Alzheimer's Disease Detection using Artificial Intelligence

by Abdulhamit Subasi

Submission date: 24-Feb-2021 12:04PM (UTC+0200)

Submission ID: 1516923973

File name: 95848_Abdulhamit_Subasi_Alzheimers_Disease_Detection_using_Artificial_Intelligence_1076597_1631855140.docx (328.65K)

Word count: 5283

Character count: 28165

Chapter :

Alzheimer's Disease Detection using Artificial Intelligence

Abdulhamit Subasi, Manav Nitin Kapadnis, Ayse Bulbul, Jani Petteri Saunavaara, Pekka Ruusuvuori

Institute of Biomedicine, Faculty of Medicine, University of Turku, Turku, 20520, Finland
E-mail: abdulhamit.subasi@utu.fi

Department of Computer Science, College of Engineering, Effat University, Jeddah, 21478, Saudi Arabia.
E-mail: absubasi@effatuniversity.edu.sa

Abstract

Artificial Intelligence (AI) uses statistical theory to create models from data samples. Once a model is created, its interpretation and algorithmic solution for understanding need to be competent as well. Biomedical images relevant to several diseases are generally employed to diagnose precise physiological or pathological conditions. The objective of biomedical image analysis is exact modeling by using AI algorithms for the diagnosis of diseases. This chapter explains how AI approaches are utilized in Alzheimer's disease detection. Alzheimer's disease is one of the most widespread dementia forms affecting the elderly population. On-time detection of Alzheimer's Disease (AD) is crucial to discover new approaches for AD treatment. AI is an efficient approach for AD detection since it can be utilized as a computer-aided diagnosis (CAD) system in clinical practices and play a critical role to detect changes in the brain images to identify AD. This chapter presents the recent studies and advances in AI used for medical image analysis and image processing in AD detection. The main focus is to have a consistent but simple and fast model for automated AD detection based on the application of artificial intelligence techniques. In this chapter, the concentration will be on AI techniques for AD detection from brain images. Moreover, some of the image processing, feature extraction, and artificial intelligence techniques, which were utilized for AD detection is overviewed. Then a simple AD detection approach using deep learning models will be presented. The chapter will be concluded with a summary of the current state-of-the-art, a discussion of new directions and open challenges for future research. The results show that CNN achieved a testing accuracy of 95.70% accuracy and a validation accuracy of 99.71% for the diagnosis of Alzheimer's Disease from Brain MRI images.

Keywords: Alzheimer's disease detection; Artificial Intelligence; Deep learning; Convolutional Neural Networks; Transfer Learning.

1. Introduction

An estimated 5.8 million Americans have Alzheimer's dementia. By mid-century, the number of people living with Alzheimer's dementia in the United States may grow to 13.8 million. Total payments in 2019 for health care, long-term care and hospice services for people age greater than 65 years with dementia are estimated to be $290 billion. (Thies & Bleiler, 2019). Alzheimer's disease destroys brain cells causing people to lose their memory, mental functions and ability to continue daily activities. Initially, Alzheimer's disease affects the part of the brain that controls language and memory. As a result, AD patients suffer from memory loss, confusion and difficulty in speaking, reading or writing. They often forget about their life and may not recognize their family members. They struggle to perform daily activities such as brushing hair or combing tooth (Islam & Zhang, 2018). There are three major stages in Alzheimer's Disease - very mild, mild and moderate. Detection of Alzheimer's Disease (AD) is still not accurate until the patient reaches a moderate AD. Early detection and classification of AD are critical for proper treatment and preventing brain tissue damage. In recent years, doctors are using brain Magnetic Resonance Imaging (MRI) data for earlier detection of Alzheimer's Disease(Islam & Zhang, 2017). In this work, we propose the use of brain MRI scans for early detection of Alzheimer's Disease. Using our models one can classify a given MRI scan in one of the four classes: Non- Demented, Very-Mild Demented, Mild Demented and Moderate Demented. The MRI reflects the spontaneous blood oxygen level-dependent signal. The MRI data is gained from the continuous state detecting of brain activity over a period of time by magnetic resonance imager which is the common e-health device. It has been widely used to analyze the brain function and cognitive ability, predict and diagnose brain disease(Hu et al., 2016). The contribution of this study is analysis of different machine learning models for highly accurate identification of Alzheimer's Disease from the patients' Brain MRI images.

