CS230

Deep Learning Methods for Alzheimer's Disease Prediction

Project Category: Computer Vision

Fang Shu
Department of MS&E
Stanford University
fangshu@stanford.edu

Longling Tian
Department of CME
Stanford University
longling@stanford.edu

Abstract

Alzheimer's disease is a brain disorder that is projected to affect up to 14 million people in the U.S. by 2060 [10]. Though no cure currently exists for the neurodegenerative disease, it is possible to detect it by brain scans, such as Magnetic Reasoning Imaging (MRI). In this project, we built a ResNet-50 model to predict the stage of Alzheimer's disease from brain MRI images with an F-1 score of 73% on test data. We also explored the use of contrastive learning models for this task. The supervised contrastive learning model classifies normal, very mild, and mild Alzheimer's demented images better than CNN, but failed to predict the moderate demented class.

1 Introduction

Alzheimer's disease is a brain disorder that progressively destroys a person's memory and thinking skills which can possibly lead to the loss of conversational ability and response to the outside environment. According to the National Institute of Aging, Alzheimer's disease is the seventh leading cause of death in the U.S. and the most common cause of dementia among elderly people [11]. By 2020, 5.8 million Americans were living with Alzheimer's disease, and the number is projected to increase to 14 million by 2060.

However, the ultimate cause of the disease is still unclear. Potential causes such as age, family history, diet, and living environment have all been speculated, but none have been confirmed [2]. Nevertheless, as changes in the brain may begin years before the first symptoms begin to show, it is crucial for the medical field to have tools to predict or possibly confirm Alzheimer's presence and subsequently the disease stage. Currently, tools such as Computed Technology (CT), Magnetic Resonance Imaging (MRI), and Positron Emission Tomography have all been used to detect the disease. In particular, MRI scans of the head is the preferred tool, as it uses a conjunction of powerful magnetic fields and radio frequency pulses to produce detailed images of internal body structures. An MRI scan may be normal during the first stages but may show a decrease in the temporal and parietal lobes part of the brain during later stages. Hence, by employing deep learning, we hope to gain more insights into how to accurately detect Alzheimer's based on MRI scans.

Our main motivation behind this project is that we believe Alzheimer's disease is an important health issue that is unfortunately often overlooked. For instance, we have met people where elderly people in the family showed symptoms of memory loss for years, but the family did not think about Alzheimer's is the cause. Therefore, we believe that if the disease could be detected as soon as possible with a reliable tool, it would both benefit the patient and the patient's family, and could also allow scientists to find the underlying cause of the disease as soon as possible.

For this project, our input is MRI scans that correspond to one stage of Alzheimer's disease. We will build a variety of Deep Learning Algorithms on those scans to classify the patients into one of those four stages. In particular, we built different Convolutional Neural Networks Architectures. We also implement two different methods of Contrastive Learning. Finally, we compare the classification accuracy of the different networks.

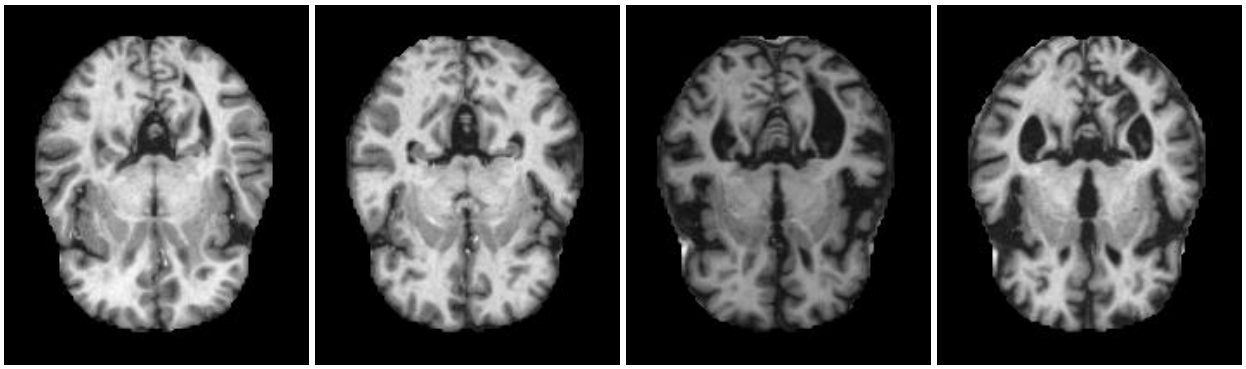
2 Related work

Different studies have employed a variety of machine learning or deep learning methods to detect and classify the stages of Alzheimer's disease. For example, a study conducted by the Governmental Model Engineering College Kochi of India [13] found pre-trained Deep Neural Networks such as AlexNet, VGG-16, and VGG-19 achieved an accuracy of around 85 percent to 90 percent on the Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset. One study by the Georgia State University [6] used deep CNN models such as Inception-v4 and ResNet and achieved F1-scores around 0.92. Another study proposed a residual and plain Convolutional Neural Network Architecture for 3D brain MRI classification. The proposed architecture achieved comparable results to previously used preprocessing approaches but is able to avoid handcrafted feature generation [8].

