LEWIS UNIVERSITY

CLASSIFYING THE SEVERITY OF ALZHEIMER'S DISEASE USING MACHINE LEARNING ON MRI IMAGES

RESEARCH PROJECT SUBMITTED

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE, DATA SCIENCE

BY

BENJAMIN J HAMMACK

DIRECTOR: DAWN M GRAUNKE, MS, PhD

ROMEOVILLE, IL

DATE: JULY, 2020

Abstract

This paper covers the process by which neural network and support vector machine models were used to analyze MRI brain scan images and classify whether Alzheimer's Disease (AD) was present and at what severity. After testing multiple approaches and developing a custom Synthetic Minority Oversampling Technique (SMOTE), the models were able to detect the existence of AD with 98% accuracy, 98% precision, and 98% recall and were able to classify the severity of the AD with 99% accuracy, 99% precision, and 99% recall. This suggests image classification is a strong contributor to Alzheimer's Disease and medical treatment/research as a whole. The potential applications include: automated diagnosis of conditions, classifying the stage of a condition for determining treatment, and tracing phylogenetic indicators to identify the underlying genetic factors of a condition.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

AD	Alzheimer's Disease
MRI	Magnetic Resonance Imagine
SVM	Support Vector Machine
SNP	Single Nucleotide Polymorphism
OASIS	Open Access Series of Imagine Studies
ADNI	Alzheimer's Disease Neuroimaging Initiative
CNN	Convolutional Neural Network
SMOTE	Synthetic Minority Oversampling Technique
NINCDS-ADRDA	National Institude of Neurological and Communicative Disease and Stroke/Alzheimer's Disease and Related Disorders Association
DSM-III-R	Psychiatry Diagnostic & Statistical Manual of Mental Disorders—3rd Edition Revised

Chapter I - Introduction

Alzheimer's Disease (AD), named after the German psychiatrist Alois Alzheimer, is a brain disorder that affects over 5 million Americans (Alzheimer's Association, 2014) and is especially pervasive among the elderly. The Alzheimer's Association states that 11% of those 65 and older, 32% of those older than 85, and 82% of those older than 75 have AD (Alzheimer's Association, 2014). Being such a pervasive condition among one of the more vulnerable demographics of United States citizens has regularly put AD in the spotlight and has made is a regular recipient of research and experimental treatments (Jellinger, 2006).

Now, as the technological age matures and, along with it, computational power has grown beyond what anyone a decade ago could have imagined, a new age of data driven research has begun, with medical research at the forefront. Several papers could be written on the different ways data science can benefit medical research, but one way, which is also one of the more daunting techniques, is image classification. It is often the case where the most daunting tasks become the most important and rewarding, so it is with image classification. Already in the complex field of machine learning, image classification adds additional hurdles and complications that can frustrate even the most patient scientist. To counteract this, the benefits of image classification can match its frustrations.

Countless different sets of images are used in healthcare to examine and diagnose patients, speeding up this process without losing any accuracy would not only significantly help overworked and overwhelmed doctors, nurses, and techs, but it also has the potential to help save lives. As this project will show, machine learning has an important part to play in healthcare and medical image classification is at the forefront of this.

Using a collection of CT brain scan images, two models, a neural network and support vector machine (SVM), were developed to both identify whether Alzheimer's Disease is present within the scan and to gauge the severity of the condition.

The potential benefits of quickly assessing a large number of brain scans for AD and its severity are near endless. As Wang et. al. (2012) show in their association study between phenotypic markers and AD—r

elated single nucleotide polymorphisms (SNPs), it is very possible to use the measuring and tracking of a brain's AD development via brain scan to assist in the identification and tracing of SNPs that could ultimately lead to the development of gene therapy treatments for AD. This, coupled with Shen et. al. (2020)'s work combining and analyzing multiple types of data relating to AD, neuroimaging included, establishes strong causal relationships, shows the very far-reaching applications image classification can have in regards to Alzheimer's Disease, let alone medical research as a whole.

CHAPTER II - BACKGROUND AND LITERATURE REVIEW

Image Classification Use Case

The introduction briefly touches on just how beneficial image classification can be to medical research and treatment, but a deeper examination of an image classification use case will help establish image classification as a truly viable medical research technique. The use case to be examined is an image classification program created by K. Somasundaram and T. Kalaiselvi that leverages maxima transformation techniques to separate brain tumors from the brain in magnetic resonance imaging (MRI) brain scans; this program assists surgeons in identifying where the brain ends and tumor begins to more easily remove the tumor during brain surgery (Somasundaram, 2010). Prior to this work, brain tumor detection techniques fell into three broad categories and many of the techniques required a human hand to implement or verify (Somasundaram, 2010, p. 136). Somasundaram and Kalaiselvi's technique takes advantage of the "bilateral symmetry property" of the human brain, both sides of the brain are symmetrical down a vertical axis, to identify differences between the two sides, extract the parts of the image that can be identified as part of the brain or skull, and leave just an image of the brain tumor. They used an H-maxima transformation which, according to Somasundaram (2010), has its basis in morphological reconstruction, which involved repeated dilations of an image and ends with masking of the image (p.138). The results of this program, as seen in Figure 1, provide a surgeon with not only the location of the tumor, but much more importantly a more precise shape of the tumor. This can provide surgeons with much needed additional information during the planning stage of their surgery, as knowing exactly where to look and what they are looking for will help keep invasiveness and surgery time to a minimum.

