EARLY DETECTION OF ADHD AMONG CHILDREN USING MACHINE LEARNING

NUR ATIQAH BINTI KAMAL

Bachelor of Computer Science (Software Engineering) with Honours

UNIVERSITI MALAYSIA PAHANG

UNIVERSITI MALAYSIA PAHANG

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name : NUR ATIQAH BINTI KAMAL

Date of Birth :

Title : EARLY DETECTION OF ADHD AMONG CHILDREN USING MACHINE LEARNING

Academic Session : SEMESTER II SESSION 2022/2023

I declare that this thesis is classified as:

- ☐ CONFIDENTIAL (Contains confidential information under the Official Secret Act 1997)*
- ☐ RESTRICTED (Contains restricted information as specified by the organization where research was done)*
- ☑ OPEN ACCESS I agree that my thesis to be published as online open access (Full Text)

I acknowledge that Universiti Malaysia Pahang reserves the following rights:

1. The Thesis is the Property of Universiti Malaysia Pahang
2. The Library of Universiti Malaysia Pahang has the right to make copies of the thesis for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

(Student's Signature)	(Supervisor's Signature)
NUR ATIQAH BINTI KAMAL	AHMAD FAKHRI BIN AB. NASIR
	Name of Supervisor
Date: 24/06/23	Date: 24/06/23

THESIS DECLARATION LETTER (OPTIONAL)

Librarian,
Perpustakaan Universiti Malaysia Pahang,
Universiti Malaysia Pahang,
Lebuhraya Tun Razak,
26300, Gambang, Kuantan.

Dear Sir,

CLASSIFICATION OF THESIS AS RESTRICTED

Please be informed that the following thesis is classified as RESTRICTED for a period of three (3) years from the date of this letter. The reasons for this classification are as listed below.

Author's Name
Thesis Title

Reasons (i)

(ii)

(iii)

Thank you.

Yours faithfully,

____
(Supervisor's Signature)

Date:

Stamp:

Note: This letter should be written by the supervisor, addressed to the Librarian, *Perpustakaan Universiti Malaysia Pahang* with its copy attached to the thesis.

SUPERVISOR'S DECLARATION

I/We* hereby declare that I/We* have checked this thesis/project* and in my/our* opinion, this thesis/project* is adequate in terms of scope and quality for the award of the degree of Bachelor of Computer Science (Software Engineering)

(Supervisor's Signature)

Full Name : AHMAD FAKHRI BIN AB. NASIR

Position : SENIOR LECTURER

Date : 24 JUNE 2023

(Co-supervisor's Signature)

Full Name :

Position :

Date :

STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

____

(Student's Signature)

Full Name : NUR ATIQAH BINTI KAMAL

ID Number : CB20178

Date : 9 JUNE 2023

EARLY DETECTION OF ADHD AMONG CHILDREN USING MACHINE LEARNING

NUR ATIQAH BINTI KAMAL

Thesis submitted in fulfillment of the requirements
for the award of the degree of
Bachelor of Computer Science (Software Engineering) with Honours

Faculty of Computing
UNIVERSITI MALAYSIA PAHANG

JUNE 2023

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my supervisor, Dr. Ahmad Fakhri Bin Ab Nasir, for his direction, encouragement, and support throughout this project. Dr. Ahmad Fakhri Bin Ab Nasir guided me through the research process and successfully helped me finish this project with their knowledge and patience.

I owe a sincere debt of appreciation to my family for their constant support and inspiration along this trip. They have been my source of encouragement and inspiration; without them, I would not have been able to complete the task. Their compassion and kindness have served as a constant source of inspiration.

Additionally, I want to thank my friends and colleagues for their encouragement and support throughout this project. Their assistance has been vital in helping me accomplish my objectives. They have consistently been willing to listen, offer information, and make insightful suggestions. I appreciate all of your kindness and encouragement.

Last but not least, I want to express my sincere gratitude to everyone who helped me finish this project and supported me, mainly my thesis advisor, colleagues, family, friends and everyone involved. Your assistance and contributions have been crucial to the success of this project and have been of the utmost value.

ABSTRAK

Pengesanan awal gangguan kekurangan perhatian/hiperaktif (ADHD) pada kanak-kanak adalah penting untuk campur tangan tepat pada masanya dan hasil yang lebih baik. Pengimejan resonans magnetik berfungsi (fMRI) telah muncul sebagai alat yang berharga untuk memahami asas saraf ADHD. Abstrak ini meneroka kepentingan pengesanan awal ADHD, potensi fMRI untuk diagnosis ADHD, dan peranan pembelajaran mesin dalam memudahkan pengecaman awal. Dengan mengukur corak aktiviti otak, fMRI memberikan pandangan tentang keabnormalan fungsi yang dikaitkan dengan ADHD. Algoritma pembelajaran mesin boleh menganalisis data fMRI dan mengenal pasti biomarker yang menunjukkan ADHD, membolehkan klasifikasi yang tepat. Penyepaduan fMRI dan pembelajaran mesin menawarkan pendekatan yang menjanjikan untuk pengesanan awal ADHD, membolehkan intervensi yang diperibadikan dan strategi rawatan yang disesuaikan. Pengenalpastian awal menggunakan fMRI dan pembelajaran mesin mempunyai potensi besar untuk meningkatkan kehidupan kanak-kanak dengan ADHD melalui intervensi yang tepat pada masanya dan sokongan yang disasarkan.

ABSTRACT

Early detection of attention-deficit/hyperactivity disorder (ADHD) in children is vital for timely intervention and improved outcomes. Functional magnetic resonance imaging (fMRI) has emerged as a valuable tool for understanding the neural basis of ADHD. This abstract explores the significance of early ADHD detection, the potential of fMRI for ADHD diagnosis, and the role of machine learning in facilitating early identification. By measuring brain activity patterns, fMRI provides insights into the functional abnormalities associated with ADHD. Machine learning algorithms can analyze fMRI data and identify biomarkers indicative of ADHD, enabling accurate classification. The integration of fMRI and machine learning offers a promising approach to early ADHD detection, allowing for personalized interventions and tailored treatment strategies. Early identification using fMRI and machine learning holds great potential for improving the lives of children with ADHD through timely interventions and targeted support.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Attention Deficit Hyperactivity Disorder (ADHD) is a mental illness in which a person may struggle with impulsivity, hyperactivity, and attention difficulties. Hyper disorder is another name for ADHD. Less concentration and chaotic behaviour are signs of ADHD. ADHD symptoms can develop in childhood and last throughout adulthood. Each year, millions of people are impacted by ADHD. Due to overthinking and inactivity, a child with ADHD may struggle greatly in social situations, academic achievement, and peer relationships.

To prevent detrimental effects on children's social and academic lives, early diagnosis is crucial. One of the most common neuro-developmental and mental disorders, ADHD affects 5–10% of young children and is associated with lifetime disability, low quality of life, and a significant financial burden on the families of those who experience it. The basic mechanisms of ADHD are still unknown, just like those of many other neurological illnesses. The diagnosis of ADHD, which might take several months to complete and is based on observations made by parents or medical professionals, cannot be made using a single verified diagnostic procedure.

A common illness in children, attention deficit hyperactivity disorder (ADHD) manifests as lack of focus and hyperactivity. Children with ADHD frequently struggle to stay still, organise, and do their assignments. They may also occasionally be partially unaware of their surroundings. According to estimates, 3.9% of girls and 12.1% of males have ADHD.