CS230: Deep Learning, Winter 2018, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

3 Dataset and Features

Our data is obtained from Kaggle. In particular, the data was compiled in 2020 by Sarvesh Dubey.[1] He hand-collected images and labels from various websites, and the entire dataset consists of 6400 MRI images each with a size of $176 \star 208$. The training set consisted of 5121 scans, and the test set contained 1279 scans. The data can be classified into four stages: Non-Demented, Very Mild Demented, Mild Demented, and Moderate Demented. We develop deep learning models that can discern whether the disease is present or absent and if present, to classify the stage. We used this dataset as it has the largest number of samples among all MRI scan datasets we could find. But we also note that a sample size of 6400 images is not that large for most deep learning architectures. A typical MRI scan from each class is shown in Fig. 1 below.



(a) (b) (c) (d)

Figure 1: A typical MRI scan of the (a) Normal (b) Very Mild Demented (c) Mild Demented (d) Moderate Demented class

3.1 Exploratory Data Analysis

We want to get a general view of the difference between each stage of Alzheimer's. We first take the mean of each class. We observe that as the stage progresses, brain atrophy in the cerebral cortex (the outer layer that lies on top of the cerebrum) becomes more evident, as shown in Figure 4 of the appendix. We then subtracted the mean of the other 3 classes from the NonDemented class and observed a similar result. We show this result in Figure 5 of the appendix.

3.2 Data Augmentation

According to figure 8(a), the dataset is highly imbalanced. The moderately demented and the mildly demented class are scarce compared to the other two classes. First, we tried modifying the loss function to weigh more toward minority classes. However, the optimizer failed to converge no matter how we set up class weights. Therefore in this study, we performed data augmentation on the two minor classes using TensorFlow *ImageDataGenerator*. Operations like flipping, shifting, and modifying brightness were carried out. Figure 8(b) shows the distribution of 4 classes after data augmentation.

4 Methods

4.1 Model Architectures

Several state-of-the-art convolutional neural network (CNN) models were applied to our dataset. Specifically, VGG-16, ResNet-50, and Inception-V3 models were chosen as baseline models because of their popularity and good performance in similar computer vision tasks. [9] [14] The similarity of the three models is that they all have some kind of convolutional blocks at first, followed by a pooling layer, several fully-connected layers, then a softmax unit at the end. Compared to VGG-16 as a standard CNN model, ResNet-50 has residual blocks that shortcut previous layers to prevent vanishing gradients. On the other hand, the InceptionV3 model breaks large convolutional layers into multiple $3 * 3$ and $1 * 1$ layers to minimize computational cost.

We also considered the possibility of transfer learning the ImageNet parameters. However, simply doing this and refitting dense layers was not applicable to this study, as the model predicted all images to the same class. This is likely because the difference between images in different classes is very small compared to the dataset on ImageNet. Therefore, we retrained all of the model parameters for this study.

4.2 Loss Function and Optimizers

Since this study is a prediction with multiple classes, categorical cross entropy was used as the loss function. Similar to binary cross-entropy, the formula of categorical cross entropy is given by $-\sum_{c=1}^{M} y_{o,c} \log(p_{o,c})$, where y is the indicator if class label c is the correct classification of observation o, and p is the predicted probability observation o is of class c.

In this study, Adam and Stochastic Gradient Descent (SGD) with momentum were chosen as the optimizers of our models. Both are adaptive methods evolved from the normal gradient descent scheme: $\theta := \theta - \eta \nabla_\theta J(\theta)$. SGD with Momentum takes in all the past gradients and performed a weighted average with the current gradient, then perform a gradient descent using the new value. $v_t = \gamma v_{t-1} + \eta \nabla_\theta J(\theta);\ \ \theta := \theta v_t$. In this way, SGD with momentum gathers the momentum in one direction and accelerates

gradient descent in that direction, while decreasing that in the other directions. This generalizes well to most problems having a steep curve in one direction but not the other.

On the other hand, Adam is very similar to momentum except that it also takes into account the second-order derivative (2nd moment) of the loss function. In this way, Adam has one more hyperparameter to tune.

4.3 Evaluation Metrics

We have 3 evaluation metrics for this study: accuracy, F-1 score, and Area Under ROC Curve (AUC) score. We did not choose precision or recall individually because we want the model both to learn the difference between all classes and identify the correct stage of illness (high recall); and be responsible for its predictions because medical treatments cost highly (high precision). Therefore, we chose F-1 score, the harmonic mean of precision and recall, as the major metric to look upon.