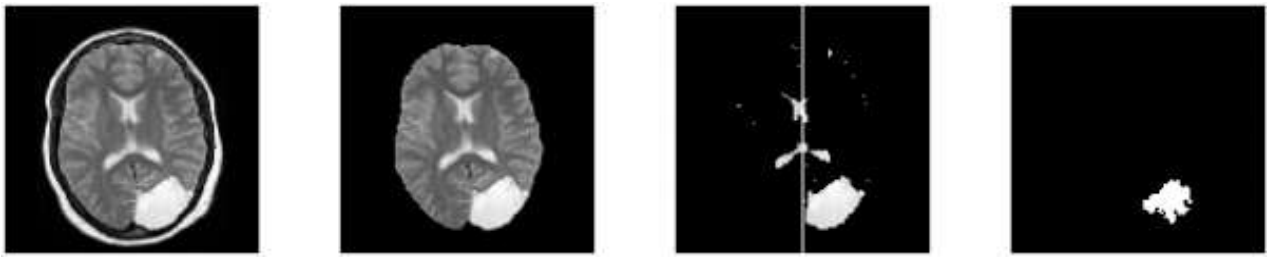

Figure 1. Stages of identifying brain tumors in MRI scans. Original image (far left), to identifying just the brain, to extracting key portions of the brain, then extracting only the image of the tumor (Somasundaram & Kalaiselvi, 2010).

Identifying Alzheimer's Disease via MRI Brain Scan

Somasundaram et. al. (2010)'s work provides a strong basis for image classification as a whole, but just because a technique works for one problem does not mean that will be transferrable to other problems. In this case, will image classification work for identifying Alzheimer's Disease?

Kloppel et. al. (2008) help elucidate the answer to this question with their work aiming to differentiate various brain conditions, including Alzheimer's Disease, from each other and normal brain degradation, using MR scans. Utilizing a linear SVM, Kloppel used scans of 120 subjects with normal cognitive function, varying degrees of AD, age-related brain degredation, and subjects with frontotemporal lobar degeneration, a form of dementia that atrophies the frontal and temporal lobes (van der Zee & Van Broeckhoven, 2014), to develop a model that works to separate AD from what may be another brain condition or the normal brain lifecycle. After these datasets and their related control images trained the SVM, the model provided an average of 91.1% accuracy, 90.1% sensitivity, and 90.7% specificity. According to Kloppel et. al. (2008), these results are equal to or better than other classification techniques that analyze MR images (p. 686) and they also match or outperform typical clinician accuracy, using methods outlined in NINCDS-ADRDA or DSM-III-R (p. 687).

This work makes a very strong case that there is a clear correlation between Alzheimer's Disease development and the patient's MRI brain scan results, allowing for the analysis of these images to identify the condition. Coupling this with the previous use case and its consistent results in identifying brain tumors via MRI scans, and it becomes very clear that image classification on brain scans to identify the severity of Alzheimer's Disease is an extremely viable prospect.

Varying Approaches to Image Classification

With the necessary groundwork laid for establishing the validity of image classification on MRI brain scans to identify the severity of Alzheimer's Disease, it is now important to review the various approaches to image classification one can take. This will make clear why the methods outlined in this paper were ultimately chosen and why their results are both valid and reliable.

In their thorough survey of various (over 30) image classification techniques, Lu et. al. (2007) show that many factors go into deciding the best techniques, including the source of the data and what techniques are readily available, and even with these factors accounted for, each technique has its own pros and cons (p. 827). In the case of the AD detection and grading, the images in the dataset are uniform, consistent in size, orientation, and scale, and colored on a linear scale, rather than with unrelated colors (see Figure 2). Because of these factors, a per-pixel technique would be most appropriate for analysis. This leaves a handful of viable techniques, once techniques not easily available for programmatic implementation are eliminated: neural network, SVM, decision tree, nearest-neighbor, and discriminant analysis. Of these, nearest-neighbor and decision tree can be eliminated due to their over-simplistic approaches. Boiman et. al. (2008) make a strong case for the value of using nearest-neighbor for image classification, due to its ability to run strong models without training the model on a set of training data. But even they acknowledge that, when available, the "state-of-the-art" techniques with training stages will outperform nearest-neighbor. Discriminant analysis, on the other hand, is a robust technique, but requires assumptions that cannot be easily made for this dataset and pose risks for getting reliable results. In particular, discriminant analysis requires that gaussian distribution is assumed and, according to Lu & Weng (2007), discriminant analysis struggles greatly to integrate spatial and contextual attributes (p. 830). Because of the spatial specific nature of AD degradation, discriminant analysis becomes a risky technique to employ. This leaves neural networks and SVMs as the most viable techniques for AD classification. Reviewing the background and details of both these techniques will confirm their value as image classification models.