There may be some signs of ADHD in youngsters that are more common than others. One of the most efficient ways to treat ADHD is to start the medical treatment process right away, which is made possible by an early diagnosis. ADHD symptoms typically start in the preschool years, but the issues they cause start in the school years. Self-expression by persons or those nearby is one of the most popular approaches to diagnose ADHD. Due to its ease of recording, non-invasiveness, and high temporal resolution, the use of fMRI is a very promising choice for early diagnosis in cases where clinical procedures for detecting ADHD in preschool-aged children are likely to be troublesome and slow the process. Clinical analysis takes longer since the length of time that the patient is being watched can change. It is made easier and more precise by machine learning. The use of machine learning ensures that programme error is kept to a minimal. Using a larger data collection will increase accuracy.

Machine learning is a branch of artificial intelligence that focuses on the development of algorithms and models capable of automatically learning and making predictions or decisions without explicit programming. It empowers computers to learn and improve from experience, enabling them to tackle complex tasks and extract meaningful insights from large datasets. Machine learning algorithms can analyze patterns, identify trends, and discover hidden relationships within data, allowing for the development of accurate predictive models. These models are trained on labelled datasets, where they learn from the provided examples to make predictions or classifications on new, unseen data. Machine learning has found applications in various domains, including image and speech recognition, natural language processing, recommendation systems, fraud detection, and medical diagnosis. With advancements in computing power and the availability of vast amounts of data, machine learning continues to evolve and revolutionize numerous industries, offering immense potential for innovation and automation.

1.2 Problem Statement

Today, ADHD is one of the most commonly known disorders in children of school age, but the lot of parents are unaware of its influence as one of the existing mental disorders that might influence or develop in their children at an early age. Children with ADHD are diagnosed when they start struggling in school. Teachers at the school are the one who observes the children in class. Children behaviour such as having problems in class, not listening when spoken to directly, daydreaming, fidgeting, failing to complete their homework and other assignments in class, etc., which makes the children stand out from the rest are one of the ADHD symptoms in children. However, because the teacher are unaware of the disorder, action was taken for treatment.

Children with ADHD experience challenges in all aspects of daily life, including social interactions, academic achievement, and low self-esteem. Additionally, if treatment is delayed, individuals may experience difficulties with language and speech, language and verbal memory, stimulation and activation, time processing of information, and timing.

Evidence indicates that early detection can influence the likelihood for a negative developmental progression, and frequently the signs are noticeable as early as kindergarten. In fact, it's thought that signs can be detected as early as age 4. It has been demonstrated that children with ADHD may be at low increased risk for a wide range of issues related to school, including functional disability during primary education and ongoing low academic performance. The examination of the mental characteristics connected to this disorder in children is seen as crucial due to the early development of ADHD symptoms. Abstract thinking, language development, critical thinking and many more are all impacted by the presence of ADHD symptoms.

Early detection of ADHD can greatly benefit parents, teachers, and doctors involved in the well-being and education of children. However, the current reliance on subjective assessments and delayed diagnosis poses significant challenges in identifying ADHD at an early stage. This leads to missed opportunities for timely intervention, appropriate support, and tailored educational strategies for affected children. Consequently, parents may struggle to understand and address their child's unique needs, teachers may face difficulties in managing classroom behavior, and overall, the child's academic and social development may be compromised. Therefore, there is a pressing need to explore and develop effective early detection methods, such as leveraging machine learning algorithms, to enable early identification of ADHD. By doing so, parents, teachers, and doctors can gain crucial insights into a child's ADHD risk at an early stage, allowing for proactive interventions, targeted support, and informed decision-making to optimize the child's academic, social, and emotional well-being.

The current implementation of how parents detect their children whether they have ADHD is by detecting the children behaviour by reading about the diagnosis or by seeing the doctor where the doctor examines their children backgrounds and behaviours. When seeing the doctor, they shall ask the parents some questions through questionnaire or survey to know more about the children and the children can go through physical test such as fMRI test, brain test, IQ test and many more.

Doctors uses the fMRI test on children and achieve the result based on the test. The fMRI approach tracks small variation in blood flow caused by brain activity. It can be used to look at the functional anatomy of the brain, pinpoint the parts of the brain that control key processes, evaluate other sickness, or choose the most effective course of treatment for the brain. Therefore, we can apply the use of fMRI and machine learning algorithm to detect the children fMRI images and obtained the results whether the children are categorised as Normal or ADHD.

1.3 Objective

The goal of this study is to develop an early detection of ADHD among children using Machine Learning algorithm. To achieve the goal, several objectives are set to achieve which are as follows:

i. To study ADHD among children and machine learning model.
ii. To develop machine learning model for detecting ADHD among children.
iii. To evaluate the performance of the developed model.

1.4 Scope

The scope that is focused in this study are outlined below.

i. ADHD 200 dataset.
ii. Data Collection and Acquisition.
iii. Data Pre-Processing.
iv. Feature extraction through Transfer Learning (TL) model used are Inception v3, VGG-16 and VGG-19.
v. Classification model used are k-NN, SVM, Random Forest, Naïve Bayes, and Logistic Regression.
vi. Performance Measure.

1.5 Thesis Organization

This thesis consists of five chapters. Chapter 1 shall discuss the introduction and the background of the study for this project. Chapter 2 will discuss the literature review where the proposed solution will be compared with other solutions that have been done. Chapter 3 will generally explain the methodology that will be used in this project to achieve the solution. Chapter 4 contains the implementation and results of testing from the solution that is produced. Chapter 5 will summarize the founding and conclude the results of this project.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter presents about the collection of existing work done and analysis of comparison on existing work by other researchers also relevance of comparison with the research title. The literature review shall explain on the related case studies with the research title, Early Detection of ADHD among Children. This chapter will also talk about the existing work that had been studied before by many articles and research paper available, the comparison of each existing work and the relevance of the existing work to the research title.

2.2 Related Case Studies

Table 2.1 shows the summarization of related case studies for analyzing and detecting ADHD among children.

Table 2.1 Literature Review Summary

Authors (Year)	Data Collection	Classifier	CA
Yanli Zhang-James, Ali Shervin Razavi, Martine Hoogman, Barbara Franke, and Stephen V Faraone (2022)	**Type of data used/collected:** fMRI image **Collected dataset:** ADHD-200	**Cross-validation:** K-Fold **Classifier:** CNN	The study received the accuracy of 72% applying training set class ratio between 0.4 ~0.6.
Zhenyu Mao, Yi Su, Guangquan Xu, Xueping Wang, Yu Huang, Weihua Yue,	**Type of data used/collected:** fMRI image	**Cross-validation:** NA	The study received the accuracy of 71.3%.

Authors (Year)	Data Collection	Classifier	CA
Li Sun, Naixue Xiong (2019)	**Collected dataset:** ADHD-200	**Classifier:** 4-D CNN	
Yuanze Qin, Yiwei Lou, Yu Huang, Rigao Chen, Weihua Yue (2022)	**Type of data used/collected:** fMRI image **Collected dataset:** ADHD-200	**Cross-validation:** NA **Classifier:** Trans3D-ensemble	They proposed method reaches excellent results compared to most methods based on single modality and gets the accuracy of 74.5%.
Atif, Muhammad Asad, E. Alonso, Greg Slabaugh (2020)	**Type of data used/collected:** fMRI image **Collected dataset:** ADHD-200	**Cross-validation:** NA **Classifier:** CNN	Their proposed method was able to achieve classification accuracy of 73.1%.
Md. Maniruzzaman, Jungpil Shin, and Md. Al Mehedi Hasan (2022)	**Type of data used/collected:** Survey **Collected dataset:** National Survey of Children's Health.	**Cross-validation:** NA **Classifier:** Random Forest (RF)	Their proposed method was able to achieve the highest classification accuracy of 85.5%.
Zhaobin Wang, Xiaocheng Zhou, Yuanyuan Gui, Manhua Liu & Hui Lu (2023)	**Type of data used/collected:** rsfMRI image **Collected dataset:** ABCD Research Consortium	**Cross-validation:** Nested **Classifier:** MKL	The study reached the highest accuracy of 74%.
Azadeh Mozhdehfarahbakhsh, Saman Chitsazian, Prasun Chakrabarti, KS Jagannatha Rao, Babak Kateb, Mohammad Nami (2016)	**Type of data used/collected:** rsfMRI image **Collected dataset:** ADHD-200 and ABIDE	**Cross-validation:** NA **Classifier:** CNN (12-layer)	They study received 92% accuracy applying the CNN with 12-layer architecture.
Senuri De Silva, Sanuwani Udara Dayarathna, Gangani Ariyarathne, Dulani	**Type of data used/collected:** fMRI image	**Cross-validation:** NA	The study shows that the SBC gained an accuracy between 84% and 86%.