4.4 Novelty - Contrastive Learning

Contrastive learning is the novelty part of this study. It contrasts samples against one another to learn attributes that are common to or different among different classes. In this way, our model will be able to identify attributes and classify them at the same time. Contrastive learning has rarely been implemented on Medical image classification tasks, so this study is an exploration of the feasibility of CL models compared to canonical CNN models illustrated earlier.

4.4.1 Supervised Contrastive Learning

We follow the general workflow in this paper [7], which first extended contrastive learning to the fully-supervised setting. We use all 5121 MRI scans as our training set and the other 1279 MRI scans as our test set. To address the class-imbalance problem, we perform *Synthetic Minority Over-Sampling* [3] of our samples and generated 2560 samples of each class.

We first perform data augmentation. We normalize each data sample $\boldsymbol{x}$, randomly flip the images horizontally, and apply rotation randomly. Therefore, for each batch of data, we apply data augmentation twice to obtain two copies of the batch. We then forward propagate both copies to the Encoder Network that follows.

We then train an Encoder to learn to produce vector representations of the MRI scans where representations of the image in the same class will be more similar compared to representations in different classes. The Encoder Network maps the image samples $\boldsymbol{x}$ to a representation vector $\boldsymbol{r}$. We input both augmented samples separately to the Encoder, which results in a pair of representation vectors. We first experimented with a ResNet Encoder and obtain a normalized embedding, or feature vector.

Finally, the vector representation is further propagated through a projection network, which we discard at test time. The Projection Network maps $\boldsymbol{r}$ to a vector $\boldsymbol{z} = Proj(\boldsymbol{r})$. We instantiate this Projection Network as a multilayer perceptron with a single hidden layer and an output vector.

We compute the loss on the outputs of the Projection Network. Instead of using the commonly-used Cross Entropy Loss in image classification problems, we measure the loss by leveraging label information. This Supervised Contrastive Loss (see section 8.1 of the Appendix) contrasts the set of all samples from the same class as positives against the negatives from the remainder of the batch. Therefore, normalized embeddings from the same class are pulled closer together than embeddings from different classes.

4.4.2 Semi-Supervised Contrastive Learning

Semi-Supervised Learning relies on partially labeled datasets. It is label-efficient by utilizing a relatively small portion of labeled data. We largely follow the *Simple Framework for Contrastive Learning* (SimCLR) framework that was proposed in this paper [4]. SimCLR learns representations by maximizing agreement between differently augmented views of the same MRI scan through a contrastive loss in the latent space.

We first transform our dataset into a partially labeled format. We randomly select 90 % of the MRI scans in each of the four classes, mix them together, and use them as our large, unlabelled portion of the dataset. We then use the remaining 10 percent of the scans as our labeled portion of the dataset. Therefore, we simultaneously load a large batch of unlabelled scans along with a smaller batch of labeled scans when training the model.

We perform data augmentation which transforms any MRI scan randomly and results in two correlated views of the same scan. We consider them as a positive pair and use random Gaussian blur and random horizontal flips.

After data augmentation, we then defined a base encoder that extracts representation vectors from augmented data examples. We then pretrained the Encoder on the unlabelled images with Contrastive Loss, and also use Contrastive Accuracy to monitor the pretraining performance. We then attach a nonlinear projection head to the top of the Encoder, which maps the representation vectors to the space where Contrastive Loss is applied.

Next, by attaching a single randomly fully-connected classification layer on top of the projection head, we finetune the encoder on the labeled examples. During this process, we measure our model's performance by using the Contrastive Loss function, which is defined in section 8.2 of the Appendix.

Model	Accuracy (%)	Precision (%)	Recall (%)	F-1 score (%)
VGG16	32	15	17	16
Inception	49	15	24	17
Resnet-50	35	9	25	13

Table 1: Baseline Model Performance

5 Results & Discussions

5.1 Baseline CNN Model

Table 1 shows the performance of our baseline models. The confusion matrices are also given in figure 6(a) in the appendix. Almost every model performed badly, especially in the minority classes if one looks into the confusion matrix. Besides, the models listed on the table are all in which we trained all parameters. The transfer learning models were even worse and predicted only "normal", so we ended up training all parameters for this study. Since it's hard to choose the best model only from the results in the table, we turn to the model size and training time. Among the three, ResNet and InceptionV3 are only about 20% in size compared to VGG-16. And between those two ResNet-50 took less epochs to converge. model parameters and ResNet50 model was reported in other studies to have better performance than VGG16 [9] and InceptionV3 [14], we chose to conduct further research on it.

5.2 Data augmentation

One can tell from the previous results that it's crucial to perform some data augmentation. As mentioned in part 3.3, ImageDataGenerator class was applied to augment the two minority classes. The performance on the validation set of the same Resnet-50 model fitted on original data and augmented data was shown in figure 6. When training on the original dataset, the model only predicts normal or mild demented. After augmentation, the model learns more information about on the two minority classes and can predict them.