Support Vector Machines

Support vector machines, or support vector networks, as their creators Corinna Cortes and Vladimir Vapnik refer to them, are, according to the authors' landmark 1995 paper, designed build a linear decision surface by mapping non-linear input vectors to a large feature space (Cortes & Vapnik, 1995, p. 274). This perfectly describes per-pixel image classification, if different pixels and the collection of pixels are imagined as the feature space and a matrix of pixels is a non-linear input vector.

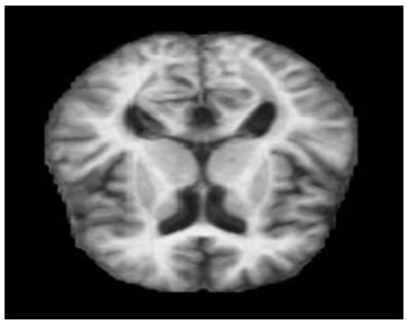

Figure 2. Example of the MRI brain scan images being used for image classification. Note how the coloring is linear rather than using various different colors.

To understand how an SVM functions, it is first important to understand its overarching goal. In this case, an SVM is attempting to define a "margin" between two classes that can be considered the largest separator between the classes, then using that margin to tell those two classes apart (Figure 3). This is done in five primary steps (Figure 4). First, the support vectors (training data) establish a "decision function" that will be used to determine the largest margin between classes. The decision function can be mathematically defined as:

$$I(z) = sign\left(\sum_{support\ vectors} \alpha_i z_i \cdot z + b_0\right) \ldots (\text{Eq. } 1)(\text{Cortes \& Vapnik, } 1995)$$

where z is the feature space. Then the input vector is given to the machine (in this case a matrix of pixel values). Third, the transformation on the non-linear input vector is performed, allowing for the input vector to be compared against the support vectors in step four. After step four's comparison, using the predefined decision function, the input vector can be classified in step five (Cortes & Vapnik, 1995, p. 276-277).

Neural Networks

With sufficient background into SVMs and their strong case for being the binary classifier of choice laid out, it is important to now do the same for neural networks. Neural networks are considered by many to be the premier technique for large-scale image classification (Simonyan & Zisserman, 2017). This position is even further bolstered by a convolutional neural network (CNN) winning The Imagenet Large Scale Visual Recognition Challenge (a competition to most effectively classify images from a famous dataset of 1.2 million images and 1000 classes) by 10% (Krizhevsky, 2012). The reason neural networks, and specifically convolutional neural networks, which is what this project employs, are so successful in image classification is the combination of a neural networks general ability process many features from large datasets quickly and a CNN's added convolution and max pooling layers to extract key features from a very large feature space (such as a matrix of pixel values).

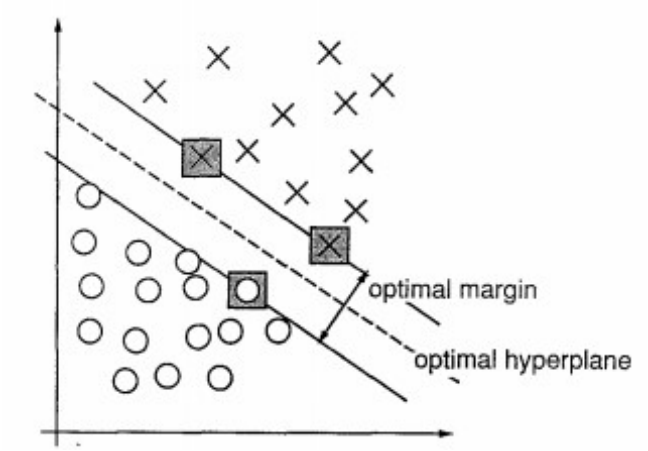

Figure 3. Visual example of how an SVM will establish an "optimal hyperplane" between two classes, with the gray boxes representing the borders of the widest margin (Cortes & Vapnik, 1995).

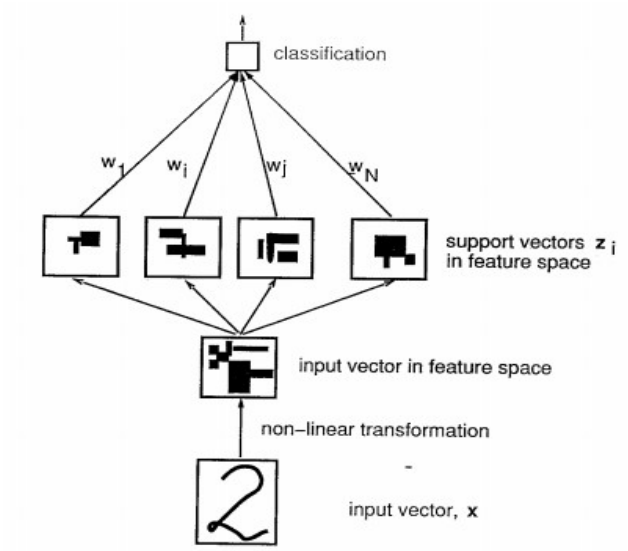

Figure 4. Process by which an input vector is transformed, compared against the support vectors, and classified (Cortes & Vapnik, 1995).

