Detecting Alzheimer's Disease Using Convolutional LSTM Neural Networks

Jacob Diaz (jld349@cornell.edu)
Joseph Whelan (jfw225@cornell.edu)

CS 4701: Practicum in Artificial Intelligence, Fall 2022, Cornell University

1. Introduction

Alzheimer's disease is a progressive neurodegenerative disorder that is characterized by a decline in cognitive abilities and memory. Early detection and diagnosis of Alzheimer's disease is crucial for patients to receive timely treatment and support. In in our project we explore the use of a convolutional LSTM neural network to predict Alzheimer's disease (AD) using functional magnetic resonance imaging (FMRI) data.

FMRI is a neuroimaging technique that allows for the non-invasive study of brain activity. Previous studies have shown that FMRI data can be used to differentiate between individuals with Alzheimer's disease and healthy controls. Additionally, Convolutional LSTM neural networks have been shown to be effective in modeling spatial-temporal data [6], making them a promising approach for predicting Alzheimer's disease using FMRI data.

In this paper, we present our methodology and results in using a convolutional LSTM neural network to predict Alzheimer's disease from FMRI data. We also discuss the potential directions for future work.

2. Background on Alzheimer's Disease

Alzheimer's disease is a progressive neurodegenerative disorder that affects memory, thinking, and behavior. The disease typically affects individuals over the age of 60 and the prevalence increases with age. The exact cause of Alzheimer's disease is not fully understood, but it is believed to be a combination of genetic and environmental factors.

The symptoms of Alzheimer's disease typically begin with mild memory loss and difficulty with problem-solving and decision-making. Over time, the symptoms become more severe, leading to a decline in cognitive abilities and the ability to perform daily activities. There is currently no cure for Alzheimer's disease, but there are treatments available to manage the symptoms and slow the progression of the disease.

Alzheimer's disease is typically diagnosed by a physician or neurologist. The diagnosis is based on a comprehensive evaluation that includes a thorough medical examination as well as cognitive testing. In addition to these tests, doctors may also use

neuroimaging techniques, such as magnetic resonance imaging (MRI) or computed tomography (CT) scans, to assess brain structure and detect any abnormalities that may be indicative of Alzheimer's disease.

3. Current Literature on Using AI for Alzheimer's Disease Classification

There has been a growing interest in using AI techniques, such as machine learning and deep learning, for the early detection and classification of Alzheimer's disease. Previous studies have explored the use of various machine learning algorithms, including support vector machines, decision trees, and random forests, to predict Alzheimer's disease from various types of data, such as demographic information, cognitive test scores, and neuroimaging data.

Deep learning methods, such as convolutional neural networks (CNNs), have also been applied to the prediction of Alzheimer's disease. For example, Liang et al. [5] use a 2d CNN on the cross sections of a 3d MRI scan to extract spatial features which they pass into a fully connected layer to classify a patient as having AD or not. However, this approach limits the convolutions to two dimensions. Folego et al. [1] uses a 3d CNN to fully capture 3d features scan to diagnose AD.

Some studies have also made use of recurrent neural networks (RNNs) to capture the temporal features of an fMRI scan. Li et al. [3] tracks the activation levels of 90 regions of interest in the 3d scan over time. These 90 regions of interest can be represented as a vector which was then passed sequentially into in LSTM, a type of RNN, to encode information about the sequence. This sequence encoding was then passed into a fully connected layer for classification of AD.

The majority of literature we found focuses heavily on the two previous deep learning methods. These techniques can be effective, however, they utilize spatial and temporal data independently. We believed that it would make sense for a model to consider the spatial and temporal aspect of the fmri scan simultaneously. There are far fewer examples of this combined approach. One paper we did find, by Li et al. [4], uses a 3d convolutional layer extract the spatial embeddings of the brain at each time step of the fMRI scan and then feed these embeddings sequentially into an LSTM. Since brains have a complex structure are not static this architecture seemed to be the most promising and was what we decided to reimplement for our project.