5.3 Hyperparameter Tuning

We performed hyperparameter tuning on the ResNet model with the augmented dataset. Tuned hyperparameters include the size of the two fully-connected layers, learning rate, β_1 and β_2 for the Adam optimizer, and momentum for the SGD optimizer. All tunings were done with the Keras tuner RandomSearch class. [12] The optimal hyperparameter values for dense layer units were $(512, 128)$ and $(1024, 256)$, learning rate $2.25 * 10^{-4}$ and $2.7 * 10^{-3}$ for Adam and SGD respectively. The β_1 and β_2 values were 0.963 and 0.9997 for Adam, and momentum was 0.95 for SGD.

Figure 7 visualizes model performance with Adam and SGD. SGD optimizer is slightly better in predicting the minority classes, while Adam did well on non-demented images. The training speed for the two optimizers was also similar, both around 10-15 iterations to converge. Since Adam was more commonly used in academia, it was chosen to be our final model. It reached 73% accuracy, 67% recall, 77% precision, and 73% F-1 score on the test set. The confusion matrix is given below.

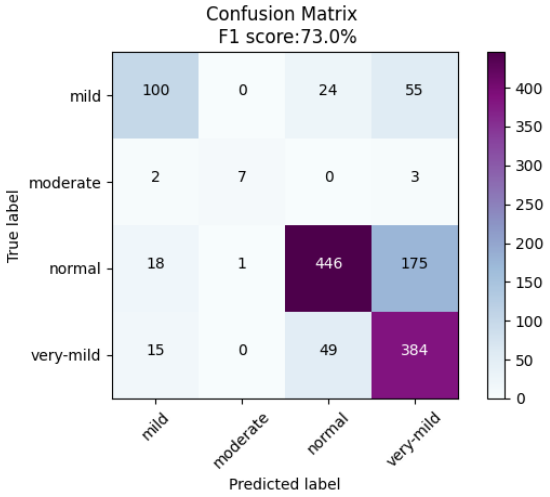


Figure 2: Best Model Performance On Test Set

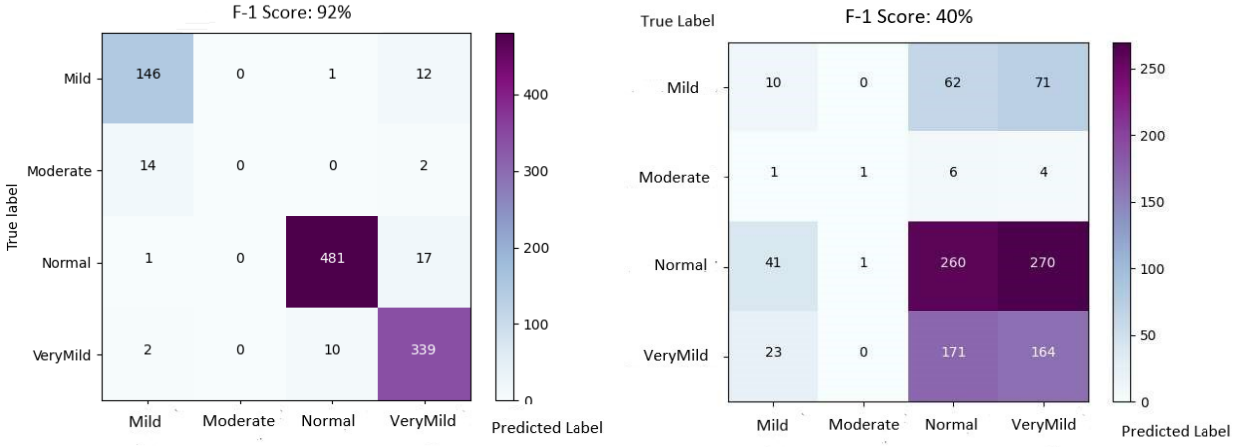
The model predicted well for all 4 classes, with no obvious biases toward any class. A 73% F1 score also beat our baseline models by about 4 times. However, it's worth noticing that if we compare the result of the same model on the validation set (Fig. 7(a)) and on the test set (Fig. 2), the model performance on the validation set largely outperforms. One possible explanation is that since the training and test images all come from Kaggle [1], they might actually come from different distributions. But on the other hand, this also indicates that our model is not 100% robust to images coming from all distributions. Therefore, future studies should be carried out on how this model could be generalized to predict any given MRI images. One possible improvement is to find more relevant data to feed our model.