Neural networks operate on three primary layers: the input layer, the hidden layers, and the output layer. These layers are full of "nodes" that have their own rules and activation functions that treat the data differently[19]. It is the proper combination of input, hidden, and output layers that allow neural networks to perform high level functions and transformations on large sets of data at a computationally acceptable rate. Convolution then, as explained by Kalchbrenner et. al. (2014), is a series of mathematical functions between a vector of weights and a vector of inputs. This can be considered simply as a series of transformations that alter the data as to provide the CNN with a new perspective and potentially new features to extract. Convolutional layers, combined with max pooling layers, which aim to reduce the dimensionality of the input, preventing overfitting, result in a very strong model for image classification, especially for multi-class datasets. Figure 5 visualizes this process, showing the sequence convolutional and max pooling layers and ultimately the dimensionality reduction that results.

CHAPTER III - DATA

Data Source Overview

A dataset of 6,400 T1-Weighted MRI brain scan images of AD patients and non-AD patients were compiled from various online sources, including the Open Access Series of Imaging Studies (OASIS). These images were labeled based on the context provided by the source or professional validation into four classes: non-demented, very mild demented, mild demented, and moderate demented. Figure 2 provides an example of one of the T1-Weighted MRI images, which is one of the standard brain scan types, providing a clear, top-down view of the brain anatomy and general brain development. The images for this dataset are consistent in size and orientation, which will make data preparation more streamlined. Label distribution can be observed in Figures 6 and 7. Two things became clear by examining the label distribution: one, the even split between positive AD and negative AD images will serve nicely for identifying those images with AD, and two, the extreme distribution of the four classes of AD severities will make the risk of overfitting the CNN model very present. The first thing to do to mitigate this risk is to eliminate the non-demented samples entirely. This is done by employing the aforementioned SVM model

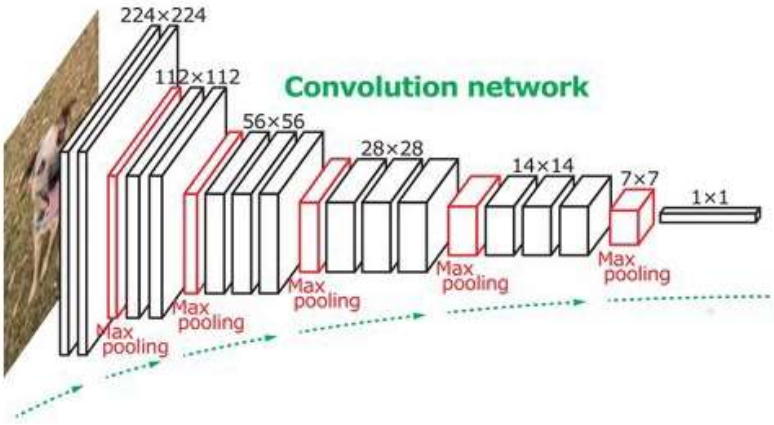

Figure 5.Visual representation of a CNN using convolutional layers combined with max pooling layers to extract features and reduce dimensionality (Ng, 2020).

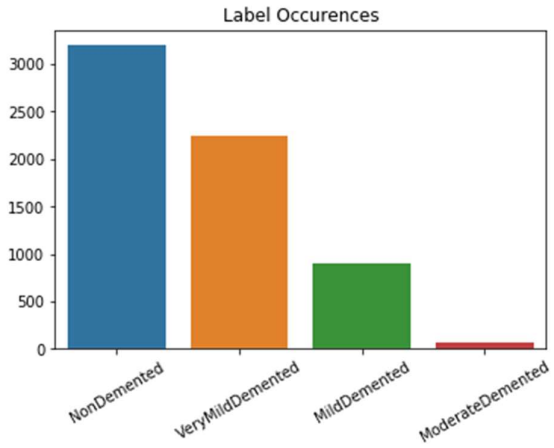

Figure 6. Label breakdown of the images available in the dataset. 3,200 NonDemented, 2,240 VeryMildDemented, 896 MildDemented, and 64 Moderate Demented.