Authors (Year)	Data Collection	Classifier	CA
Meedeniya, Sampath Jayarathna (2021)	**Collected dataset:** ADHD-200	**Classifier:** CNN (seed-based correlation (SBC))	
Christopher Sims (2022)	**Type of data used/collected:** fMRI image **Collected dataset:** ADHD-200	**Cross-validation:** NA **Classifier:** SM-3DLSTM SM-3DGRU MM-3DLSTM MM-3DGRU	The study received the average accuracy of 99.69% for the GRU model.
Donglin Wang, Don Hong, Qiang Wu (2023)	**Type of data used/collected:** fMRI image **Collected dataset:** ADHD-200	**Cross-validation:** NA **Classifier:** ICA-CNN Autoencoder	The study showed that ICA-CNN received an accuracy of 67% and while the correlation-autoencoder approach gives an accuracy rate of 69%.
Abhay M S Aradhya, Aditya Joglekar, Sundaram Suresh, M. Pratama (2019)	**Type of data used/collected:** fMRI image **Collected dataset:** ADHD-200	**Cross-validation:** NA **Classifier:** Deep Transformation Method	The study shows the results that the proposed Deep Transformation Method (DTM) achieves a mean classification accuracy of 70.36%.
Anshu Parashar, Nidhi Kalra, Jaskirat Singh and Raman Kumar Goyal (2021)	**Type of data used/collected:** EEG Signal **Collected dataset:** NA	**Cross-validation:** NA **Classifier:** AdaBoost	The study received the highest accuracy of 82%.
William Das & Shubh Khanna (2021)	**Type of data used/collected:** Pupillometry **Collected dataset:** NA	**Cross-validation:** 10-fold Nested **Classifier:** SVM	The SVM classifier achieved an average accuracy of 85.6%.
Ying Chen, Yibin Tang, Chun Wang Xiaofeng Liu, Li Zhao, Zhishun Wang (2019)	**Type of data used/collected:** fMRI	**Cross-validation:** Leave-one-out cross-validation	The study received an accuracy of about 88.1 %

Authors (Year)	Data Collection	Classifier	CA
	Collected dataset: ADHD-200	**Classifier:** CNN	
M Duda, N Haber, J Daniels and DP Wall (2017)	**Type of data used/collected:** Survey **Collected dataset:** NA	**Cross-validation:** Five-fold **Classifier:** Enet LDA	The study received an average classification accuracy of 82%.
Rohit Kale (2019)	**Type of data used/collected:** MRI and Survey **Collected dataset:** NA	**Cross-validation:** NA **Classifier:** Enet K-NN	NA
Md. Maniruzzaman, Jungpil Shin, Md. Al Mehedi Hasan and Akira Yasumura (2022)	**Type of data used/collected:** EEG Signal **Collected dataset:** NA	**Cross-validation:** NA **Classifier:** LASSO+SVM	The study received the highest accuracy of 94.2%.
Sahdev Saini, Rinkle rani, Nidhi Kalra (2022)	**Type of data used/collected:** EEG Signal **Collected dataset:** NA	**Cross-validation:** NA **Classifier:** K-NN	The study received the highest accuracy of 86%.
M Duda, R Ma, N Haber, DP Wall (2016)	**Type of data used/collected:** Survey **Collected dataset:** NA	**Cross-validation:** NA **Classifier:** Linear discriminant analysis	The study proposed a Linear discriminant analysis which received an accuracy of 96%.
S. De Silva, S. Dayarathna, G. Ariyarathne, D. Meedeniya (2019)	**Type of data used/collected:** EEG and fMRI **Collected dataset:** ADHD-200	**Cross-validation:** NA **Classifier:** Support Vector Machines (SVM) Extreme Learning Machine (ELM)	The study received an accuracy of 90% applying the ELM classifier on the EEG signal and 84.7% for fMRI dataset applying the SVM.

2.3 Existing Works

2.3.1 Questionnaire/Survey

One of the main methods of assessing ADHD symptoms is through surveys or questionnaires. A survey involves the process for gathering, organizing, and analyzing the answers for the survey. A questionnaire is any written sequence of questions. A questionnaire is a list of questions that is given to participants in specific research. It might be a part of a larger study.

The goal of a questionnaire is to gather data from a target audience. It may include a combination of closed-ended and open-ended questions. As they answer questions on a questionnaire, participants offer insightful information. It is possible to collect both quantitative and qualitative data. Quantitative information is measurable and numerical. Qualitative data are written, non-numerical data that should be further studied.

A survey is used to look into data about the respondent. Surveys frequently include questionnaires. A survey is often designed for a professional or academic purpose, and its compilation can be time-consuming and meticulous. They are strategic research methods that can give researchers important information. The accuracy of data collection or interpretation could reduce the validity of the results. This makes the entire survey pointless. Survey accuracy is essential because it costs money to conduct them. Learning more about a certain group of people is the major objective of a survey. There are several reasons to do this.

2.3.1.1 Dataset Collection

Parents of children aged 2 to 17 years participated in a survey for the study Crowdsourced validation of a machine-learning classification system for autism and ADHD, which gathered the data. Through social media networking sites like Facebook, Twitter, and Yahoo Groups, the research's objective is shared to the community across the United States (Duda et al., 2016). Parental consent was obtained for all responses, and general demographic data about the kid, including their gender, age, race, ethnicity, and annual household income, was gathered. The parents were required to respond to 15 questions regarding the child's routine behaviour.

Besides that, a research paper (Jafari et al., 2022) did a clinical sample to examine how differently the child and parent responses on the KINDL. Along with 127 and 1061 of their parents, the sample contained 1086 healthy children and 1086 children with ADHD. The KINDL was independently filled out by healthy school-age children and their parents, whereas the clinic version was done by ADHD-afflicted children and their parents. The KINDL has 24 questions in six domains for the child and parent reports: family, friends, school, self-esteem, physical well-being, emotional well-being, and self-worth.

In addition, a study focusing on child health and wellbeing that was taken from the 2018–2019 NSCH has been published. From the NSCH in 2018–2019, 59,963 young people between the ages of 0 and 17 participated. 56,006 subjects between the ages of 3 and 17 were enrolled in the study. For the questions they posed to the parents, they had classified the outcome variable as 1 for Yes and 2 for No. (Maniruzzaman, Shin, & Hasan, 2022)

2.3.1.2 Method Used

One of the techniques employed in all the papers analysed for the questionnaire/survey work is machine learning. Machine learning is a subfield of artificial intelligence, have the ability to act intelligently to replicate human behaviour. Artificial intelligence systems are utilised to complete challenging jobs in a manner similar to how humans solve problems. These systems use statistical methods to create intelligent computer systems that can learn from databases that are already available.