5.4 Abnormal Behavior During Early Training Period

Throughout the training period, our Resnet50 model exhibited a weird loss curve compared to VGG16. According to figure 9, the VGG model (which represents normal model training behavior) has both training loss and validation loss decreasing smoothly over epochs. However, on the ResNet model, the loss curves on both curves were very different. On the training set, the model achieved losses very close to zero just after $2 \sim 3$ epochs. However, the model did not perform any predictions on the validation set until epoch 5, and once past this period, suddenly learned how to do well. This behavior was seen on ResNet models with both Adam and SGD optimizers. Similar behaviors were reported in the original ResNet-50 paper where there was a sudden drop in loss on both the training set and test set. [5] The authors did not suggest a reason for this, but it's likely because the learning rate changes during certain epochs. Besides, the increasing behavior during the first few epochs is worth studying. But anyways, for our ResNet model, both curves still converged at the end.

5.5 Contrastive Learning

By using the Random Search Tuner [12], we tuned hyperparameters including batch size, learning rates, hidden units, and the temperature parameter for both Contrastive Learning methods. For Supervised Contrastive Learning, we found that a batch size of 64, a learning rate of 0.001, and a temperature parameter of 0.05 gave the best prediction results. We also tried out different architectures such as ResNet100 and EfficientNetB2 for the Encoder network and found that ResNet50 gave the best performance along with using a Stochastic Gradient Descent optimizer in the encoder and an Adam optimizer in the projection network. We observed that the Supervised Contrastive Learning had an F1-score of 92%, and the confusion matrix is shown in the left figure. However, we observe the model fails to accurately classify any of the Moderate Demented MRI scans.



(a) (b)

Figure 3: (a) Supervised Contrastive Learning (b) Semi-supervised Contrastive Learning

On the other end, the semi-supervised contrastive learning model was only able of achieving an F-1 score of 40%. We also tried increasing the proportion of unlabeled data to 95 %, but the improvement was minimal. We speculate that the results would improve if the dataset size was greatly increased. For example, many studies used the STL-10 dataset, which contained 100000 unlabelled images and 5000 labeled images.

6 Conclusion & Future Work

In this project, we trained Convolutional Neural Networks (CNNs) and Contrastive Learning (CL) models to predict stages of Alzheimer's disease from MRI brain images. To tackle imbalanced data, we performed data augmentation on training sets of mild demented and moderated demented images. Among different state-of-the-art CNN models, ResNet-50 was chosen because of its fast convergence and a smaller amount of model parameters. After all hyperparameter tunings, the best model achieved a decent 73% F-1 score on the test set, while predicting each individual class with recall value $\geq 60\%$.

At the same time, there're also multiple areas to possibly conduct a future study. First of all, the same ResNet-50 model had about 15% more accuracy on the validation set than on the test set. In the future, we can feed in more brain MRI images coming from as many different sources (including, but not limited to the ADNI dataset) to our model to make it more robust to unseen data. Besides, the abnormal behavior of ResNet models in the first few training epochs is worth studying into.

For the contrastive learning portion of this study, supervised CL models performed surprisingly well, outperforming the ResNet-50 model with a 92% F-1 score. However, it failed to predict the moderate stage images. We believe that the performance of Semi-Supervised Contrastive Learning should further improve if we had a larger dataset. One could also improve this model by improving the data augmentation techniques if he/she still wants to use the Kaggle dataset. For future work, we plan to fit our models on the ADNI dataset once it's correctly preprocessed. Meanwhile, we'd also look for other MRI images, especially those of the more severe class.

7 Contributions

Both members contributed equally to the project. Specifically, Fang focused on implementing the two Contrastive Learning methods, while Longling focused on implementing baseline models and CNN models. Both members contributed equally to writing up the proposal, milestone, and report. We also thank our TA, Sarthak Consul, for his generous help on our project.

References

[1] *Alzheimer's Dataset (4 class of Images)*. en. URL: https://www.kaggle.com/datasets/tourist55/alzheimers-dataset-4-class-of-images.

[2] Centers for Disease Control and Prevention. *Alzheimer's Disease and Related Dementias*. https://www.cdc.gov/aging/aginginfo/alzheimers.htm#:~:text=Alzheimer's, 2020.

[3] Nitesh V Chawla et al. "SMOTE: synthetic minority over-sampling technique". In: *Journal of artificial intelligence research* 16 (2002), pp. 321–357.

[4] Ting Chen et al. "A simple framework for contrastive learning of visual representations". In: *International conference on machine learning*. PMLR. 2020, pp. 1597–1607.

[5] Kaiming He et al. *Deep Residual Learning for Image Recognition*. 2015. DOI: 10.48550/ARXIV.1512.03385. URL: https://arxiv.org/abs/1512.03385.

[6] Jyoti Islam and Yanqing Zhang. "Brain MRI analysis for Alzheimer's disease diagnosis using an ensemble system of deep convolutional neural networks". In: *Brain informatics* 5.2 (2018), pp. 1–14.

[7] Prannay Khosla et al. "Supervised contrastive learning". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 18661–18673.