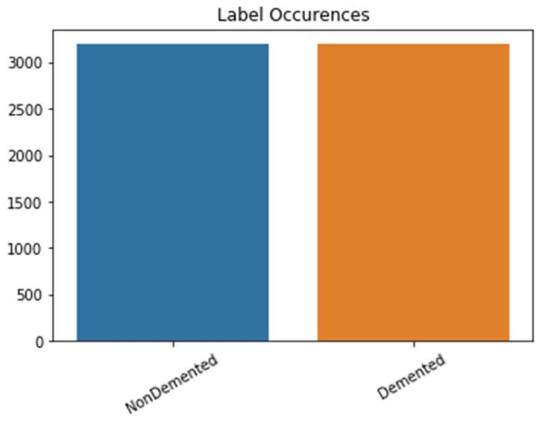

Figure 7. Binary label breakdown of the images in the dataset. Both NonDemented and Demented have 3,200 images.

first, to identify if an image is positive for AD, then only send the positive images to the CNN, which will be trained on only positive images. The second, more technical way of dealing with this distribution of data is oversampling, a technique by which you create additional samples for the underrepresented classes using various approaches. The approach to be used in this case is the synthetic minority oversampling technique or SMOTE.

SMOTE

Originally introduced in Chawla et. al. (2002)'s landmark paper, SMOTE provides an effective new approach to improving skewed datasets and producing more robust models as a result. The technique is very simple, but quite effective: for classes that are under-sampled in the dataset, find *n* nearest neighbor samples of one of the items and produce a new "synthetic" sample by using means of the neighbor samples (Chawla, 2002). This creates a new sample, that still has all the key features of that class, allowing models to identify those features with greater precision due to the increased volume of samples.

Chawla et. al. (2002) tested their technique on eight datasets of varying size and content; the results of testing for all the datasets were that using the combination of SMOTE and majority over-sampling improved loss and accuracy over using no oversampling technique or just using majority under-sampling. In conclusion, SMOTE improved the ability to classify any under-sampled classes (p. 339). These results, combined with the strong reception of SMOTE by the data science community at large (Fernandez, 2018), make it a strong candidate to solve the AD dataset's under-sampling issue. To implement it, a custom python library was built, employing numpy's robust array functions and Euclidean distance as the distance measure. This custom library allowed for multiple parameters to be set by the user, including number of neighbors to be used and ratio of under-sampled datasets to create. This allowed for testing a wide variety of combinations, until the optimal over-sampling parameters were defined.

Data Preparation

The data went through two stages of preparation prior to modeling. First, the data was extracted from folders and better organized for streamlined loading. This was done in python by looping through the

series of folders that held the data, reading the images in these folders into numpy arrays, then labeling them according to the name of the folder they were pulled from. After this, the data and their related labels were stored together in a dictionary, which was then randomized to ensure random sampling in the future. Finally, the images, labels, and binary labels were stored into three .npy files (numpy's unique file format), which allowed the data to be quickly loaded or transferred, without performing all the loading steps again.

After the data is prepared for loading, a series of transformations is performed on the data to prepare it for the SVM and CNN. For the SVM, every pixel is divided by 255, to convert the values from being between 0-255 to between 0 and 1. Next the data is flattened from a three-dimensional array to a one-dimensional array, allowing it to be read by the SVM. Finally, the data is split for training and testing, with 25% of the data (1,600 images), being reserved for testing and 75% for training. For the CNN, a similar process is followed, with the pixel values first being divided by 255, but then the data is not flattened. This is because CNNs are designed to receive more complex arrays, allowing them to maintain their original form. The data is then split into train and test sets, with the label sets then being converted to categorical values with keras' to_categorical function. This is because the CNN will be predicting which of three labels the image is, so it needs the labels in the format [0,1,0], where "1" is the correct label, instead of a single value.

CHAPTER IV - METHODOLOGY

The core of this project was done in the Python programming language. The SMOTE, data evaluation, and data loading libraries were made exclusively in Python, using the core Python library along with some of the more popular imported libraries, including numpy and pandas. Data analysis and the testing of both the SVM and CNN were done in Python via Jupyter Notebooks. The SVM was built using scikit-learn's LinearSVC library. The CNN was built using keras with a TensorFlow backend, running on a NVIDIA 1080 Ti GPU. All visualizations were created using matplotlib and seaborn.

CHAPTER V - RESULTS

SVM Binary Classification

After all necessary transformations were completed, the first set (the training data) was trained on the SVM via scikit-learn's LinearSVC model. This model uses the typical loss function for SVMs hinge loss function. After training, which took roughly five minutes, and predicting the labels of the test dataset, a custom evaluator library, which takes the predictions from the test data and returned the most common evaluation statistics along with a confusion matrix of the predictions, displayed the results of the modeling.

The model performed extremely well, with an accuracy of 0.98, F1-score of 0.98, recall of 0.98, and a precision score of 0.98. Figure 8, the confusion matrix, provides additional insights into the performance of the model. In particular, the confusion matrix makes clear that all 30 misclassified values are false negatives, images that had AD but were predicted otherwise. Examining these errors individually revealed that all 30 were labeled as "VeryMildDemented" in the multi-class dataset. This suggested that the line could blur between a brain scan with AD and a healthy brain when the disease is in its earliest stages, but the fact that only 30 of the 2,240 VeryMildDemented brain scans were misclassified reassured that there is predictive power in examining brain scans with early-onset AD.

