**Predictive Classification for Alzheimer Disease**

Aaron Ramirez, Jaymin Patel, and Geoffrey Fadera

University of San Diego

Masters of Applied Artificial Intelligence

AAI-501: Introduction to Artificial Intelligence

Professor David Friesen

April 15 2024

**Background on Alzheimer Disease and Project’s Goal**

Alzheimer's disease is a neurodegenerative disorder that affects cognitive functions such as memory and behavior (National Institute on Aging, 2023). It is primarily caused by dementia, although the exact etiology remains incompletely understood, it is believed to be influenced by genetic, age-related, and lifestyle factors. Symptoms of Alzheimer's disease progressively worsen, leading to frequent memory loss, difficulty in performing simple tasks, and sudden mood changes. While there is no cure for Alzheimer's disease, available medications can help manage symptoms. Early detection is crucial in managing the disease, as it enables the development of appropriate treatment plans to mitigate its progression and plan for the future.

Given the severity of Alzheimer's disease, this project aims to leverage AI techniques and algorithms to develop a model for early detection and severity assessment. The disease is categorized into four stages: Non-Demented, Mild Demented, Very Mild Demented, and Moderate Demented. Various models were tested and evaluated, including Convolutional Neural Network (CNN), Deep Neural Network (DNN), CNN with graph-based features, K-nearest Neighbors (KNN), Support Vector Machine (SVM), Random Forest Classifier (RFC), RFC with Histogram of Gradient (HOG) feature extraction, and Recurrent Neural Network (RNN).

**Related work**

In the realm of Alzheimer's disease detection, researchers have been diligently working on finding effective solutions for many years. Several noteworthy studies have contributed significant advancements in this area. For instance, Irshad Ahmad and his research team conducted a study at Jouf University, demonstrating impressive results with a recognition rate of 99.35%. They achieved this milestone by employing principal component analysis in conjunction with stepwise linear discriminant analysis and an artificial neural network (Ahmad et al., 2023). Additionally, a research team from China successfully predicted Alzheimer's disease from 2.5D MRI images with an accuracy of approximately 80% (Lin et al., 2018). Furthermore, A.M. El-Assy and colleagues from Mansoura, Egypt, achieved remarkable success by categorizing and predicting Alzheimer's disease from MRI images with an accuracy exceeding 99% (El-Assy et al., 2024). These findings collectively highlight the promising progress being made in Alzheimer's disease research, indicating a positive trajectory towards the development of robust diagnostic methods. With continued efforts, it is hopeful that researchers will devise foolproof techniques to address this pressing issue effectively.

**Data Set and Data Splitting**

In our research, we utilized a dataset sourced from Kaggle, comprising a diverse collection of MRI images obtained from multiple websites. This dataset was categorized into four distinct classes, as mentioned earlier, which were utilized for classification purposes. The dataset initially consisted of 6400 images, which had already been pre-split into training and testing subsets. Specifically, approximately 20% of the images were allocated for testing purposes, while the remaining 80% were designated for training the models. This pre-allocated split allowed us to ensure a standardized evaluation process and facilitated the seamless implementation of our testing procedures.

**Data Preprocessing**

In our preprocessing pipeline, we adhered to a standardized set of procedures tailored to meet the specific requirements of each individual model while also incorporating a general preprocessing approach. Initially, all images underwent grayscale conversion to ensure uniformity across the dataset. Subsequently, each image was resized to the target dimensions of 224 by 224 pixels. Additional preprocessing steps included thresholding and connected component analysis to eliminate background black space, resulting in brain outlines closely aligned with the boundaries of the images. Furthermore, pixel normalization was performed to enhance the effectiveness of model training. These preprocessing steps were applied uniformly to both the training and testing datasets, ensuring consistency in data preparation. Ultimately, the processed images were converted into arrays, rendering them suitable for utilization within our AI models. The shapes of the images arrays were:

Train:

Image shape: (5121, 224, 224)

Labels shape: (5121,)

Test:

Image shape: (1279, 224, 224)

Labels shape: (1279,)

**Metrics definition**  
The evaluation of our models revolves around four key metrics: accuracy, precision, recall, and F1-score.

* **Accuracy:** This metric quantifies the percentage of correct predictions made by the model, providing an overall measure of its performance.
* **Precision:** Precision gauges the accuracy of a specific classifier by measuring the proportion of true positives among all positive predictions made by the classifier. It focuses on the precision of positive predictions.
* **Recall:** Also known as sensitivity, recall assesses the model's ability to identify all actual positives, including those that were incorrectly classified as negatives (false negatives). It answers the question of how many actual positives were correctly identified by the model.
* **F1-score:** The F1-score is a harmonic mean of precision and recall, providing a balanced measure of performance that considers both false positives and false negatives. It offers a single value that encapsulates both precision and recall, making it a useful metric for overall model evaluation.