[8] Sergey Korolev et al. "Residual and plain convolutional neural networks for 3D brain MRI classification". In: *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*. 2017, pp. 835–838. DOI: 10.1109/ISBI.2017.7950647.

[9] Sheldon Mascarenhas and Mukul Agarwal. "A comparison between VGG16, VGG19 and ResNet50 architecture frameworks for Image Classification". In: *2021 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON)*. Vol. 1. 2021, pp. 96–99. DOI: 10.1109/CENTCON52345.2021.9687944.

[10] Kevin A Matthews et al. "Racial and ethnic estimates of Alzheimer's disease and related dementias in the United States (2015–2060) in adults aged 65 years". In: *Alzheimer's & Dementia* 15.1 (2019), pp. 17–24.

[11] National Institute on Aging. *Alzheimer's Disease Fact Sheet*. https://www.nia.nih.gov/health/alzheimers-disease-fact-sheet, 2021.

[12] Tom O'Malley et al. *KerasTuner*. https://github.com/keras-team/keras-tuner. 2019.

[13] PC Muhammed Raees and Vinu Thomas. "Automated detection of Alzheimer's Disease using Deep Learning in MRI". In: *Journal of Physics: Conference Series*. Vol. 1921. 1. IOP Publishing. 2021, p. 012024.

[14] Yuan Yang et al. "A comparative analysis of eleven neural networks architectures for small datasets of lung images of COVID-19 patients toward improved clinical decisions". In: *Computers in Biology and Medicine* 139 (Dec. 2021), p. 104887. ISSN: 0010-4825. DOI: 10.1016/j.compbiomed.2021.104887. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8461289/ (visited on 12/10/2022).

8 Appendix-A: Loss functions and ADNI Dataset

8.1 Supervised Contrastive Loss

$$L^{sup} = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{p \in P(i)} log \frac{exp(z_i \cdot z_p/\tau)}{\sum_{a \in A(i)} exp(z_i \cdot z_a/\tau}$$

, where $P(i)$ is the set of all positives in the batch corresponding to the anchor i, $|P(i)$ is its cardinality, $z_i = Proj(Enc(\tilde{x_i})$, τ is the temperature parameter, and the index i is called the anchor.

8.2 Semi-Supervised Contrastive Loss Function

Randomly sample a minibatch of *N* examples and define the contrastive prediction task on pairs of augmented examples derived from the minibatch. This results in 2N data points, and we treat the other *2N-2* augmented examples within the minibatch as negative examples.

Then the loss for a pair of positive examples (i,j) is defined as

$$l_{i,j} = -log \frac{exp(sim(z_i, z_j)/\tau)}{\sum_{k=1}^{2N} I_{k \neq i}(exp(sim(z_i, z_k)/\tau)}$$

, where $sim(\mathbf{z}_i, \mathbf{z}_j = \frac{\mathbf{z}_i^T \mathbf{z}_j}{||\mathbf{z}_i|| ||\mathbf{z}_j||}$ measures the cosine similarity between z_i, z_j, and τ is the temperature hyperparameter.

8.3 ADNI Dataset

As our original dataset contained only 6400 MRI scans, we also tried to gather more MRI scans. We gathered additional data from the Alzheimer's Disease Neuroimaging Initiative (ADNI) website. ADNI is a long-term study designed to develop clinical and imaging biomarkers for the early detection and tracking of Alzheimer's. We applied for and downloaded 1200 MRI scans from the ADNI website. After preprocessing, we used our original dataset as the training set and this new dataset as the test set. However, both Supervised and Semi-supervised Contrastive Learning failed to give decent results. We speculate that this may be due to the fact that some of the ADNI MRI scans require more preprocessing and augmentation, such as skull-stripping. Additionally, as many images were taken from different angles of the brain and had different sizes and voxel intensities, we may need to spend more time carefully selecting the images. Due to time constraint, we will continue to work on preprocessing the ADNI dataset using tools such as the Statistical Parametric Mapping and use it to further test our models' robustness.