CNN Multi-Class Classification

To train and test the CNN required a somewhat different approach than the SVM, because incorporating the synthetic samples via the custom SMOTE library was an important additional step. When SMOTE is implemented can be just as important as implementing it at all, so a deliberate plan of implementation is required. Nick Becker, a data scientist at NVIDIA, one of the leading computer GPU companies and a data analytics leader, provides a strong example as to when in your model SMOTE should be implemented. To summarize, it is important to perform SMOTE only after the data is split into training and testing sets and on the training set only. This prevents pieces of the data the model trained on from being used in the validation set, even if it is not the exact same sample (Becker, 2016). With this in mind, the process for the CNN was: immediately after the train/test split, implement SMOTE on the training set, then incorporate the synthetic samples into the training set as a whole. After this, the model is trained on

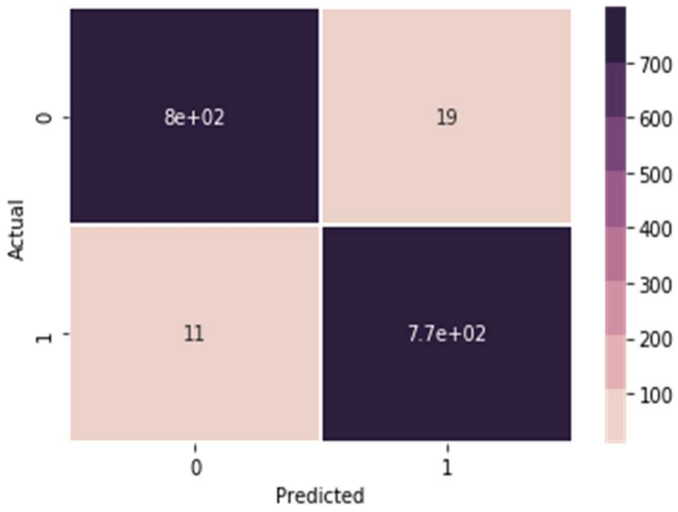

Figure 8. Confusion matrix of SVM model predicting whether AD is present in the test dataset.

the new training data, then evaluated on the test set, just as the SVM was.

The design of the CNN follows the design principles proposed by the pioneer of CNNs, Yann LeCun, in his 1995 paper on the topic: several matching convolutional layers, followed by a max pooling layer, and then a dropout layer to help avoid overfitting is the core of the CNN, with that pattern repeating, three times in this case, to begin the CNN. The layers end with a flattening layer followed by two dense layers, to ensure the output of the model is the same shape as the number of classes being predicted. Each convolution layer uses the ReLu activation function, which has the advantage of being non-linear, like the dataset, and being a sparse activation function, meaning it chooses to regularly not fire certain neurons, greatly increasing training speed (Glorot, Bordes, and Bengio, 2011). This is important when classifying images, as the dataset and the samples within can be very large and complex and thus greatly increase computational power needed to train them. A sparse activation function helps mitigate this by consistently holding neurons back, saving the computing power that would have otherwise gone to them. The CNN ends with a softmax activated layer, which is necessary for multi-classification problems, as it will return the confidence score for each class, i.e. the possibility of the image being each of the labels.

The results of the CNN evaluations, after training on the regular dataset with the added synthetic samples, was equally, if not more, positive than that of the SVM's results. The final loss and accuracy of the model after training were 0.029 and 0.994 respectively; Figures 9 and 10 show the historical loss and accuracy while training. Evaluations of the test dataset were equally encouraging. Accuracy was 0.99, F1-score: 0.99, recall: 0.99, precision 0.99. The confusion matrix (Figure 11) showed only one potential pattern of error, that being all errors were MildDemented and VeryMildDemented being confused with each other. This, like the errors in SVM model, makes sense since these two classes would be most similar. What is most encouraging is that all Moderate demented images, the sparsest class of data, were correctly classified, suggesting that overfitting was successfully avoided. No other error patterns were able to be identified, suggesting that any other errors would be random in nature.

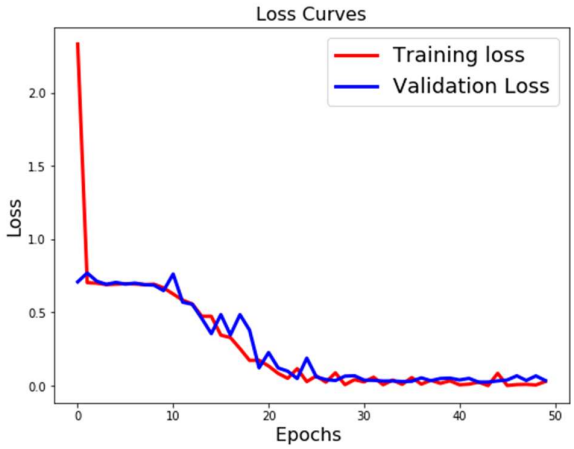

Figure 9. Historical loss of the CNN.