**Models with metrics**

**CNN**

Additional preprocessing steps were implemented to enhance the model's performance. The labels were hot encoded to facilitate classification. The sequential model architecture comprised three 2D convolutional layers followed by Batch Normalization, which normalized the activation of the convolutional layers to expedite training. Max pooling layers were then employed to down sample the data and retain only the maximum values within each region. Subsequently, the data was flattened into a 1D vector to connect to the fully connected layer, collapsing the spatial dimensions of the image into a single dimension. The first fully connected layer initiated the classification process, followed by a dropout layer to mitigate overfitting. Finally, the last dense layer executed the prediction task.

To optimize learning, the Adam optimizer and categorical cross-entropy loss function were utilized. Additionally, early stopping was incorporated to halt training if validation did not improve. Two learning rate scheduling techniques were evaluated during training: Cyclic Learning Rate, which dynamically adjusts the learning rate, and exponential decay learning. Empirical results indicated that exponential decay learning outperformed Cyclic Learning Rate in terms of maintaining a consistent model, particularly with respect to validation loss and validation accuracy increases.

Test Loss: 1.3505847454071045

Test Accuracy: 0.7208756804466248

precision recall f1-score support

MildDemented 0.69 0.45 0.55 179

ModerateDemented 1.00 0.33 0.50 12

NonDemented 0.77 0.81 0.79 640

VeryMildDemented 0.66 0.71 0.68 448

accuracy 0.72 1279

macro avg 0.78 0.58 0.63 1279

weighted avg 0.72 0.72 0.72 1279

**DNN**

The preprocessing steps for the CNN model were akin to those of the previous model. The images were trained with batch size to preserve the original batch structure. The model architecture was sequential, beginning with flattening the data and subsequently incorporating two dense layers coupled with batch normalization. Dropout was included, albeit with a reduced rate. Similar to the CNN model, the Adam optimizer and categorical cross-entropy loss function were employed, along with exponential decay learning scheduling.

A novel addition to this model was the utilization of an Image Data Generator. This tool augmented various parameters of the images, introducing random shifts in features to facilitate more effective learning. By introducing this level of randomness and variation, the model could better generalize and adapt to unseen data, ultimately enhancing its performance.

Test Loss: 1.1579606533050537

Test Accuracy: 0.5215011835098267

precision recall f1-score support

MildDemented 0.00 0.00 0.00 179

ModerateDemented 0.00 0.00 0.00 12

NonDemented 0.52 0.98 0.68 640

VeryMildDemented 0.47 0.08 0.14 448

accuracy 0.52 1279

macro avg 0.25 0.27 0.21 1279

weighted avg 0.43 0.52 0.39 1279

**CNN with Graph base features**

The preprocessing pipeline for this model begins by organizing the images into batches and processing them using the MobileNet\_v2 model, which is optimized for feature extraction. The processed images are then fed into a feature extraction function to extract relevant features, which are subsequently simplified using principal component analysis (PCA) to reduce feature complexity. Following PCA, a function computes the similarity between the extracted features and vectors using cosine similarity, resulting in a feature similarity matrix. An adjacency matrix is then constructed based on this feature similarity matrix.

Both the training and test sets undergo this preprocessing pipeline, with the inclusion of a grayscale index for the model. The model architecture itself resembles a CNN model, similar to the previously described CNN model. However, it includes additional inputs such as image input, graph input, and adjacent input. These inputs are concatenated and passed through a flattened layer before being fully connected. Additionally, a cyclic learning rate scheduler is applied to enhance training results.

Test Loss: 1.9398

Test accuracy: 0.6450

precision recall f1-score support

MildDemented 0.51 0.39 0.44 179

ModerateDemented 1.00 0.25 0.40 12

NonDemented 0.65 0.91 0.76 640

VeryMildDemented 0.70 0.38 0.50 448

accuracy 0.65 1279

macro avg 0.72 0.48 0.52 1279

weighted avg 0.65 0.65 0.62 1279

**KNN**

In the additional preprocessing step, the images are flattened to ensure compatibility with the KNN algorithm. The KNN algorithm is then tested using different numbers of neighbors ranging from 1 to 14, and the results are reported.

The choice of ‘k’ was critical. Smaller values of ‘k’ can be noise sensitive and lead to overfitting, while large values tend to smooth the decision boundary excessively. After testing values ranging from 1 to 14, we found that k=3 yieled superior results compared to testing with 14 variations of the number of neighbors. The model was relatively precise in identifying all stages of Alzheimer’s particularly for the ‘NonDemented’ classes, however there is still further tuning that could be done to improve the overall model.