4. FMRI Data: Source and Structure

The FMRI data used in this study was obtained from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database. The ADNI is a large, multi-center study that aims to identify biomarkers of Alzheimer's disease using various neuroimaging techniques, including FMRI. The ADNI dataset includes a wide range of imaging and cognitive data

from individuals with Alzheimer's disease, mild cognitive impairment, and healthy controls.
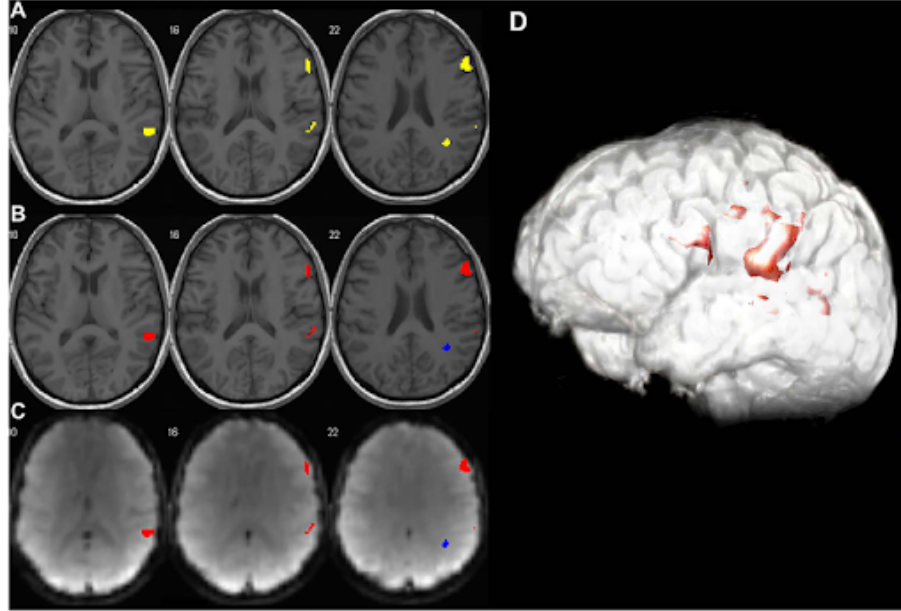

Figure 1. Example Brain Scan – A, B, and C are 2D cross-sections of the brain. D is the full 3D composition of the brain which is constituted by several 2D slices. Each A, B, C, and D are from the dataset.

[2]

The FMRI data in the ADNI dataset was collected using a 3T MRI scanner and a standard blood-oxygen-level-dependent (BOLD) contrast. The data consists of FMRI time series data, which contains information about the spatial and temporal patterns of brain activity. More specifically, each FMRI sample is a four dimensional object. At the most fundamental level, the last two dimensions represent a 2D grayscale cross-sectional image of the brain as seen in Figure 1 from patients A, B, and C. The number of 2D cross sections that can be stacked on top of each other is the second dimension. This stack composes a full 3D image of the brain as seen in Figure 1 from Patient D. The first dimension is a series of these 3D brain scans over time. So overall, each of our samples contains 140 3D brain scans over time that are each constituted by stacking 48 2D cross sections of the brain which are each 64x64 pixels in size. More rigidly, each sample is a 4D tensor of size $[140, 48, 64, 64]$.

To make our data learnable by our model (this problem is talked about in depth in Section 6), we had to preform preprocessing. We started by normalizing our data by subtracting the mean and dividing by the standard deviation of all entries across all fMRi scans. We also applies a 6mm * 6mm * 6mm Gaussian blur to every scan as done in prior work [4].

5. Methodology

While our main focus of the project was to train a convolutional LSTM classifier, we also implemented a 3d CNN for classification. This simple model takes in the final 3d scan from the 4d fmri and outputs a probability that this person has AD. The fundamental reason for making this model was to overfit to the training set to (1) verify that our data was in a form that would allow our model to learn, and (2) verify that the convolutional layer model has a sufficient representational capacity. Therefore, since we were only looking to overfit our CNN, we did not attempt tune the hyperparamaters of this model.