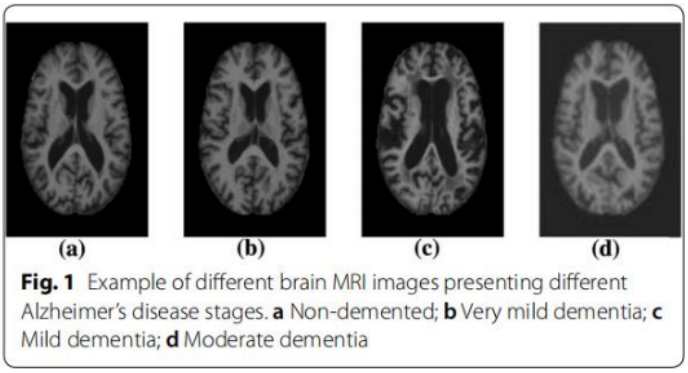

Fig. 1 Example of different brain MRI images presenting different Alzheimer's disease stages. **a** Non-demented; **b** Very mild dementia; **c** Mild dementia; **d** Moderate dementia

2. Background Review

3. Classification Methods for Alzheimer's Disease Detection

3.1 SVM

A linear classification function for a dataset that can be linearly divided and then separated corresponds to a separate hyperplane passing through the middle of the two classes, separating the two. As there can exist a number of hyperplanes that can linearly divide, what SVM additionally guarantees is that by maximizing the margin between the two classes the best such function is found. The margin is defined as the hyperplane that defines the difference between two groups. Geometrically, the margin refers to a point on the hyperplane which stands at between the points that are close and have shortest distance. Having this geometric concept helps one to explore how to opt the margin, so that only a few get selected from a vast variety of planes. As these planes generalize on a greater scale than other planes SVM uses this. This not only provides opportunity for correct classification in the future but also gives high training set accuracy (Wu et al., 2008).

3.2 Random Forest

The random forest classifier represents another modified version of the bagging. It is more justified than boosting, as random forest use bootstrap data samples as training sets for the creation of the base models. Improvement in random forest consists of stimulating the bigger base model diversity by randomizing the modeling algorithm, which is a decision tree algorithm. The algorithm uses two approaches for the creation of the base model: instance sampling and algorithm nondeterminism. Nondeterminism is accomplished by performing a random split choice that is employed for tree growing. Random forest draws a random subset of existing attributes in every node and restricts the succeeding split choosing process to split using those attributes. In order to not have the growing process unchanged, the usual criteria for the split evaluation for decision trees is applied. Stop criteria for decision tree growing are set up to produce comparatively huge, precisely tailored trees and no pruning is applied. Individual overfitting of the models is removed by the aggregation process that makes the random forest ensemble to be highly resistant to overfitting (Cichosz, 2014). developed a framework to preprocess MRI images from the organizers of a neuroimaging challenge by attempting to enumerate the classification performance of MRI features to instantaneously distinguish among AD, MCI, cMCI, and HC. This novel scheme which contains multiple feature selection employing Random Forest from subsets of the whole set of features by employing a fusion method and ensemble classification using majority voting (Dimitriadis et al. (2018)).

3.3 K-nearest neighbour (k-NN)

The k-nearest-neighbor method has been extensively used as a benchmark classifier in the area of AI. This method classifies data by comparing a given test data with the training data similar to it. Every data instance is a point in an n-dimensional space. Thus, all training instances are kept in this space. Once an instance is to be classified, the k-NN looks for the n-dimensional space for the k training instances that are neighboring to the given input. These k training instances are termed “nearest neighbors” of the input instance. The “closeness” of the specified input and

training instances are defined in terms of a distance metric, such as Euclidian distance. The given input instance can be classified as the most common class among its "neighbors" (Han et al., 2011). There are three major elements of this method: a set of training instances that are labeled, a distance between instances, and the value of nearest neighbors k. In order to classify the input instance, its distance from the labeled object is calculated, then the number of nearest neighbors is determined, and finally, the class of nearest neighbors is allocated as the label to the given instance (Wu et al., 2008). Kalbkhani et al. (2013) proposed a robust algorithm for disease diagnosis applied to MRI. In the first step, two-level two-dimensional DWT (2D DWT) of MRIs was computed. In the second step, PCA and LDA were applied to extract the relevant features. In the last step, the extracted features were used as input to k-NN and SVM classifiers individually, so as to find the normal MRIs or MRIs with some disorder. The obtained performance results showed that the suggested approach resulted in high classification accuracy rates and outperforms the newly presented techniques, whereas it takes smaller amount of features for the classification task.

3.4 Artificial Neural Networks (ANN)

Artificial neural networks (ANNs) is a machine intelligence technique of higher interest in various research areas. However, it has the smallest number of operational applications. ANN is capable of producing high prediction accuracy for various types of problems. Originally, it was developed to imitate the function of a human brain in making predictions, detecting patterns, or learning from previous experience. This process is reflected in a computer program, which implements pattern recognition and machine learning algorithms for successful predictive model creation. While constructing the ANN, the analyst needs to decide on how many nodes are optimal for the network, how the nodes are to be connected, what will be the most appropriate weight value for each connection, and how the training is to be conducted. In ANN, the input data is received through the input nodes, whose values are multiplied with the weight values and stored in the connections. The hidden layer is added between the input and output layer; the output of the previous layer is an input for the estimate of the target. The approach that is considered in neural networks is the so-called multilayer. Additionally, a transfer function is applied in the hidden layer, and value of the results is produced. This value, which is a prediction, is passed to the output node. The nodes in ANN could be grouped in input and output layers. The input layer is a group of input variables, while the output layer is a target variable. Unit is every input variable in neural networks that has its own unit. The model can be developed in forward and backward propagation. The backward propagation approach is more common in research studies. However, a disadvantage of the ANN is the complexity of the resulting model. The process of model building is similar to the "black-box." As the complexity of the model increases, additional hidden layers could be added. The result of ANN depends much on how the learning process evolves and what starting values are used in the training stage (Subasi, 2020).