In one of the experiments, the best algorithms for differentiating ADHD were chosen. Elastic Net (ENet), Logistic Regression with l1 Regularization (Lasso), Linear Discriminant Analysis (LDA), Logistic Regression with l2 Regularization (Ridge), and Support Vector Classification (SVC) are the five techniques that were employed. They used the Grid Search tool to optimise the parameters for each model after training the five algorithms on the archive sample, testing them on the survey sample, and applying all five models to each subsampled training set. In order to find the optimal set of parameter values, the Grid Search function runs internal cross-validation (CV) on the training data set, accepting arrays of potential parameter values for each method as input. The ideal set of parameter values is usually found using a stratified three-fold CV. The model is fitted to the complete training set using the parameter values that produce the greatest accuracy on the held-out fold.

In another study, the uniform DIF across children with and without ADHD was detected using the generalised partial credit model with lasso penalty (GPCMlasso). When the difference in item response probabilities is constant across the entire construct scale, this is known as uniform DIF. The GPCMlasso model for evaluating uniform DIF across children with and without ADHD (Group variable), adjusted for child sex and age, can be expressed mathematically as follows:

$$\log\left(\frac{P(Y_{pi}=r)}{P(Y_{pi}=r-1)}\right)=\beta_i[\theta_p+x_p^T\alpha-\delta_{ir}-(\gamma_{i1}\times Group+\gamma_{i2}\times Age+\gamma_{i3}\times Sex)]$$

The GPCMlasso package used a variety of criteria, including the fivefold cross validation technique (CV) and Bayesian information criterion (BIC), to determine the ideal tuning parameter for variable selection and DIF assessment. BIC and CV differ technically and theoretically in DIF evaluation. BIC is consistent with regard to variable selection, but CV is intended to choose the optimal model in terms of prediction. Because the technique must be repeated for various training and testing data sets, CV also has the drawback of taking a lot of time.

For the article, Predicting Children with ADHD Using Behavioral Activity: A Machine Learning Analysis (Maniruzzaman, Shin, & Hasan, 2022), did statistical analysis to compare the differences in variables between ADHD and healthy children. When one class label is greater than the other class label, a dataset is said to be imbalanced, the imbalanced management method is utilised. An ML-based algorithm will be biased toward the majority class when classifying imbalanced data. We used two different data sampling techniques—undersampling and oversampling—to address this issue. As a result, ML-based classifier performance will be enhanced. Variable selection Feature selection (FS) is also known as variable selection in statistics and machine learning. FS is a method for determining which characteristics are the most helpful so that ML-based algorithms can perform better. To make models easier for readers to understand, to reduce overfitting and problem complexity, to shorten training times and costs, to avoid the dimensionality curve, and to increase the accuracy of ML-based models, FS is required. In order to identify the most important risk variables for the children with ADHD, the study employed LR as an FS approach. Since it can perform better if unnecessary features are eliminated from the model, linear regression (LR) is a useful model for testing feature selection techniques. By calculating the probability of the logit function, LR assesses the relationship between the output and one or more input variables.

The studies used machine learning methods to pick the classifier with the highest performance score out of eight ML-based classifiers that were used to predict the presence of ADHD in children. Random Forest (RF), Naive Bayes (NB), Decision Tree (DT), XGBoost, K-Nearest Neighbor (KNN), Multilayer Perceptron (MLP), Support Vector Machine (SVM), and 1-dimensional Convolution Network were the eight ML-based classifiers chosen for the study (1D CNN). On the basis of the grid search function, the hyperparameters were optimised. The grid search algorithm employs a cross-validation (CV) technique on the training set to extract the ideal values of the hyperparameters, taking as input arrays of all possible hyperparameter values for each classifier. The sets of hyperparameter values with the highest classification accuracy were chosen for this study's 10-fold CV.

2.3.1.3 Result

According to the article, Use of machine learning for behavioural distinction of autism and ADHD (Duda et al., 2016), Decision Tree and Random Forest were not suitable for the classification issue at hand. To create a classification model with "majority rules," the Random Forest algorithm uses multiple Decision Trees. Four of the six algorithms used in the study—SVC, LDA, Categorical Lasso, and Logistic Regression—performed well (AUC40.96), used just five behaviours, and reduced the number of behaviours recorded by more than 92%.

AUC ranged from 0.78 to 0.94 and accuracy ranged from 69.8% to 85.5% for eight ML-based classifiers employed in a different investigation that predicted the presence of ADHD in children. With an accuracy of 85.5% and AUC of 0.94, RF-based classifiers successfully predicted the ADHD-affected kids. This study shown that LR with RF-based classifiers can accurately identify and predict children with ADHD with good accuracy. This study will help medical professionals identify and treat youngsters with ADHD early on.

Another study showed (AUC = 0.89 0.01) that the ENet and LDA classifiers extended to survey data better. Given that these models are designed to handle intricate relationships between variables, they performed the best in machine learning studies for categorization. While ENet was designed to address correlation concerns with Lasso, LDA explicitly takes into account correlation between inputs in its generative criterion.

2.3.2 Functional Magnetic Resonance Imaging (fMRI)

The fMRI technique monitors small variations in blood flow that are brought on by brain activity. It can be used to examine the functional anatomy of the brain, identify the areas of the brain responsible for vital functions, assess the consequences of a stroke or other illness, or determine how best to treat the brain. Other imaging methods may be unable to find abnormalities in the brain that fMRI can. Medical professionals utilise magnetic resonance imaging (MRI) as a non-invasive technique to identify medical disorders. A strong magnetic field, radiofrequency pulses, and a computer are all used in MRI to provide precise images of inside body structures. Radiation is not used in MRI (x-rays). Medical professionals can study the body and find disease thanks to detailed MR pictures (Chen et al. 2020).

The preferred diagnostic technique for determining how a normal, ill, or injured brain functions as well as for weighing the risks of brain surgery or other invasive treatments is functional MRI (fMRI). FMRI is used by doctors to:

1. Look at the functional anatomy of the brain.
2. Perform a process known as brain mapping to identify the region of the brain responsible for important processes like thought, speech, movement, and sensation.

3. Assist in evaluating how brain function is affected by stroke, injury, or degenerative diseases like Alzheimer's.
4. Keep an eye on the growth and function of brain tumours.
5. Direct the preparation for brain surgery, radiation therapy, or other invasive procedures.

According to research, brain disorders including Alzheimer's, epilepsy, and ADHD can change the functional connectivity of the brain network. This connectivity alteration can be used to detect disorders like ADHD. There has been a lot of research done on ADHD, including studies using fMRI data and machine learning to look at changes in functional connectivity in ADHD.

2.3.2.1 Dataset Collection

The dataset from OpenNeuro.org is used in the paper where the purpose of this dataset is to investigate the working memory of the children and the feedback processing of the normal and ADHD children (N. Lytlea Marisa et al., n.d.). Participants complete 8 n-back tasks while undergoing functional magnetic resonance imaging (fMRI). 35 of the 79 children, who ranged in age from 8 to 12, and who had neuroimaging and underwent routine testing, had an official diagnosis of ADHD. The design of the multi-factor task is shown in the image below.