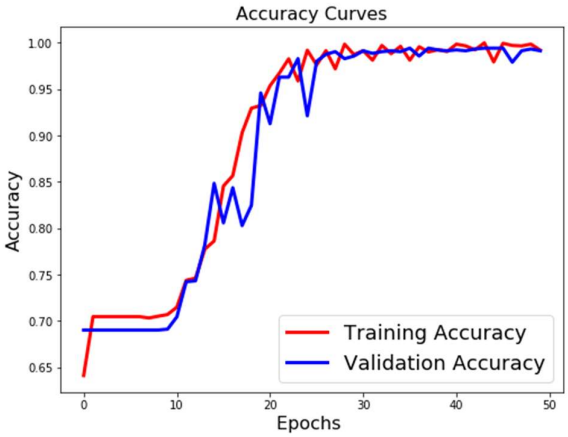

Figure 10. Historical accuracy for the CNN.

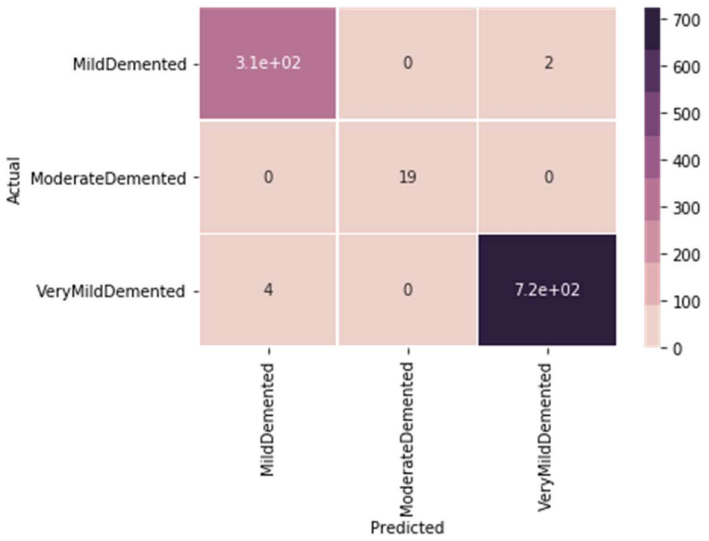

Figure 11. Confusion matrix of the CNN's predictions on test dataset.

CHAPTER VI - CONCLUSION AND PERSPECTIVES

The overwhelmingly positive results observed throughout this project are both very heartening and encouraging for AD research and image classification as a whole. A strong machine learning pipeline that first identifies AD in patients, then grades the severity of it can serve as a strong assistor in research and treatment. Despite the clear positives of the project, it is important to also consider the next steps and areas for improvements. Perhaps the biggest area for improvement is the dataset. A model is only as good as the data it is given and while 6,400 images is by no means a small dataset, it is also unreasonable to be satisfied with the results of only 6,400 images, all of which were thoroughly cleaned and organized. More data, with less consistent parameters, like a live stream of data directly from an MRI machine, would be a great next step for both challenging these models and to refine the data preparation process to look out for and handle inconsistencies in the data. Another big next step would be in regards to the custom SMOTE library, which based on the literature available, is one of the first attempts at utilizing SMOTE for creating synthetic images. While the results here are very encouraging for this practice, one trouble area during the building of the library was defining what is worth considering a "neighbor", since just because an image is the *nth* closest image, does not mean it is similar enough to be a valuable addition to a collection of neighbors. Currently, the library uses a maximum Euclidean distance to filter these images out, which was successful for this dataset, but may be too unrefined a solution for messier datasets. A new approach that identifies key characteristics of a specific image, then defines viable neighbors as those images that are within range and display these key features seems like a potentially strong solution that will need a fair amount of testing to confirm.

Regardless of areas of improvement or potential hurdles, it is worth reemphasizing how positive of a sign this is for medical research at large. Imaging has been and will continue to be a huge area of focus for medical doctors and researchers and the ability to quickly identify the key characteristics of these images, whether they be MRI scans, blood smears, or one of the many other forms of imaging performed in medical fields, can accelerate research exponentially and potentially save lives.

Alzheimer's disease research in particular, which according to the National Institute on Aging, has put a lot of focus on neuronal research recently, could be a huge beneficiary. Identifying and classifying the severity of AD, while important and still being steadily researched by teams like Sperling et. al. (2011), is only part of the future of AD research. Combining those phenotypic signs with the tracing of neuronal pathways as AD develops in a patient could be the frontier where the next big AD breakthroughs occur, as the dots are connected on which specific neurons are affected at each stage of AD and why.