3.5 Deep Learning

The research in Deep learning has been growing fast in the medical imaging field, including Computer-Aided Diagnosis (CAD), medical image analysis, and radiomics. CAD is an expeditiously developing area of research in the medical industry. The recent researchers machine learning guarantee the enhanced efficiency in detection of disease(Deepika Nair et al., 2019). Deep learning algorithms differ from conventional machine learning method by the fact

that they require little or no image pre-processing and can automatically infer an optimal representation of the data from the raw images without requiring prior feature selection, resulting in a more objective and less bias-prone process (LeCun et al., 2015; Vieira et al.,2017). Therefore, deep learning algorithms are better suited for detecting subtle and diffuse anatomical abnormalities (LeCun et al., 2015; Vieira et al., 2017). From previous research in the medical domain, it has been proved that MRI data can perform a significant role for early detection of Alzheimer's Disease. We plan to analyze brain MRI data using deep learning model for Alzheimer's Disease detection and classification (Islam & Zhang, 2017).

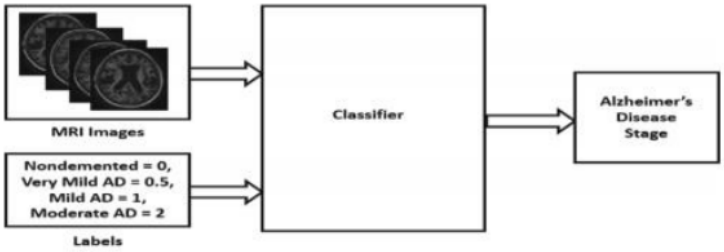

Fig. 2. Diagram of a generic Alzheimer's disease detection and classification framework.

3.6 Convolutional Neural Network

Traditional computer vision methods are composed of three important stages. First stage is feature extraction, second is feature reduction, and third is classification. Nevertheless, scholars combine these stages in standard convolutional neural network (CNN). That means, CNN does not need to set the feature manually. By contrary, the weights of its initial layers served as feature extraction, and their values were obtained by iterative learning. Also, CNN can obtain a better performance than peer classifiers, for example, feedforward neural network and stochastic network. The convolution layer performs feature extraction, and pooling layer performs reduction, and the softmax layer performs classification (Wang et al., 2018). CNN categories, processes, and classifies an image input into those categories. Computers interpret an input image as an array of pixels and this depends on the image resolution. Based on image resolution it can see h x w x d (h = height, w = distance, d = dimension). Regularized models of CNNs are multilayer perceptrons. Multilayer perceptrons usually mean completely linked networks, that is to say, each neuron in one layer is connected to all neurons in the next layer. "Absolute reliability" of such networks makes them vulnerable to overfitting results. Typical methods of regularization involve adding the loss function to some form of weight calculation. CNNs use a particular approach to regularization: they take advantage of the hierarchical structure of data by using smaller and simplified patterns, and generate more complicated patterns. Therefore, CNNs are at the lower end, on the connectivity scale and complexity. Convolution is the first layer for extracting features from an input file. Convolution maintains the relationship between pixels through the use of small squares of input data to learn image properties. It is a mathematical operation containing two inputs such as an image vector, and a filter or kernel. Part pooling layers will reduce parameter numbers when the images become too large. Spatial pooling also known as subsampling or down sampling reduces the dimensionality of each map but retains important information. Fully connected layers in one layer bind each neuron to each neuron in another layer. This is in theory the same as the conventional multilayer perceptron neural network (MLP). To define the images the flattened matrix moves

through a totally connected row. (Albawi et al., 2017)