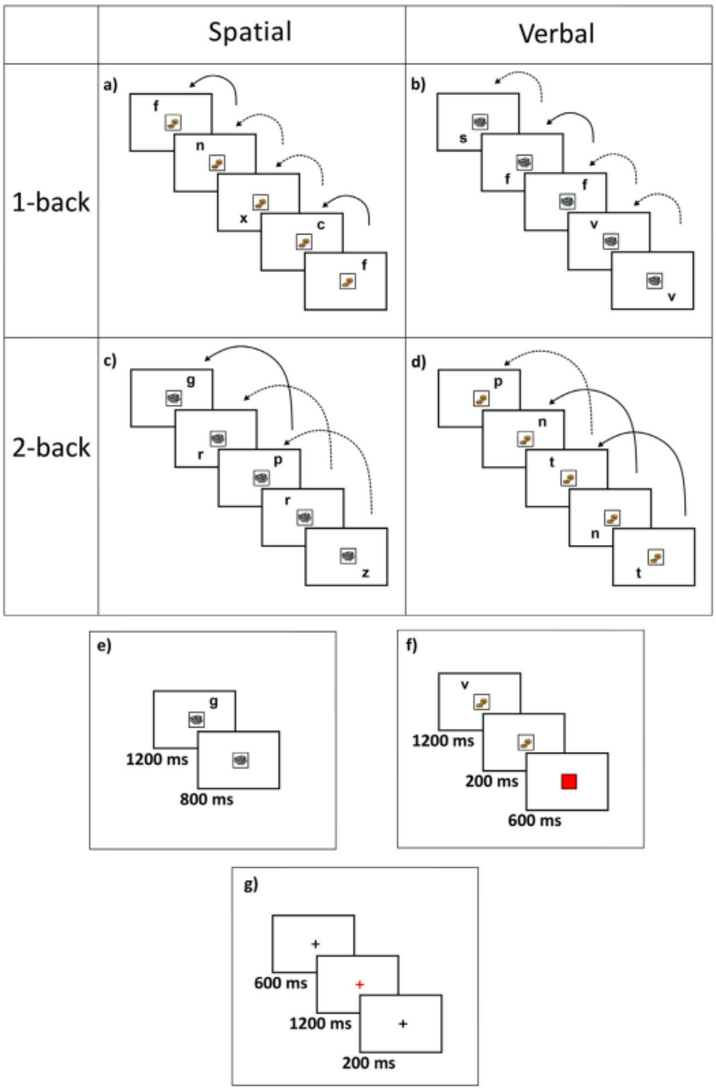

Figure 2.1 Task Design

In the scanner, the children will complete eight n-back working memory tasks. Participants were shown a set of letters one at a time for each task.

The children's resting state fMRI data from the ADHD-200 consortium were used in two more papers. The subjects can be divided into four categories: normal typically developing, ADHD, ADHD hyperactive impulsive, and ADHD inattentive. To examine the binary classification performance of those with ADHD and healthy control subjects, they have combined all kinds of ADHD into one group (Riaz et al. 2020).

2.3.2.2 Method Used

Dual subspace learning model was performed on the labelled ADHD and healthy control subject. The learned subspaces in this model should have three characteristics. In the right subspace, the projected component energy of the subject is maximised, whereas the energy is minimised in the rest subspace. Second, there is a significant difference between the ADHD and healthy control groups in each subspace. The subspace separations between the members of the two groups are increased. Finally, the learnt subspaces can represent the chosen FCs due to relationship consistency. It indicates that for patients in the same group, the projected components are quite near to one another (Chen et al., 2020b) .Therefore, model is presented as

$$(\tilde{\mathbf{Q}}_a,\ \tilde{\mathbf{Q}}_c) = \arg \min_{\mathbf{Q}_a,\, \mathbf{Q}_c} \frac{E_{inf}}{E_{major}} + \gamma \frac{1}{D} + \eta G + \lambda \|\mathbf{Q}_c^T \mathbf{Q}_a\|_F^2$$

$$s.\,t. \quad \mathbf{Q}_a^T \mathbf{Q}_a = \mathbf{I}, \quad \mathbf{Q}_c^T \mathbf{Q}_c = \mathbf{I}$$

And have summarized the dual subspace learning method in Algorithm 1, the figure below shows Algorithm 1.

Algorithm 1 Dual subspace learning

Input: Selected feature set $\mathbf{X} = [\mathbf{X}_a\ \mathbf{X}_c]$, weighted parameters $(\alpha, \gamma, \eta, \lambda)$, subspace basis number r, covariance matrices $\mathbf{R}_a$, $\mathbf{R}_c$ and $\mathbf{R}_{a,c}$, Laplacian matrices $\hat{\mathbf{L}}_a$ and $\hat{\mathbf{L}}_c$, convergence threshold T_{thr}.

1. Initialize subspaces $(\mathbf{Q}_a, \mathbf{Q}_c)$.

2. Calculate the cost function $f(\mathbf{Q}_a, \mathbf{Q}_c)$ in (14).

3. ***while*** $f(\mathbf{Q}_a, \mathbf{Q}_c) > T_{thr}$

4. Optimize subspace $\mathbf{Q}_a$ by (16).

5. Optimize subspace $\mathbf{Q}_c$ by (18).

6. Calculate the cost function $f(\mathbf{Q}_a, \mathbf{Q}_c)$ in (14).

7. ***end***

Output: Optimal dual subspaces $(\tilde{\mathbf{Q}}_a, \tilde{\mathbf{Q}}_c)$.

Figure 2.2 Algorithm 1

The databases' performance was also assessed in the article. Leave-one-out cross validation is used to achieve the classification accuracy. They have compared the classification model with many state-of-the-art (SOTA) techniques, including deep learning techniques like FCNet, 3D-CNN, and Deep fMRI, as well as machine learning techniques like graph fMRI, fusion fMRI, R-Relielf, and L1BioSVM.

2.3.2.3 Result

The methods make use of the binary hypothesis testing of test data and the sub-space-based categorization framework. The accuracy ranges for the most of the accuracy metrics produced by the existing machine learning and deep learning techniques were between 62% and 87%. The FCs of the test data are employed in the selection of the training data's individual resting-state FCs, which then have an impact on how well the learnt subspace's function when subjected to various hypotheses with changing projected energies. Consequently, the approach uses the suggested dual subspace classification algorithm to produce the best performance among these methods, with an average accuracy of 88.1%.

The results of the studies demonstrate that the method greatly surpasses state-of-the-art classification techniques, achieving an amazing accuracy of roughly 88.1% in the ADHD databases. The method's main flaw concerns the robustness of parameter setup because different databases, including NYU, KKI, NI, PU, and PU 1, require different parameter settings. As the stability of features heavily depends on the database size, there are several significant data imbalances and the children ADHD database is a small size.

2.3.3 EEG Signal

By inserting a scalp surface with attached electrodes, electroencephalography (EEG) is a popular physiological technique for recording electrical activities produced in the brain. EEG signals have a high sample rate, are nonstationary, nonlinear, and noisy, and may effectively detect brain wave patterns in cortical areas. The brain regions that are processing at a particular time can be determined via EEG. Each area has its own function, such as processing language, motor functions, and visual stimuli. These frequency patterns are also utilised to pinpoint the motor regions, sleep depth, relaxed states, and memory encoding. Therefore, alterations in the nervous system characteristics of neurological illnesses can be detected using EEG.

Many research has been conducted to identify ADHD using EEG data and deep learning and machine learning. Research shows that the brains of people with ADHD are different than the ones of those without. To check for changes in brain patterns, some clinicians do a physical examination. In 2013, the FDA authorised the use of electroencephalograms (EEG) to identify ADHD. Boris Kovatchev was the first to identify a physiological marker for ADHD in children. In his first investigation, he discovered that EEG signals contained several indicators, including indicators for ADHD, learning difficulties, and abnormalities. In his second study, he discovered that beta power had significantly dropped and theta activity had sharply increased.