REFERENCES

[1] 2014 Alzheimer's disease facts and figures. (2014). *Alzheimer's and Dementia,* 10(2). doi: 10.1016/j.jalz.2014.02.001

[2] Alzheimer's Disease Fact Sheet. (2019, May22). Retrieved from https://www.nia.nih.gove/health/alzheimers-disease-fact-sheet

[3] Jellinger, K. Alzheimer 100 – highlights in the history of Alzheimer research. *J Neural Transm* 113**,** 1603–1623 (2006). https://doi.org/10.1007/s00702-006-0578-3

[4] Kloppel, S., Stonnington, C.M., Chu, C., Draganski, B., Scahill, R.I., Rohrer, J.D., Frackowiak, R.S.J. (2008). Automatic classification of MR scans in Alzheimers disease. *Brain, 131*(3), 681-689. Doi: 10.1093/brain/awm319

[5] Wang, H., Nie, F., Huang, H., Yan, J., Kim, S., Nho, K, … Shen, L (2012). From phenotype to genotype: an association study of longitudinal phenotypic markers to Alzheimers disease relevant SNPs. *Bioinformatics, 28*(18), i619-i625. Doi: 10.1093/bioinformatics/bts411

[6] Shen, X., Ma, S., Vemuri, P., & Simon, G. (2020). Challenges and Opportunities with Causal Discovery Algorithms: Application to Alzheimer's Pathophysiology. *Scientific Reports*, *10*(1). Doi: 10.1038/s41598-020-59669-x

[7] Boiman, O., Shechtman, E., & Irani, M. (2008). In defense of Nearest-Neighbor based image classification. *2008 IEEE Conference on Computer Vision and Pattern Recognition*. Doi: 10.1109/cvpr.2008.4587598

[8] Jack, C., Bernstein, M., Fox, N., & Thompson, P. (2008). The Alzheimer's disease neuroimaging initiative (ADNI): MRI methods. *JMRI*, *27*(4), 685–691. doi: https://doi.org/10.1002/jmri.21049

[9] Lu, D., & Weng, Q. (2007). A survey of image classification methods and techniques for improving classification performance. *International Journal of Remote Sensing*, *28*(5), 823–870. doi: 10.1080/01431160600746456

[10] Simonyan, K., Vedaldi, A., & Zisserman, A. (2017). Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. Retrieved from https://arxiv.org/pdf/1312.6034.pdf

[11] Somasundaram, K., & Kalaiselvi, T. (2010). Automatic Detection of Brain Tumor from MRI Scans Using Maxima Transform. *Image Processing*.

[12] Sperling, R., Aisen, P., & Laural, B. (2011). Toward defining the preclinical stages of Alzheimer's disease: Recommendations from the National Institute on Aging-Alzheimer's Association workgroups on diagnostic guidelines for Alzheimer's disease. *Alzheimer's and Dementia*, *10*(2), 280–292. doi: https://doi.org/10.1016/j.jalz.2011.03.003

[13] Thai, L. H., Hai, T. S., & Thuy, N. T. (2012). Image Classification using Support Vector Machine and Artificial Neural Network. *International Journal of Information Technology and Computer Science*, *4*(5), 32–38. doi: 10.5815/ijitcs.2012.05.05

[14] Becker, Nick. The Right Way to Oversample in Predictive Modeling. *Github.io*, 23 Dec. 2016, beckernick.github.io/oversampling-modeling/

[15] van der Zee, J., Van Broeckhoven, C. (2014) Frontotemporal lobar degeneration, building on breakthroughs. *Nat Rev Neurol* 10**,** 70–72. https://doi.org/10.1038/nrneurol.2013.270

[16] Cortes, C., Vapnik, V. (1995) Support-vector networks. *Mach Learn* 20**,** 273–297. https://doi.org/10.1007/BF00994018

[17] A. Krizhevsky, I. Sutskever, and G. Hinton. (2012) Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems* 25.

[18] Kalchbrenner, Nal, et al. (2014) A Convolutional Neural Network for Modelling Sentences. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, doi:10.3115/v1/p14-1062.

[19] A Basic Introduction To Neural Networks. *University of Wisconsin*, pages.cs.wisc.edu/~bolo/shipyard/neural/local.html.

[20] Ng, James. Image Classification With Convolutional Neural Network. *Medium*, Towards Data Science, 2 May 2020, towardsdatascience.com/using-artificial-neural-network-for-image-classification-9df3c34577dd.

[21] Dubey, Sarvesh. (2019). Alzheimer's Dataset (4 Class of Images). *Kaggle*, 26 Dec. www.kaggle.com/tourist55/alzheimers-dataset-4-class-of-images.

[22] Chawla, N. V., Bowyer, K., Hall, L., Kegelmeyer, W. (2002). SMOTE: Synthetic Minority Over-Sampling Technique. *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357., doi:10.1613/jair.953.

[23] Fernandez, Alberto, Garcia, S., Herrera, F., Chawla, N. (2018). SMOTE for Learning from Imbalanced Data: Progress and Challenges, Marking the 15-Year Anniversary. *Journal of Artificial Intelligence Research*, vol. 61, pp. 863–905., doi:10.1613/jair.1.11192.

[24] LeCun, Yann, Bengio, Yoshua (1995). Convolutional Networks for Images, Speech, and Time-Series. *The Handbook of Brain Theory and Neural Networks*, MIT Press

[25] Glorot, Xavier, Bordes, Antoine, Bengio, Yoshua (2011). Deep Sparse Rectifier Neural Networks. *AISTATS*