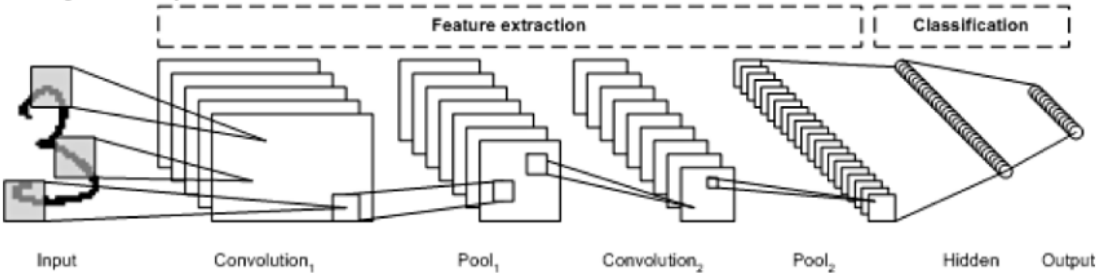

Fig. 3- Mechanism of Convolutional Neural Networks

3.7 Transfer Learning

Deep learning techniques require many examples or a large set of data. The number of samples in our dataset are fewer as compared to modern data standards, i.e. the number of MRI images for the classification function is fewer, so we can go for Transfer Learning. Transfer learning (Hussain et al., 2018; Pan & Yang, 2010) is about taking features learned on one issue, and using them on a different, related issue. Export learning is aimed at improving learning in the target task by incorporating information from the source task. The key facts to test before using Transfer Learning are that the source task should have been trained on a larger dataset than the target task, and the source and target tasks should be identical in nature. There are three common measures which could improve learning through transfer. First is the initial achievable output in the target task utilizing only the transferred information, compared to the initial output of an ignorant person, before any more learning is performed. Second, the amount of time it takes to thoroughly understand the target mission, despite the information transferred, relative to the amount of time it takes to understand it from scratch. Second, the final level of output which can be reached in the target role relative to the final level without transition.

Here we have used different models like VGG16, VGG19, Inception v3, MobileNet, MobileNetV2, DenseNet169, DenseNet121, InceptionResNetV2, MobileNetV2, ResNet101. Simonyan and Zisserman presented the VGG network architecture in their 2014 article, Very Deep Convolutional Networks for Large Scale Image Recognition (Simonyan & Zisserman, 2015). This network is distinguished by its simplicity, using in rising detail just 3 universal layers layered on top of each other. Max pooling works to high volume capacity. In comparison to conventional sequential network architectures like AlexNet, OverFeat, and VGG, ResNet is instead a type of "exotic architecture" based on micro-architecture modules (also called "network-in-network architectures"). The term micro-architecture refers to the collection of "building blocks" that were used to create the network. The ResNet architecture, first presented by He et al. in their 2015 paper Deep Residual Learning for Image Recognition (He et al .,2015), has been a landmark work, demonstrating that incredibly deep networks can be equipped using normal SGD (and a fair initialization function) by using residual modules. The inception module 's goal is to act as a "multi-level feature extractor" by computing 1 bracket, 3 bracket, and 5 brackets within the same network module — these filter outputs are then stacked along the channel dimension before being fed into the next network layer. Google LeNet was the initial implementation of this architecture, but later implementations were simply named Iteration vN where N corresponds to the version number Google brought out. Xception was proposed by

François Chollet, founder and chief maintainer of the Keras library. Xception is an extension of the standard replacing Inception architecture.

4. Alzheimer's Disease Detection Using Artificial Intelligence

4.1 Experimental Data

We use a public dataset for the task which was taken from the following link: https://www.kaggle.com/tourist55/alzheimers-dataset-4-class-of-images . The dataset contains total of 5480 images. We have divided the dataset into three sets namely training, validation and test sets. The Training and Test set split were given by the dataset creator themselves. The training set was further divided into 80% and 20% into Training and Validation set respectively. The following table gives a compact description of class distribution of each of the four classes in the three sets.

	Non-Demented	**Very Mild Demented**	**Mild Demented**	**Moderate Demented**	**Total**
Train	2048	1434	574	41	4097
Val	512	358	143	11	1024
Test	640	448	179	12	1279