2.3.3.1 Dataset Collection

The studies used an open-access resource for the EEG recordings dataset. The database included 121 participants (boys and girls, ages 7 to 12), including 61 ADHD-positive children and 60 typically developing children. A psychiatrist used DSM-5 criteria to make the diagnosis of ADHD in the youngsters. 19 channels (Fz, Cz, Pz, C3, T3, C4, T4, Fp1, Fp2, F3, F4, F7, F8, P3, P4, T5, T6, O1, and O2) of EEG data were recorded at a sample rate of 128 Hz (Maniruzzaman, Shin, Hasan, et al. 2022).

Since one of the main weaknesses in children with ADHD is visual attention, EEG recordings were made using tasks that tested visual attention. The kids were asked to count the cartoon characters after being given a collection of photos. The number of characters in each image varied from 5 to 16, and the image sizes were big enough for the kids to easily view and count. For continuous stimulation during the signal of the recording, each image was shown when the kids' answers were captured, instantly and without interruption. Therefore, based on the children's performance, the duration of the EEG recording during this cognitive visual task was determined.

2.3.3.2 Method Used

Figure 2.3 depicts the framework for the ADHD classification based on machine learning. Current datasets for ADHD have been subjected to a variety of machine learning methods in order to conduct a comparative analysis. machine learning models for the experiments using the Python language and Scikit-learn, which is included in Python were implemented. Here, the ADHD data set is applied to the AdaBoost, SVM, and Random Forest ML algorithms, and extensive experimentation has been carried out for

1. Choosing the right combination of features that would help the system achieve the target level of prediction accuracy.
2. Choosing the best machine learning algorithm to improve prediction accuracy.

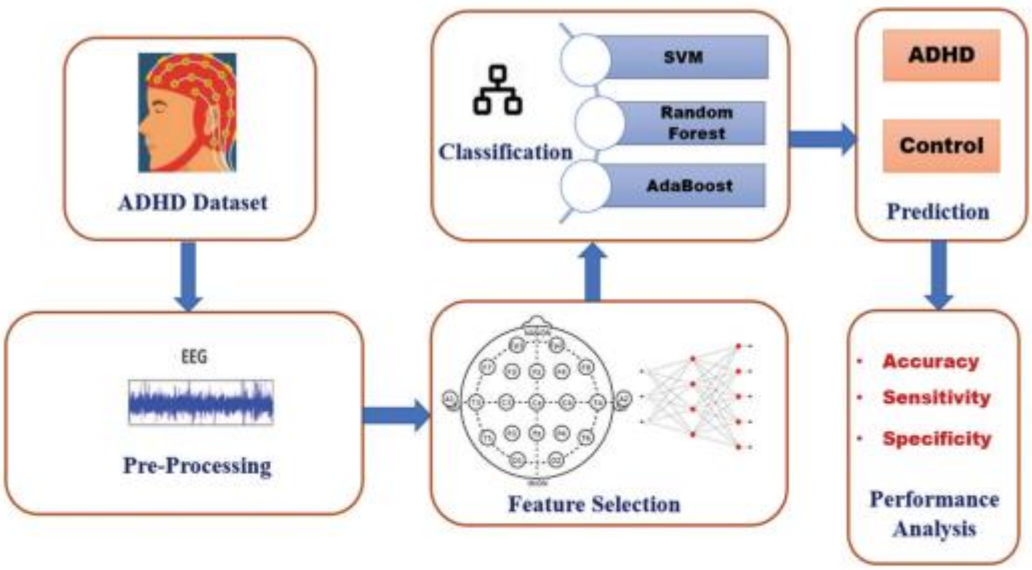

Figure 2.3 The framework for the Machine Learning-based Classification of ADHD

After extracting the various feature channel combinations, the feature set is then provided as input to the machine learning techniques Support Vector Machine, Random Forest, and AdaBoost. The most often used supervised classification method is Random

Forest. Random forests randomly choose a dataset from which to build a decision tree. The bagging principle underlies how Random Forest operates. For each iteration, it chooses a fresh dataset and builds a decision tree specifically for it (Parashar et al., 2021).

To put it another way, this dataset is divided into smaller datasets via Random Forest, and a decision tree is built for each smaller dataset. Every decision tree's forecast is used in the prediction stage. The predicted result is then chosen by the majority voting value. To overcome the problem of over-fitting, Random Forest is regarded as an ensemble classifier with a prediction rate that is higher than that of decision trees because of averaging the results. Figure 2.4 diagrammatically illustrates the working. The Random Forest classifier's algorithm is described as follows:

Algorithm-Random Forest (RF)Classifier

Input-Input Dataset of ADHD and Control Patients

Output- Classification of the patient (1-ADHD and 0- Control)

Step 1-Select Randomly, sub-datasets from the training input dataset.
Step 2-Create a decision tree for sub-datasets.
Step 3-Get the prediction result from every decision tree.
Step 4-Perform majority voting or compute average on outcome of Step 3.
Step 5-Select the most voted result as the final prediction.

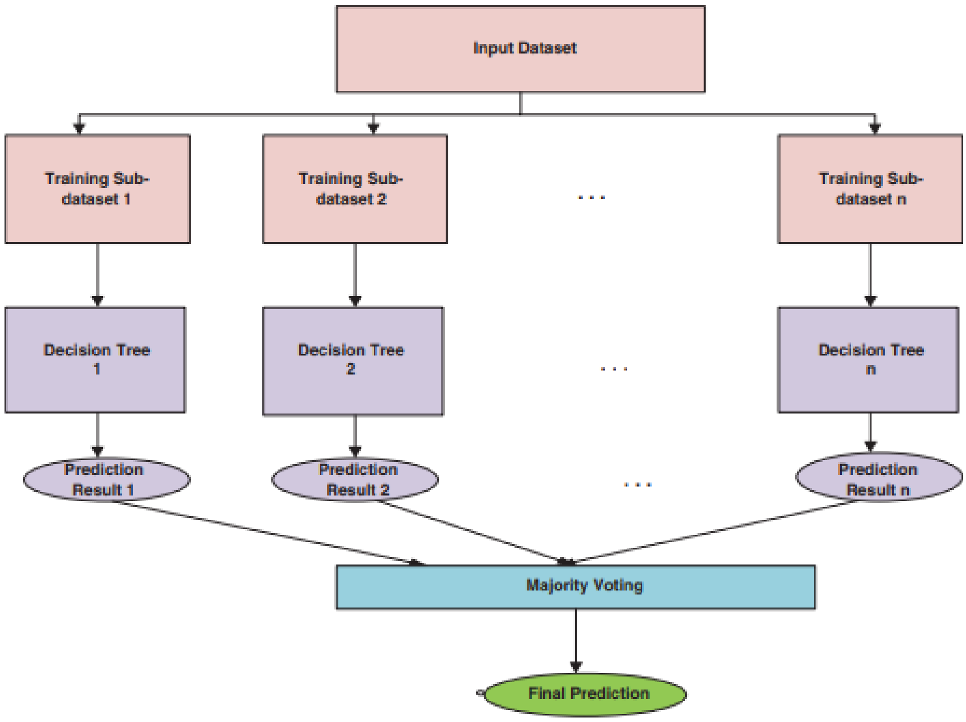

Figure 2.4 The algorithm for the Random Forest classifier

A supervised classification method is SVM. During the training phase, the Support vector machine encodes all data values as points in multidimensional space. A hyperplane linking entities of different classes is then constructed via SVM. SVM seeks to construct a hyperplane with a maximum margin level that properly classifies the dataset. During testing or prediction, new dataset values are additionally represented in the same multidimensional space, and they are categorised into one of several classes based on how close they are to the hyperplane. Figure 2.5 explains how the SVM classifier operates (Parashar et al., 2021). The SVM classifier's algorithm is described in depth as follows:

Algorithm-Support Vector Machine (SVM) Classifier

Input-Input Dataset of ADHD and Control Patients

Output- Classification of the patient (1-ADHD and 0- Control)

Step 1- Mark all the entries of datasets as data points in Multidimensional space.