The model layers along with the corresponding tensor output shape at each layer are shown below (the number before the @ refers to the number of channels):

- input shape: (48, 64, 64)
- convolution-1: 8 @ (42,58,58)
- relu: 8 @ (42,58,58)
- maxpool: 8 @ (21,29,29)
- convolution-2: 16 @ (17,25,25)
- relu: 16 @ (17,25,25)
- maxpool: 16 @ (8,12,12)
- convolution-3: 32 @ (6,10,10)
- relu: 32 @ (6,10,10)
- maxpool: 32 @ (3,5,5)
- flatten: (2400)
- fully connected layer: (1)
- sigmoid: (1)

The convolutional LSTM we built took in a tensor of shape (140, 48, 64, 64) and applied the following convolutional layer to each of the 140 3d frames in the fMRI scan:

- input shape: (48, 64, 64)
- convolution-1: 8 @ (46, 62, 62)
- relu: 8 @ (46, 62, 62)
- maxpool: 8 @ (23, 31, 31)
- convolution-2: 16 @ (21, 29, 29)
- relu: 16 @ (21, 29, 29)
- maxpool: 16 @ (10, 14, 14)
- convolution-3: 32 @ (8, 12, 12)
- relu: 32 @ (8, 12, 12)
- convolution-4: 32 @ (6, 10, 10)
- relu: 32 @ (6, 10, 10)
- maxpool: 32 @ (3, 5, 5)

- convolution-5: 64 @ (1, 3, 3)
- relu: 64 @ (1, 3, 3)
- flatten: (576)
- dropout: (576)
- fully connected layer: (192)

Then each vector of length 192 was passed sequentially into an LSTM whose final hidden layer was passed into a fully connected layer that output the probability that the person in the scan had Alzheimer's.

We trained both models for 100 epochs using the Adam optimizer with a learning rate = 0.0001 and for our loss function used the cross entropy loss.

6. Challenges

There were several challenges and limitations that we encountered in the course of our work. Our first goal was to train a model to overfit the data, which would allow us to evaluate the model's performance on a held-out test set. In addition, accomplishing this goal would confirm that our model architecture is actually able to learn our FMRI data.

We started with the convolutional LSTM model that was described in the original paper. However, we were unable to train the model to overfit the data, which was an extremely confusing problem. The train-test split that we chose was very simple, but we continued to see high training error. To debug this issue, we continued to decrease the size of the training set until there were only four samples: 2 with a label of 0 (cognitively normal) and 2 with a label of 1 (Alzheimer's disease). We were still unable to train the model to overfit the data, which was extremely frustrating. And in fact, we observed that the accuracy was constantly 0.5 due to the fact that the model would always predict every sample as either a 1 or a 0.

We further iterated over the problem by regressing the model architecture. More specifically, we removed the dropout, normalization, and other layers that introduced regularization in addition to the LSTM layer. From this, we were left with a simple convolutional neural network whose sole purpose was to overfit the data. Even with this simple model, we were still unable to improve the training accuracy.

We then decided to try a different approach and asked ourselves: what if the problem is not with the model, but with the complexity of the data? Taking this idea in stride, we began by stripping the fourth dimension of the data (time) and only using the spatial information. But even with this, we still could not train the model to overfit the data. We then generated a bunch of really simple fake data and tried to train the model to overfit that. To our excitement, this worked! We were able to train the model to overfit the data, which confirmed that the problem was not with the model, but with the complexity of the

data.

At this point, we went back and re-read the prior literature and focused our attention toward preprocessing. We noticed that each of the prior research papers preprocessed their data, but they did not put much emphasis on this step. Our thought was that preprocessing would improve the test accuracy of our model, but we didn't anticipate that it would have such a significant impact on the training accuracy. After preprocessing the data, we were able to train the model to overfit, getting perfect train accuracy (see Section 4 for specific preprocessing steps). From this point, we incrementally added back the pieces of the model that we had removed in order to debug the problem until we were left with the original convolutional LSTM architecture.

7. Results

In this section, we present the results of our study on using a convolutional LSTM neural network to predict Alzheimer's disease from FMRI data. We used the FMRI data from the ADNI dataset, which included a total of 116 individuals with Alzheimer's disease and 186 individuals that were cognitively normal.

The convolutional LSTM model was trained on a random subset containing roughly 80% of the ADNI dataset, and it was then tested on the remaining data. The model was trained for for several hyperparameter configurations as seen in table 1.