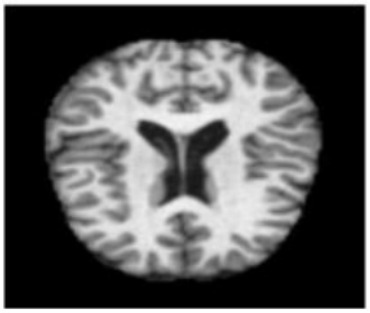

Fig. 3: Non-Demented MRI scan

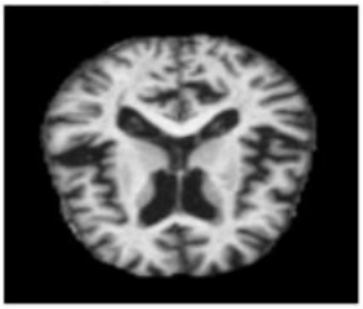

Fig. 4: Very Mild Demented MRI scan

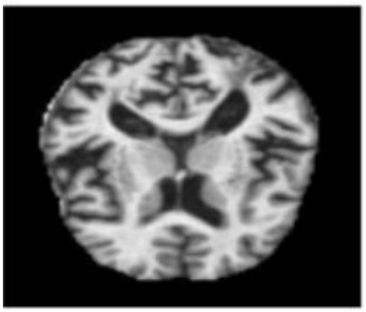

Fig. 5: Mild Demented MRI scan

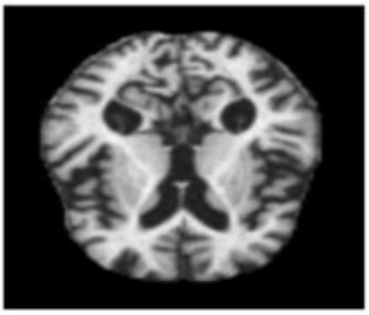

Fig. 6: Mild Demented MRI scan

4.2 Performance Evaluation Measures

The training set performance measures cannot be assumed similar to that of test set. The training set may hold the bulk of examples but the test set is designed to be more similar to the real-world data. For ConvNets, having low amount of data is a great hindrance. The methods used to evaluate the performance measures in such a case are still controversial. The majority of the support is given to the performance measures obtained from the cross-validation sets. The classifier is tuned to provide the best performance measures in case of the training set. The most important fact to have in mind while training is that none of the examples from the test set should be used while designing the classifier. Hence the performance measures from test set can be assumed to be a close comparison to the performance measures on the real-world images. A classifier classifies an image into a category. If the category matches the given category, then it is assumed to be success. If there is any mismatch then it is assumed to be an error. The performance measures are mostly on the basis of error-rate of a classifier.
The training set hold a bulk of examples. The validation dataset should have the distribution almost similar to test set to have the best performance measures. The validation set should be utilized for parameter tuning and the test set for finding the final values of performance measures (Hall et al., 2011).
When vast volumes of data are available, there is no problem: We take a big sample and use it for preparation, and presume that the remainder should be used for processing. In general, the efficiency of a classifier increases with an improvement in the size of the training range, but after a certain stage the efficiency stays constant, and does not change further. (Hall and coll., 2011)
Sensitivity applies to the percentage of individuals with disability who have a positive test result — that is, TP. Specificity refers to the percentage of non-illness people with unfavorable test outcomes. The result from these is also used as a general measure (Hall et al., 2011):

$$Recall = \frac{TP}{TP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} * 100\%$$

Another performance measure in information retrieval is the F-measure, or F1-measure also, which is,

$$F1 - measure = 2 * \frac{Precicison * Recall}{Precision + Recall}$$

Kappa statistic
A measure called the Kappa statistic takes into consideration this expected figure by deducting it from the predictor 's successes and expressing the result as a proportion of the total for a perfect predictor. The mean value of Kappa is 100 per cent, and the predicted value is 0 for a random indicator of the same total column. To sum up, the Kappa statistics are used to measure

the relationship between the predicted and observed categorizations of a data set, while at the same time accounting for an association that occurs by mistake. Nevertheless, it takes no account of losses like the basic performance rate (Hall et al., 2011). The kappa statistic (ÿ) uses the fact that by chance the classifier will simply agree or disagree. The kappa statistics is the most widely used metric for categorical data measurement where there is no objective way of determining the
probability of chance agreement between two or more observers.. A kappa value of 0 is an approximation of chance agreement, while a kappa value of 1 is a total agreement (Viera & Garrett, 2005)(Lantz & Nebenzahl, 1996).Cohen (Cohen, 1960) defined the kappa statistic as an agreement index and defined as the following,

$$K = \frac{Po - Pe}{1 - Pe}$$

Where Po is observed agreement and Pe measures the agreement expected by chance (Yang & Zhou, 2015).

4.3 Experimental Results

The results have been divided into three parts. In the first part we have used only Transfer Learning without using any machine learning algorithm. The classifier trained consists of the base models mentioned in the table below and two 64 units Dense layers followed by three 32 units Dense layer followed by the output Dense layer with 4 units over the pre trained models, between each layer there is a Batch Normalization and Dropout Layers. In the second part we have used Convolutional Neural Networks with Different number of Layers. We assumed a layer to comprise of a convolutional layer and a pooling layer. In the third and the final part, various Machine Learning algorithms were used on the extracted features obtained from different pre-trained models.
The pre-trained models used are: DenseNet121, DenseNet169, InceptionResNetV2, InceptionV3, MobileNet, MobileNetV2, Resnet50, ResNet101, VGG16, VGG19, Xception. The Machine Learning algorithms used are: AdaBoost, ANN, KNN, RF, SVM, and XGBoost. The entire dataset of X-Ray images was divided into COVID 19, Normal and other types
of Pneumonia. The classification task was divided into the following four categories of classification. Accuracy, F1-score, Kappa, Precision and AUC were used as performance measures.

I. Transfer Learning Results

This part contains result for different Transfer Learning Models which are run for 20 epochs with RMSprop as optimizer.

Model	TRAINING ACCURACY	VALIDATION ACCURACY	TEST ACCURACY	F1 Measure	KAPPA	Precision
ResNet 50	77.25	79.08	75.7	0.166	0.3237	0.5833
ResNet 101	76.25	76.56	75.99	0.2043	0	0.5949
VGG 16	99.07	99.51	95.08	0.9506	0.9191	0.9508
VGG 19	98.58	98.73	94.22	0.9422	0.9055	0.9431
MobileNet	79.64	79.74	79.14	0.4822	0.2129	0.6345
MobileNet V2	79.5	78.01	78.22	0.4544	0.0002	0.60728
DenseNet 169	80.56	78.81	79.88	0.4807	0.064	0.6731
DenseNet 121	78.59	78.91	79	0.3778	0.064	0.722
Inception V3	79.52	76.93	77.52	0.2424	0.1014	0.761
InceptionResNet V2	78.72	78.45	78.77	0.4238	0.1402	0.6558

Here, we see that different models show different results with VGG16 and VGG19 outperforming the rest of the models. The other models apart from these two show similar accuracies around ~80%.