Step 2-Create hyperplanes in an iterative manner that separates different classes.

Step 3-Choose the best hyperplane with maximum margin, that classifies the data points into different classes.

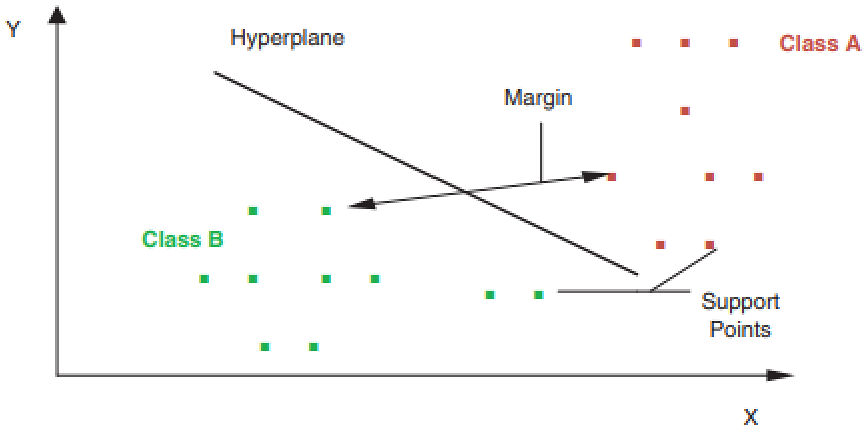

Figure 2.5 The algorithm for the SVM classifier

The AdaBoost ensemble-based classifier works on the principle that weak classifiers can be combined to produce stronger classifiers. AdaBoost improves classification algorithms that are currently weak. AdaBoost will sequentially generate several decision trees, and the following model will only accept training data from misclassified data points (Parashar et al., 2021).

Figure 2.6 shows a proposed ML-based framework for the prediction of children with ADHD and healthy children from a different research study that used the same dataset. The first phase is gathering data from 121 kids. After data normalisation, which removes bias, several morphological and time-domain features are extracted. The t-test and LASSO are two feature selection techniques that can be used to choose the most critical characteristics of ADHD in the fourth step. They utilized leave-one-out cross-validation (LOOCV) and adjusted the classifiers' various hyperparameter values. Four machine learning (ML)-based classifiers, including SVM, k-NN, multilayer perceptron

(MLP), and LR, were used to determine if a child had ADHD or not. (Maniruzzaman, Shin, Hasan, et al., 2022).

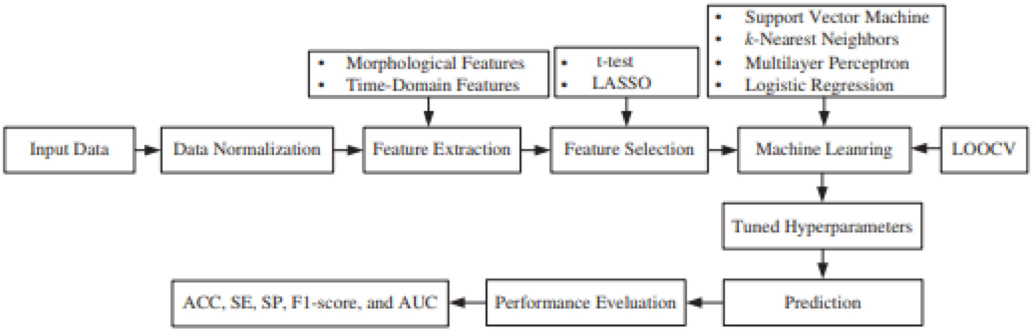

Figure 2.6 ML-based framework

Back propagation is a supervised learning method that is used in multilayer perceptron (MLP). It is also applied for classification and regression. It has input, concealed, and output node layers, which are three separate types. Except for the input node, each node uses a nonlinear activation function. Some of its hyperparameters must be estimated prior to training. The classification accuracy must be improved by adjusting these parameters (Maniruzzaman, Shin, Hasan, et al., 2022).

In the meantime, the Logistic Regression (LR) statistical model creates a link between a set of predictor variables and a dichotomous output variable. Based on the logistic function, it is used to calculate the likelihood of a particular class, such as: ADHD/healthy, diabetic/under control, alive/dead. LR can be used to predict a variety of illnesses, including diabetes, heart disease, and ADHD (Maniruzzaman, Shin, Hasan, et al., 2022).

2.3.3.3 Result

The research paper, Machine Learning Based Framework for Classification of Children with ADHD and Healthy Controls (Parashar et al., 2021), presents the classification results when individual regions are considered for distinguishing ADHD and Controls. It has been observed that the AdaBoost classifier performs best when employed with input channels from the parietal region. The linear classification model predicted by the SVM performed the worst out of the three classifiers. When individual channels are taken into account, the AdaBoost classifier performs better than the SVM's performance classifier with RBF Kernel. (Parashar et al., 2021).

	AdaBoost			Random Forest			SVM		
Brain Regions	**Accuracy**	**Specificity**	**Sensitivity**	**Accuracy**	**Specificity**	**Sensitivity**	**Accuracy**	**Specificity**	**Sensitivity**
Frontal (F)	0.71	0.53	0.85	0.78	0.75	0.79	0.55	0.43	0.74
Central (C)	0.8	0.72	0.85	0.67	0.36	0.96	0.52	0.43	0.56
Parietal (P)	**0.82**	**0.83**	**0.81**	0.71	0.53	0.85	0.5	0.57	0.46
Temporal (T)	0.69	0.40	0.92	0.69	0.52	0.83	0.57	0.61	0.51
Occipital (O)	0.8	0.59	0.93	0.71	0.43	1	0.57	0.76	0.46

Figure 2.7 Accuracy of AdaBoost, Random Forest and Support Vector Machine the (SVM)

SVM classifier, Random Forest, and AdaBoost are used for this purpose. When all the Right Hemisphere channels are taken into account, all classifiers' performances are evaluated, the AdaBoost classifier has the highest accuracy (84%).

The other study that used the accuracies of different classifiers across the t-test vs. LASSO are shown in Figure 2.8 (Maniruzzaman, Shin, Hasan, et al., 2022). When the t-test was used as FSM, the accuracy of the SVM, k-NN, MLP, and LR-based classifiers was 82.6, 79.4, 85.9, and 80.2. When applying the LASSO-based FSM, the classification accuracy of all classifiers (aside from k-NN) increased. The LASSO-based FSM feature sets produced the SVM feature sets with the highest accuracy (94.2%). Whereas, SVM with a t-test-based system achieved an accuracy of 82.6%. SVM with a t-test-based approach, on the other hand, attained an accuracy of 82.6%. Finally, it can be said that using SVM and a LASSO-based system, it is possible to distinguish between ADHD and healthy children. The research suggests that using the right features, FSM, and classifiers may help children with ADHD and healthy children better.

Accuracy (in %) of different classifiers across t-test and LASSO

Classifiers types	t-test	LASSO
SVM	82.6	94.2
k-NN	79.4	77.7
MLP	85.9	93.4
LR	80.2	81.8

Figure 2.8 Accuracies of different classifiers across the t-test vs. LASSO

The results demonstrated that performance scores from LASSO were better than t-test based FSM. The combination of LASSO-based FSM and SVM classifier, however, produced the best performance results (accuracy: 94.2%, sensitivity: 91.7%, and AUC: 0.964) (Maniruzzaman, Shin, Hasan, et al., 2022).

