images using deep convolutional neural networks. In *Tenth International Conference on Machine Vision (ICMV 2017)*, volume 10696, pp. 106960Y. International Society for Optics and Photonics, 2018.

Li, Yingzhen and Mandt, Stephan. Disentangled sequential autoencoder, 2018.

Schaefer, Rebecca. Measuring the mind's ear: Eeg of music imagery. *PhD Thesis*, 09 2011.

Stober, S. Learning discriminative features from electroencephalography recordings by encoding similarity constraints. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6175–6179, 2017.

Stober, Sebastian. Deep feature learning from eeg recordings. arXiv:1511.04306, 2016.

Stober, Sebastian, Sternin, Avital, Owen, Adrian M, and Grahn, Jessica A. Towards music imagery information retrieval: Introducing the openmiir dataset of eeg recordings from music perception and imagination. In *ISMIR*, pp. 763–769, 2015.

Stober, Sebastian, Prätzlich, Thomas, and Müller, Meinard. Brain beats: Brain beats: Tempo extraction from eeg data. pp. 276–282, 01 2016.