II. Results of CNN with different number of Layers

CLASSIFIER	TRAINING ACCURACY	VALIDATION ACCURACY	TEST ACCURACY	F1 Measure	KAPPA	Precision
2 Layers	98.83	98.73	94.14	0.9413	0.9035	0.9414
3 Layers	98.97	99.71	95.7	0.9572	0.929	0.958
4 Layers	98.58	98.63	92.34	0.9239	0.8735	0.9261
5 Layers	98.29	98.05	92.11	0.9207	0.8706	0.9214
6 Layers	74.99	75.02	75.02	0.5004	0	0.5003
7 Layers	74.99	75.02	75.02	0.6667	0	0.5005
8 Layers	74.9	75	75	0.5005	0.057	0.5005

In this part we can see that CNN with 2, 3, 4, and 5 layers classified the MRI images in a very good manner but the subsequent layers failed to do so.

III. Feature extraction with different pre-trained models:

1. **VGG 16**

Model	Train accuracy	Val accuracy	Test accuracy	F1 score	Recall	Precision
ANN	74.44	75.05	75.02	0.0164	0.0086	0.5238
KNN	60.35	42.19	42.73	0.4099	0.4273	0.4036
SVM	50	50	50	0.3333	0.5	0.25
Random Forest	100	43.65	46.02	0.4321	0.4602	0.4219
AdaBoost	49.58	49.51	49.45	0.3866	0.4945	0.3925
XGBoost	91.89	47.56	46.64	0.4257	0.4664	0.4201

Random Forest and XGBoost have higher train accuracy as compared to others, but at the same time these classifiers are seem to be overfitting. ANN on the other hand shows greater test accuracy than all of them, which in turn is deemed to be a good for AD classification.

2. VGG 19

Model	Train accuracy	Val accuracy	Test accuracy	F1 score	Recall	Precision
ANN	75.82	76.61	77.56	0.4869	0.4292	0.5677
KNN	63.21	45.8	44.69	0.4273	0.4469	0.4246
SVM	51.8599	51.559	51.09	0.3758	0.5109	0.4307
Random Forest	100	49.51	46.0899	0.431	0.4609	0.4183
AdaBoost	50.56	51.37	50	0.3952	0.5	0.4534
XGBoost	89.97	49.41	48.52	0.4509	0.4852	0.4457

Here too like in VGG 16 we see that Random Forest and XGBoost have a very high train accuracy as they beat all the models in train accuracy but at the same time these classifiers can be seen overfitting to a great extent. Here unlike previous model all classifiers are overfitting to a certain extent.

3. ResNet 50

Model	Train accuracy	Val accuracy	Test accuracy	F1 score	Recall	Precision
ANN	75	76.2	75.02	0.6667	0.5	0.5004
KNN	64.33	47.56	43.83	0.4208	0.4383	0.4156
SVM	50	50	50	0.333	0.5	0.25
Random Forest	100	48.54	46.56	0.4429	0.4656	0.4346
AdaBoost	43.36	40.62	39.69	0.3913	0.3969	0.3946
XGBoost	91.41	48.05	46.72	0.4408	0.4672	0.4336

Random Forest and XGBoost beat all others like usual in the training accuracy but are seen to be overfitting again. Here ANN can be considered as a good classifier with less overfitting.

4. ResNet 101

Model	Train accuracy	Val accuracy	Test accuracy	F1 score	Recall	Precision
ANN	76.38	77.49	76.11	0.2971	0.2048	0.561
KNN	64.62	46.88	48.59	0.4658	0.4859	0.4603
SVM	50	50	50	0.333	0.5	0.25
Random Forest	100	48.73	48.67	0.4664	0.4867	0.4598
AdaBoost	51.51	50.88	48.91	0.4424	0.4891	0.4437
XGBoost	90.23	49.32	48.05	0.4573	0.4805	0.4508

Random Forest and XGBoost have a very high train accuracy as they beat all classifiers once again. None of the classifiers seem to give good test results.

5. **MobileNet V2**

Model	Train accuracy	Val accuracy	Test accuracy	F1 score	Recall	Precision
ANN	75.53	77.05	74.78	0.374	0.3041	0.493
KNN	61.1299	41.41	42.97	0.407	0.4297	0.4003
SVM	50	50	50	0.3333	0.5	0.25
Random Forest	100	41.99	44.14	0.4135	0.4414	0.4001
AdaBoost	49.02	47.36	48.75	0.3889	0.4875	0.3839
XGBoost	90.11	43.36	46.72	0.4297	0.4672	0.4285

All classifiers are overfitting in this case, except ANN, which shows consistent accuracy of around ~75%.