Test Accuracy: 0.73

precision recall f1-score support

MildDemented 0.64 0.59 0.61 179

ModerateDemented 0.50 0.58 0.54 12

NonDemented 0.79 0.81 0.80 640

VeryMildDemented 0.69 0.68 0.68 448

accuracy 0.73 1279

macro avg 0.66 0.67 0.66 1279

weighted avg 0.73 0.73 0.73 1279

**SVM**

In our implementation, we employed a linear kernel to start due to its simplicity and efficiency in handling binary classification problems. We used the method of a separate binary classifier being trained for each class. No additional preprocessing is performed in this step. Instead, the regularization parameter is set to one, as it has been found to be the optimal value for separating the hyperplanes and achieving the best results.

Accuracy: 0.6387802971071149

precision recall f1-score support

MildDemented 0.75 0.30 0.42 179

ModerateDemented 1.00 0.33 0.50 12

NonDemented 0.67 0.80 0.73 640

VeryMildDemented 0.57 0.56 0.56 448

accuracy 0.64 1279

macro avg 0.75 0.50 0.55 1279

weighted avg 0.65 0.64 0.62 1279

**RFC**

For RFC, the data is preprocessed by creating class weights to balance the data, especially since the "Moderate Demented" class has very little data compared to others. RFC iterates through each parameter combination, recording the performance metrics. It creates classifiers and fits them using the training data, calculating accuracy at each iteration. This process is repeated to find the best validation accuracy. Additionally, there were experiments with contrast equalization and Histogram of Gradient (HOG) techniques. However, both provided subpar results compared to traditional RFC. Although HOG performed slightly better with a cell size of 40x40, it was still considered subpar overall.

Validation Accuracy: 0.6677091477716967

precision recall f1-score support

MildDemented 1.00 0.12 0.22 179

ModerateDemented 0.00 0.00 0.00 12

NonDemented 0.73 0.84 0.78 640

VeryMildDemented 0.57 0.66 0.61 448

accuracy 0.68 1279

macro avg 0.57 0.41 0.40 1279

weighted avg 0.70 0.67 0.63 1279

**RNN**

The labels are encoded into integers, and class weights are added again for training balance. The gated recurrent layer consists of 128 units and is immediately followed by batch normalization. A dropout layer with a rate of 0.2 is included to improve learning. The model is compiled with the Adam optimizer and uses sparse categorical cross-entropy loss due to how the labels are processed. Testing was also conducted without class weights, and the results between the two scenarios were so similar that the difference was deemed negligible. Below are the results without the class weights.

Validation Accuracy: 0.70

precision recall f1-score support

MildDemented 0.85 0.31 0.46 179

ModerateDemented 1.00 0.25 0.40 12

NonDemented 0.72 0.86 0.78 640

VeryMildDemented 0.65 0.63 0.63 448

accuracy 0.70 1279

macro avg 0.80 0.51 0.57 1279

weighted avg 0.70 0.70 0.68 1279

**Results**

**Figure 1**

*Bar Plot for all Model’s Accuracy*

*A graph of blue rectangular objects

Description automatically generated with medium confidenceNote.* The graph above depicts the accuracies of all models. This show DNN had struggles to provide results and that KNN and CNN had the best results around 72%.

**Figure 2**

*Bar Plot for all ModerateDemented’s Precision*

*A graph of different models

Description automatically generatedNote.* The graph above depicts the precision of ModerateDemented of all models. This shows DNN and RFC could not correctly categorize ModerateDemented levels. KNN had some struggles as well with the results being around 50%. CNN, CNN with graph and SVM achieved 100% precision.

**Figure 3**

*Bar Plot for all NonDemented’s F1-Score*

*A graph of red rectangular bars

Description automatically generated with medium confidence*

*Note.* The graph above depicts the F1-Score of NonDemented of all models. This shows KNN CNN and RFC having scores right below at 80%. We have DNN performing the worst with around 75%.

**Figure 4**

*Bar Plot for all NonDemented’s F1-Score*

*A graph of different models

Description automatically generated*

*Note.* The graph above depicts the Recall of NonDemented of all models. This shows DNN being the best with 100% and CNN\_with\_graph following behind with around 90%.