9 Appendix-B: Images and Supplementary Plots

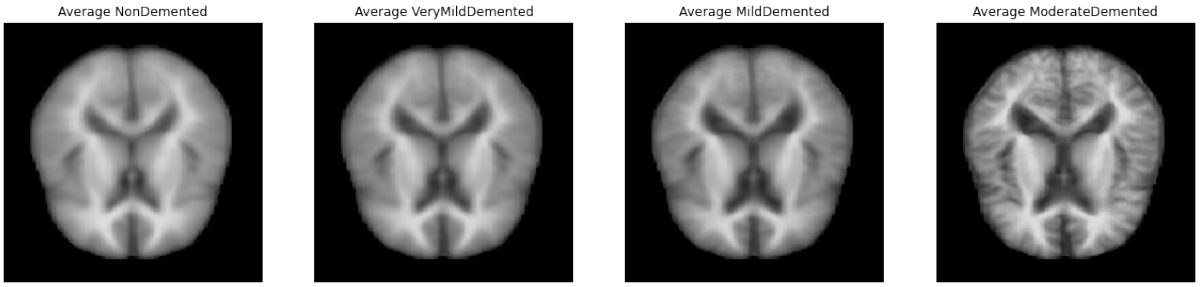
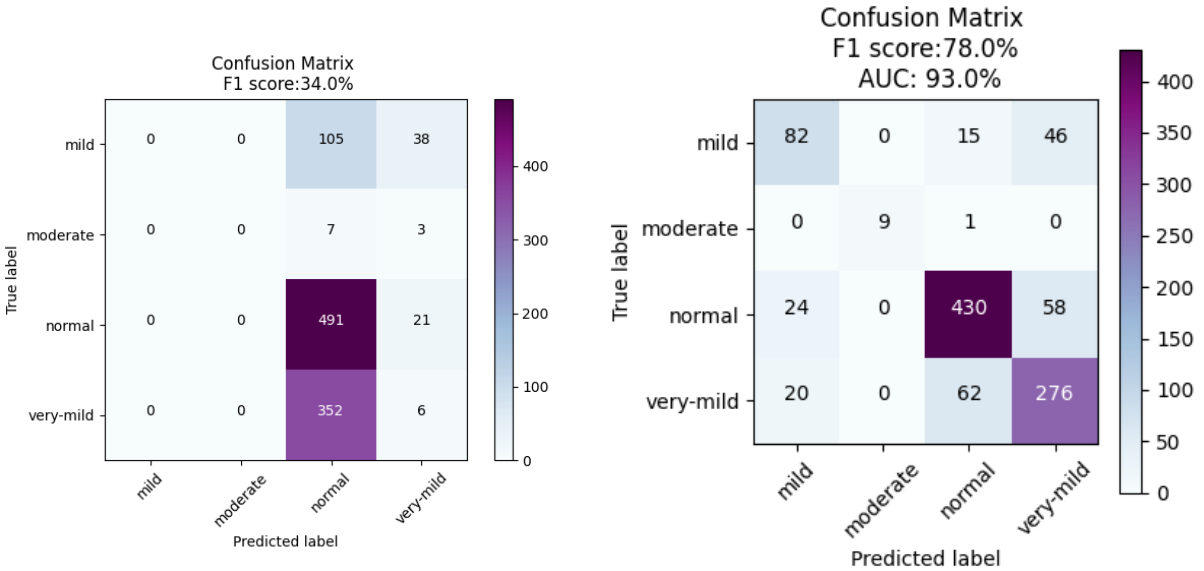


Figure 4: Average Pixel Value of MRI images in Each Stage of Alzheimer's Disease

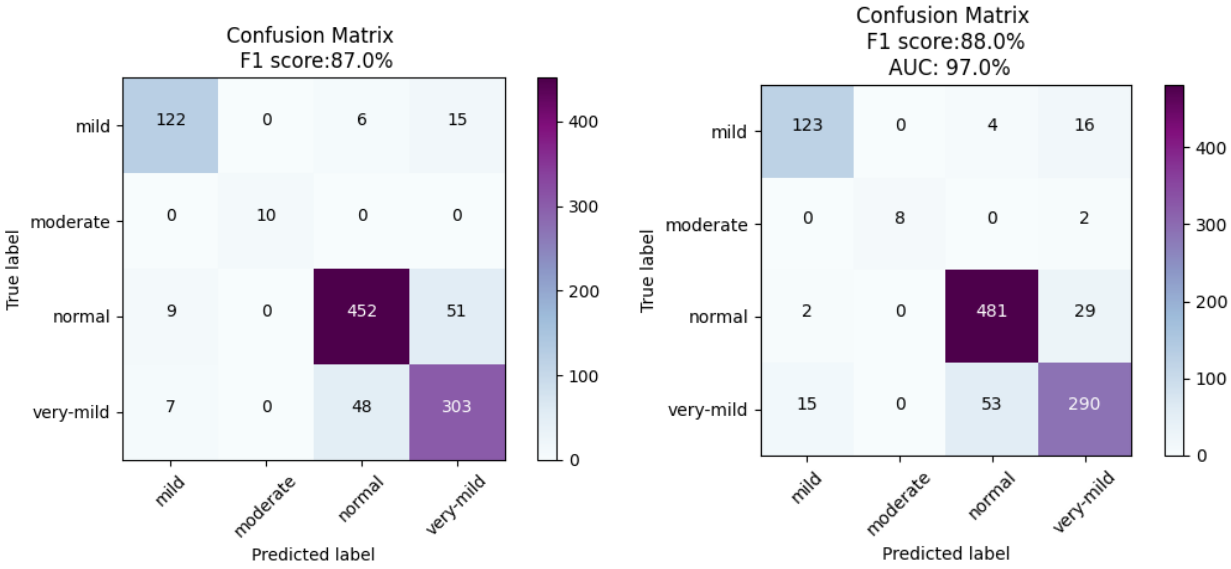
(a) (b) (c)