6. **MobileNet**

Model	Train accuracy	Val accuracy	Test accuracy	F1 score	Recall	Precision
ANN	75.67	76.05	75.06	0.1112	0.0641	0.5093
KNN	60.72	42.97	42.89	0.4028	0.4289	0.3957
SVM	50	50	50	0.3333	0.5	0.25
Random Forest	100	44.04	44.61	0.4098	0.4461	0.3963
AdaBoost	49.27	48.93	49.22	0.3872	0.4922	0.4242
XGBoost	89.33	46.29	47.27	0.4317	0.4727	0.4297

All of the classifiers are overfitting in this case, except ANN, which shows consistent accuracy of around ~75%. Here ANN can be considered a good classifier for AD classification.

7. **Inception V3**

Model	Train accuracy	Val accuracy	Test accuracy	F1 score	Recall	Precision
ANN	74.76	75.07	75.04	0.003	0.0016	1
KNN	60.77	41.21	42.03	0.3978	0.4203	0.3882
SVM	50	50	50	0.3333	0.5	0.25
Random Forest	100	45.61	42.5	0.3987	0.425	0.3859
AdaBoost	49.41	49.41	49.06	0.397	0.4906	0.4176
XGBoost	91.55	45.51	43.83	0.4073	0.4383	0.3998

Here, just like the previous cases, the classifiers are overfitting, apart from ANN, which shows consistent accuracy of around ~75%. Here ANN can be considered a good classifier for AD classification.

8. InceptionResNet V2

Model	Train accuracy	Val accuracy	Test accuracy	F1 score	Recall	Precision
ANN	75	75	75	0.6229	0.4938	0.8935
KNN	62.23	46.39	44.379	0.4257	0.4438	0.4226
SVM	50.83	50.88	50.16	0.3621	0.5016	0.3894
Random Forest	100	46.88	45.39	0.4278	0.4539	0.4151
AdaBoost	47.49	44.73	43.44	0.3927	0.4344	0.3613
XGBoost	91.039	47.85	45.78	0.4259	0.4578	0.4117

In this case XGBoost and Random Forest seem to give good accuracy on train set, but both of them perform poorly on train sets. ANN is a better classifier, as we can see from previous results as well as this.

9. DenseNet 169

Model	Train accuracy	Val accuracy	Test accuracy	F1 score	Recall	Precision
ANN	75.11	76.2	75.02	0.5004	0.5004	0.5004
KNN	60.99	41.21	44.3	0.4178	0.443	0.4123
SVM	50	50	50	0.3333	0.5	0.25
Random Forest	100	42.58	44.92	0.4123	0.4492	0.401
AdaBoost	50.1499	49.02	48.83	0.3871	0.4883	0.4168
XGBoost	88.4	43.55	44.769	0.4147	0.4477	0.4071

Here, just like the previous cases, the classifiers are overfitting, apart from ANN, which shows consistent accuracy of around ~75%. Here ANN can be considered a good classifier for AD classification.

10. DenseNet 121

Model	Train accuracy	Val accuracy	Test accuracy	F1 score	Recall	Precision
ANN	75	75	75	0.6609	0.5023	0.9853
KNN	62.5	42.77	42.97	0.4111	0.4297	0.4025
SVM	50.0199	50.29	50.16	0.3398	0.5016	0.4258
Random Forest	100	46.19	44.84	0.415	0.4484	0.4076
AdaBoost	50.27	49.8	50.08	0.3686	0.5008	0.3889
XGBoost	91.63	46.19	45.39	0.4142	0.4539	0.4021

Here, just like the previous cases, the classifiers are overfitting, apart from ANN, which shows consistent accuracy of around ~75%. Here ANN can be considered a good classifier for AD classification.

11. Xception

Model	Train accuracy	Val accuracy	Test accuracy	F1 score	Recall	Precision
ANN	77.94	78.05	76.06	0.6667	0.3542	0.5317
KNN	61.839	45.019	44.92	0.4254	0.4492	0.42
SVM	50.149	50.29	50.16	0.339	0.5016	0.368
Random Forest	100	46.68	47.81	0.4494	0.4781	0.43485
AdaBoost	51.339	51.559	51.41	0.4318	0.5141	0.4886
XGBoost	91.02	49.32	47.89	0.4392	0.4789	0.4311

5. Discussion

If we compare all the results then we see that for classification of AD into four different classes, no combination of classifiers beats the accuracies of VGG16 and VGG19 (accuracy ~95%) from simple transfer learning. Following these are, 2-layered CNN and 3- layered CNN, which also give great performance measures (accuracy ~94%). These are then followed by 4-layered CNN and 5-layered CNN, which give accuracy around 92%. The rest of classifiers then give moderate accuracies in the range of (75-80%). In this range, the best classifier is simple transfer learning DenseNet 169 model which can be used to produce an accuracy of around 79.88%.