FWHM Blur and Dropout	Weight Decay: 0	Weight Decay: 0.0001
FWHM Blur: 0mm Dropout: 0.2	Train Accuracy: 0.593 Test Accuracy: 0.714 ROC AUC: 0.5	Train Accuracy: 1.0 Test Accuracy: 0.661 ROC AUC: 0.637
FWHM Blur: 0mm Dropout: 0.5	Train Accuracy: 0.593 Test Accuracy: 0.714 ROC AUC: 0.5	Train Accuracy: 1.0 Test Accuracy: 0.607 ROC AUC: 0.695
FWHM Blur: 6mm Dropout: 0.2	Train Accuracy: 0.593 Test Accuracy: 0.714 ROC AUC: 0.531	Train Accuracy: 0.902 Test Accuracy: 0.678 ROC AUC: 0.516
FWHM Blur: 6mm Dropout: 0.5	Train Accuracy: 0.593 Test Accuracy: 0.714 ROC AUC: 0.5	Train Accuracy: 1.0 Test Accuracy: 0.661 ROC AUC: 0.656

Table 1. Model results when varying the FWHM blur, dropout, and weight decay.

While we were able to achieve perfect train accuracy we did not find our model to perform very well on the test set for any hyperparameter configuration. This was disappointing as other papers we had read were able to achieve 80% - 90% test accuracy.

8. Discussion and Future Work

The low test accuracy we achieved on a dataset with a few hundred samples would not have seemed surprising if it had not been for prior papers. Deep Neural networks perform best when trained on much larger datasets. However, there is research that shows high test accuracy on the fMRI scans from ADNI.

At this point in the project we believe that the best next steps would be to contact the corresponding authors of research papers we read. In the future we are likely going to do this and see if they could give us the exact subset of fMRI scans from ADNI they used as well as their code. This would allow us to verify if we are just using bad dataset, or if our model implementation is wrong, or both.

Another interesting direction for future work would be to explore the use of other types of data, in addition to FMRI data, for the prediction of Alzheimer's disease. For example, combining FMRI data with demographic information or cognitive test scores may improve the accuracy of the model, and it may provide additional insights into the underlying mechanisms of Alzheimer's disease.

Finally, it would also be interesting to explore alternative deep learning models for the prediction of Alzheimer's disease from FMRI data. For example, instead of using a convolutional LSTM model, we could consider using a convolutional transformer model, which has recently been shown to be effective in a wide range of NLP tasks.

Github: https://github.com/jfw225/ai-practicum

References

[1] G. Folego, M. Weiler, R. F. Casseb, R. Pires, and A. Rocha. Alzheimer's disease detection through whole-brain 3d-cnn mri. *Frontiers in Bioengineering and Biotechnology*, 8, 2020. URL https://www.frontiersin.org/articles/10.3389/fbioe.2020.534592/full.

[2] Hassan, Islam, Kotrotsou, Aikaterini, S. Bakhtiari, Ali, Thomas, Ginu, Weinberg, Jeffrey, Kumar, Jitendra, Sawaya, Raymond, Luedi, Markus, Zinn, Pascal, Colena, and Rivka. Radiomic texture analysis mapping predicts areas of true functional mri activity. *Scientific Reports*, 6, 05 2016. doi: 10.1038/srep25295.

[3] J. Li, B. Song, and C. Qian. Diagnosis of alzheimer's disease by feature weighted-lstm: a preliminary study of temporal features in brain resting-state fmri. *Journal of Integrative Neuroscience*, 2022. URL https://www.imrpress.com/journal/JIN/21/2/10.31083/j.jin2102056.

[4] W. Li, X. Lin, and X. Chen. Detecting alzheimer's disease based on 4d fmri: An exploration under deep learning framework. *Neurocomputing*, 2020. URL https://www.sciencedirect.com/science/article/pii/S0925231220301041.

[5] G. Liang, X. Xing, L. Liu, Y. Zhang, Q. Ying, A.-L. Lin, and N. Jacobs. Alzheimer's disease classification using 2d convolutional neural networks. *IEEE Xplore*, 2021. URL https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9629587.

[6] S. Wang, J. Cao, and P. S. Yu. Deep learning for spatio-temporal data mining: A survey. *CoRR*, abs/1906.04928, 2019. URL http://arxiv.org/abs/1906.04928.