Comparing all the results and metrics below is the order of best performance:

1. **K-Nearest Neighbors (KNN):**
   * Test Accuracy: 0.73
   * Precision, Recall, and F1-score: Balanced performance across all classes.
   * KNN shows good performance in predicting Alzheimer's disease.
2. **CNN:**
   * Test Accuracy: 0.72
   * Precision, Recall, and F1-score: Moderate performance with a slight emphasis on NonDemented and VeryMildDemented classes.
   * CNN demonstrates decent performance in identifying Alzheimer's disease from MRI scans.
3. **CNN with Graph:**
   * Test Accuracy: 0.65
   * Precision, Recall, and F1-score: Moderate performance across all classes.
   * CNN with Graph provides reasonable accuracy in classifying Alzheimer's disease.
4. **Recurrent Neural Network (RNN):**
   * Validation Accuracy: 0.70
   * Precision, Recall, and F1-score: Balanced performance, but slightly lower recall for MildDemented class.
   * RNN shows promising results but could be further optimized for better accuracy.
5. **Random Forest Classifier (RFC):**
   * Validation Accuracy: 0.67
   * Precision, Recall, and F1-score: Moderate performance, but struggles in predicting the ModerateDemented class.
   * RFC shows acceptable performance but needs improvement in predicting certain classes.
6. **Support Vector Machine (SVM):**
   * Accuracy: 0.64
   * Precision, Recall, and F1-score: Moderate performance with an emphasis on NonDemented class.
   * SVM performs reasonably well but could improve in predicting other classes.
7. **DNN:**
   * Test Accuracy: 0.52
   * Precision, Recall, and F1-score: Poor performance, especially in predicting the MildDemented and ModerateDemented classes.
   * DNN performs inadequately compared to other models and requires significant improvement.

Overall, KNN demonstrates the best performance among the tested models, with CNN and CNN with graph features following closely behind. Even though CNN with graph did not have the highest accuracy score, the consistency with all data is what reigns it in a higher position. However, further fine-tuning and optimization are still needed for all models to enhance their performance. It's important to note that none of the models achieved perfection yet, indicating room for improvement in future iterations.

**Conclusion**

In summary, this research explored seven different models to identify the optimal classification method for MRI images indicating potential demented symptoms. The results highlighted promising performance for KNN and both CNN architectures. Although RNN also showed promise, further refinement is necessary to enhance its performance, particularly in improving recall for mild demented cases and ensuring accurate scoring. Moving forward, the next steps involve fine-tuning KNN and both CNN models. For KNN, this includes additional testing on distance metrics and feature scaling. In the case of CNN with graph features, transitioning to a Graph Convolutional Network (GCN) and fine-tuning its hyperparameters and layers is essential. Similarly, for CNN, exploring ensemble models incorporating multiple CNNs with varied hyperparameters and metrics could lead to improved results. Also, modeling and graphing key features that are representing the data the most for classification. Continued tuning and validation with larger datasets are crucial steps to further advance these models, contributing to the field of Alzheimer's disease detection and potentially paving the way for clinical applications.

**References**

U.S. Department of Health and Human Services. (2023). *Alzheimer’s disease fact sheet*. National Institute on Aging. https://www.nia.nih.gov/health/alzheimers-and-dementia/alzheimers-disease-fact-sheet

Ahmad, I., Siddiqi, M. H., Alhujaili, S. F., & Alrowaili, Z. A. (2023). Improving alzheimer’s disease classification in Brain MRI images using a neural network model enhanced with PCA and Swlda. *Healthcare*, *11*(18), 2551. <https://doi.org/10.3390/healthcare11182551>

Lin, W., Tong, T., Gao, Q., Guo, D., Du, X., Yang, Y., Guo, G., Xiao, M., Du, M., & Qu, X. (2018). Convolutional neural networks-based MRI image analysis for the alzheimer’s disease prediction from mild cognitive impairment. *Frontiers in Neuroscience*, *12*. https://doi.org/10.3389/fnins.2018.00777

El-Assy, A. M., Amer, H. M., Ibrahim, H. M., & Mohamed, M. A. (2024b). A novel CNN Architecture for accurate early detection and classification of alzheimer’s disease using MRI Data. *Scientific Reports*, *14*(1). https://doi.org/10.1038/s41598-024-53733-6

Dubey, S. (2019, December 26). *Alzheimer’s dataset ( 4 class of images)*. Kaggle. https://www.kaggle.com/datasets/tourist55/alzheimers-dataset-4-class-of-images

**Appendix A**

Group Contribution

|  |  |  |
| --- | --- | --- |
| **Contributions** | | |
| Jay Patel | Geoffrey Fadera | Aaron |
| * Figure out meeting schedule * Helped look for dataset * Help formulate features needed for proposal * Reviewed over proposal * Worked on SVM and KNN * Created presentation with Geoffrey * Proofread final report * Help organize meetings | * Created the Slack group * Created initial meeting * Helped look for dataset * Reviewed over proposal * Worked on RNN and RFC * Created presentation * Proofread final report * Help get references | * Created second meeting * Helped look for dataset * Took team lead role for submission * Created rough draft of proposal * Worked on CNN, DNN, CNN Graph * Combined the Code * Created Comparison charts * Wrote the final report * Proof read presentation |