6. Conclusion

In this work, we have presented so[28] results on detecting Alzheimer's Disease into three different stages, a disease which is the fifth leading cause of death among Americans in the age ≥ 65 years(Thies & Bleiler, 2012), from brain MRI scans using different Machine Learning models. Specifically, the report above has compared different types of models ranging from simple Convolutional Neural Networks to Simple Transfer Learning to Feature Extraction and using traditional Machine Learning models over these pre-trained models. Findings indicate that CNN with 2 or 3 layers are able to classify the MRI images well but with increasing layers fail to do so. Simple Transfer Learning also gives promising results with models such as VGG 16, VGG 19 and DenseNet 169. The feature extraction method did not show promising results with most of the Machine Learning classifiers overfitting on the train set, thus giving poor results in test cases. ANN was one such model that did not overfi[20] any case of feature extraction and in turn gave better accuracies than the rest of ML models. For future work, we intend to train the models on more data and to evaluate more architectures for the case of Alzheimer Disease.

Alzheimer's Disease Detection using Artificial Intelligence

ORIGINALITY REPORT

37%	31%	35%	9%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Abdulhamit Subasi. "Use of artificial intelligence in Alzheimer's disease detection", Elsevier BV, 2020 Publication	17%
2	link.springer.com Internet Source	4%
3	Submitted to TechKnowledge Student Paper	2%
4	coek.info Internet Source	1%
5	braininformatics.springeropen.com Internet Source	1%
6	www.pyimagesearch.com Internet Source	1%
7	dokumen.pub Internet Source	1%
8	Chenhui Hu, Ronghui Ju, Yusong Shen, Pan Zhou, Quanzheng Li. "Clinical decision support for Alzheimer's disease based on deep learning	1%

	and brain network", 2016 IEEE International Conference on Communications (ICC), 2016 Publication	
9	"Proceedings of the International Conference on ISMAC in Computational Vision and Bio-Engineering 2018 (ISMAC-CVB)", Springer Science and Business Media LLC, 2019 Publication	1%
10	Submitted to Indiana Wesleyan University Student Paper	1%
11	Abdulhamit Subasi. "Biomedical Signal Classification Methods", Elsevier BV, 2019 Publication	1%
12	Submitted to Arab Academy for Science, Technology & Maritime Transport CAIRO Student Paper	1%
13	medium.com Internet Source	1%
14	www.cse.hcmut.edu.vn Internet Source	<1%
15	Ian H. Witten, Eibe Frank, Mark A. Hall, Christopher J. Pal. "Credibility", Elsevier BV, 2017 Publication	<1%
16	Lalita Moharkar, Sudhanshu Varun, Apurva Patil, Abhishek Pal. "A scene perception system	<1%

for visually impaired based on object detection and classification using CNN", ITM Web of Conferences, 2020
Publication

---

17 Prashant Shukla, Abhishek, Shekhar Verma, Manish Kumar. "A rotation based regularization method for semi-supervised learning", Pattern Analysis and Applications, 2021 <1%
Publication

---

18 Submitted to Middle East College of Information Technology <1%
Student Paper

---

19 Submitted to Harrisburg University of Science and Technology <1%
Student Paper

---

20 Antonios Makris, Ioannis Kontopoulos, Konstantinos Tserpes. "COVID-19 detection from chest X-Ray images using Deep Learning and Convolutional Neural Networks", Cold Spring Harbor Laboratory, 2020 <1%
Publication

---

21 "2017 Alzheimer's disease facts and figures", Alzheimer's & Dementia, 2017 <1%
Publication

---

22 Lei Lu, Ping Wang, Hua Huang. "A Large-Scale Frontal Vehicle Image Dataset for Fine-Grained Vehicle Categorization", IEEE Transactions on <1%

Intelligent Transportation Systems, 2020
Publication

23	Mohamed Hammad, Abdullah M. Iliyasu, Abdulhamit Subasi, Edmond S. L. Ho, Ahmed A. Abd El-Latif. "A Multitier Deep Learning Model for Arrhythmia Detection", IEEE Transactions on Instrumentation and Measurement, 2021 Publication	<1%
24	Mehmet Yaman, Abdulhamit Subasi, Frank Rattay. "Comparison of Random Subspace and Voting Ensemble Machine Learning Methods for Face Recognition", Symmetry, 2018 Publication	<1%
25	insidenothing.blogspot.com Internet Source	<1%
26	Submitted to University of Strathclyde Student Paper	<1%
27	Submitted to Lewis University Student Paper	<1%
28	alz-journals.onlinelibrary.wiley.com Internet Source	<1%
29	"CMBEBIH 2017", Springer Science and Business Media LLC, 2017 Publication	<1%
30	Jyoti Islam, Yanqing Zhang. "Early Diagnosis of Alzheimer's Disease: A Neuroimaging Study	<1%

with Deep Learning Architectures", 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2018
Publication

31	Submitted to essex Student Paper	<1%
32	www.researchgate.net Internet Source	<1%
33	Junhao Wen, Elina Thibeau-Sutre, Mauricio Diaz-Melo, Jorge Samper-González et al. "Convolutional Neural Networks for Classification of Alzheimer's Disease: Overview and Reproducible Evaluation", Medical Image Analysis, 2020 Publication	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		