Figure 5: (a) Difference between NonDemented mean and VeryMildDemented mean
(b) Difference between NonDemented mean and MildDemented mean
(c) Difference between NonDemented mean and ModerateDemented mean



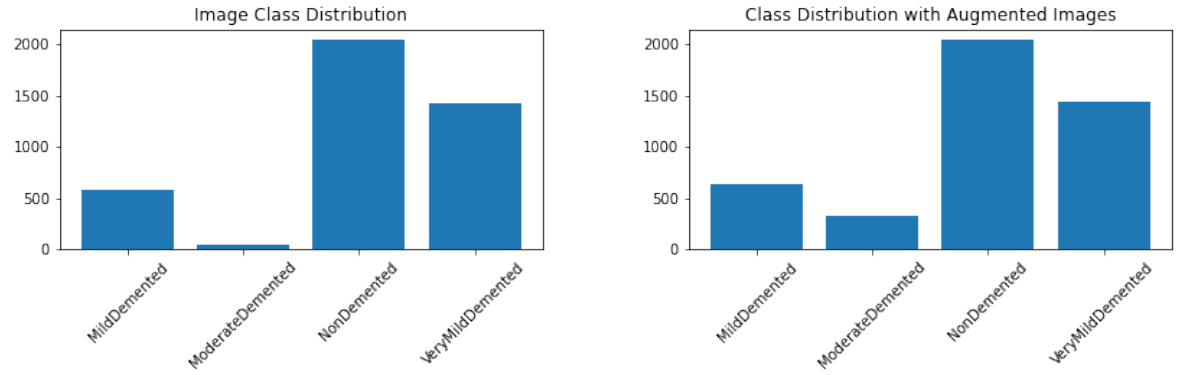
(a) (b)

Figure 6: (a) Performance on validation set using original data (b) Performance on validation set using augmented data



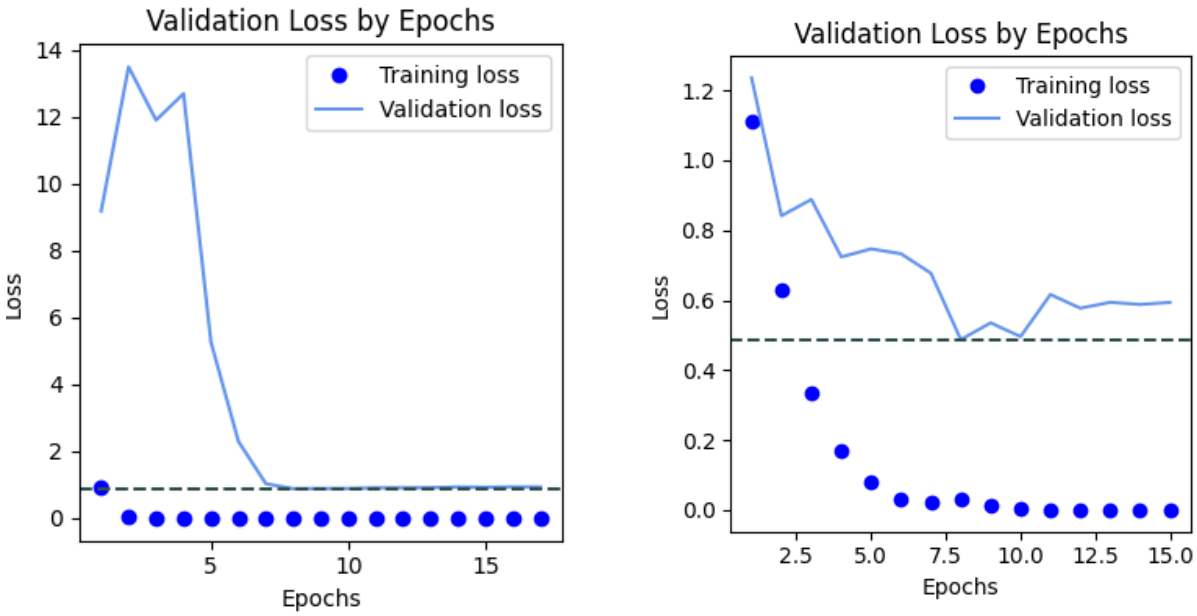
(a) (b)

Figure 7: (a) Performance of Tuned ResNet50 model with Adam optimizer on validation set
(b) Performance of Tuned ResNet50 model with SGD optimizer on validation set



(a) (b)

Figure 8: (a) Distribution between classes before augmentatation(b) Distribution between classes after augmentatation



(a) (b)

Figure 9: Training and Validation Loss for (a) ResNet-50 (b) VGG-16