2.4 Analysis/Comparison of Existing Work

In this part, a comparison of methods and algorithms used and results for each exiting work, questionnaire/survey, fMRI and EEG signal will be shown in the Table 2.2 of comparison in Section 2.4.1.

2.4.1 Analysis of comparison of existing work

Table 2.2 Analysis comparison of existing work

Criteria	Questionnaire/Survey	fMRI	EGG Signal
Dataset	From the NSCH in 2018–2019, 59,963 young people between the ages of 0 and 17 participated	Dataset available on OpenNeuro.org, fMRI data in the experiment from the ADHD-200 consortium	Open-access database consisted of 121 participants, aged 7–12 years

Advantages	Easy to plan and execute Quantitative and Qualitative data	Find abnormalities in the brain that other imaging methods are unable to find	Identify alterations in brain activity that could be helpful in identifying mental problems
Disadvantages	Incomplete or dishonesty from the respondent	Expensive and researches doesn’t completely understand how fMRI works	Poor spatial resolution
Method Used	Random Forest (RF), Elastic Net (ENet), Linear Discriminant Analysis (LDA)	Support Vector Machine (SVM), K-Nearest Neighbors (k-NN)	AdaBoost, Support Vector Machine (SVM)
Accuracy	RF - 86% ENet and LDA - 86%	SVM – 87% k-NN - 85%	AdaBoost – 84% SVM – 90%
Advantages of Method	LR with RF-based classifier can accurately identify and predict children with ADHD with good accuracy	Can achieve above 80% with the fMRI image dataset	AdaBoost is less prone to overfitting as the input parameters are not jointly optimized.
Disadvantages of Method	If there are too many trees, the algorithm may be too slow and inefficient for making predictions in real time	Unbalanced data and the small size of the children ADHD database can affect the result of accuracy	Noisy data and outliers have to be avoided before adopting an AdaBoost algorithm

2.4.2 Relevance of comparison with project title

This comparison of the existing system will help to further understand how to choose the correct technique for identifying and predicting the ADHD in children. Questionnaire, fMRI, EEG Signal with machine learning methods/algorithms are good techniques for detecting the ADHD among children, but all of them have their advantages and disadvantages.

The chosen fMRI has been proven to be a good technique applying with many different classification models for the research such as Naïve Bayes, k-NN, SVM and Random Forest after doing a comparison with other techniques and algorithm available, this also means that the fMRI dataset with different model could be applied when doing a study on early detection of ADHD among children. Applying machine learning with fMRI data can help to detect the children brain area quickly with accuracy.

2.5 Summary

In summary, it is evident that the majority of the researches in this chapter will examine the ADHD dataset with different classification model to identify the ADHD among children and that fMRI can be one of a good technique to do an early detection of ADHD and applying different classification models for the research such as Naïve Bayes, k-NN, SVM and Random Forest with the chosen dataset. On the other hand, literature review was completed to read, understand, summarize, and analyse the contents of the research in order to comprehend and learn from the existing work that other researchers had completed. In this chapter, it has discussed the techniques and algorithm that can be used for detecting ADHD among children. And the relevance of comparison with the project title to be applied for the study on early detection of ADHD among children. In the next chapter, this report will cover the methodology that will be used for this project.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter discuss about the overall approach or framework of the project. It should cover method/technique or approach to be used and discuss the methodology in details to accomplish the research. It begins with Project Management Framework/Methodology which shows the methods, processes, tasks, resources and tools needed to take the project from beginning to end. Project Requirement discuss about the requirement needed to complete the research and continue with Propose Design which describe and come out with the proposed design that related to project requirement. Data design describe the data/assets involved that related to the research conducted and proof of Initial Concept will be provided in this chapter. Testing Plan constructed to test the functionality. Next, the potential use of the proposed solution will be discussed and finally, the Gantt chart planning of the whole project will be provided.

3.2 Project Management Framework

This research project is started with supervisor meeting to discuss research objective and scope definition of the selecting topic followed by literature review. After that, the motivation and problem statement have been discussed following the literature review that has been collected. From the literature review, the historical background and all related knowledge with the research have been summarized and some methods for the machine learning have also been discussed. Next, summarized and tabulated related research paper had been done is to gain different knowledge on other research paper such as topic, objectives, methods applied, datasets, models and the efficiency of the result of other researches. Other than that, it is followed by the discussion on the approaches and some explanation on the machine learning architecture. The methodology of this research is preparing with the process flow which are showing each of the steps to perform the research, research design planning and selecting suitable algorithms and approaches. Collecting necessary dataset for selected research topic was defined and dataset details was discussed. Moreover, developing classification algorithms and obtaining the ADHD prediction or detection result is children based on the used dataset and method. The classification algorithms are evaluated by calculating the performance and accuracy of the ADHD detection results. Lastly, thesis writing will be written up with the results that has been achieved.

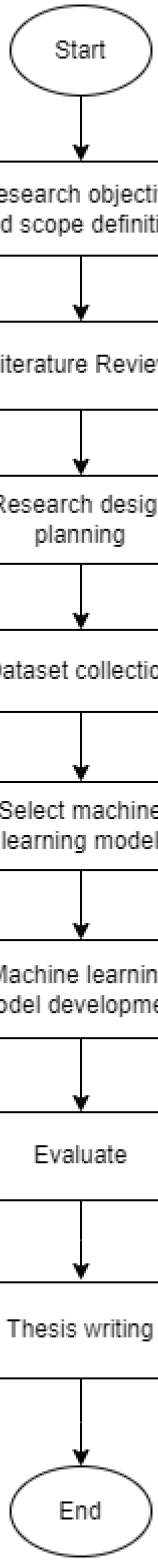

Figure 3.1 Research Process Flow Chart

The research methodology is then created which involves gathering the relevant dataset images, partitioning the dataset, choosing models for data extraction, using a classifier train and test the results using various pipelines. The research can be considered finished while the preliminary results were satisfactory. The model will be then be evaluated with performance. Figure 3.2 shows the methodology of modelling.

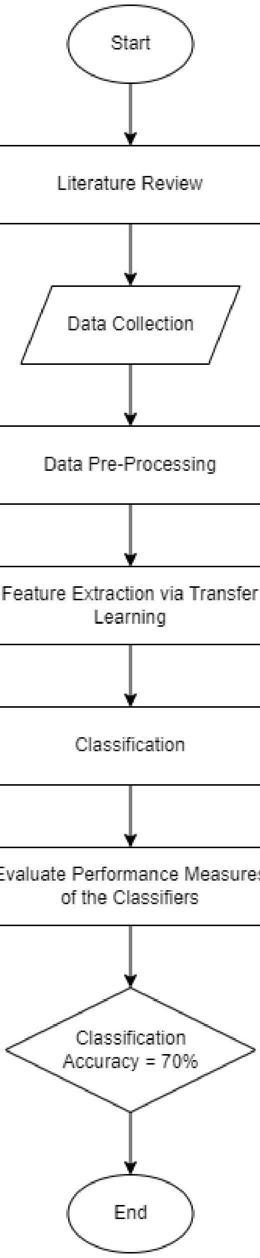

Figure 3.2 Research Methodology

In order to separate the population of ADHD and Normal fMRI images, the images dataset is obtained. Each data selected is then saved locally to be used later into the IDE using Python code. Before the data is being processed and separated into folders, the dataset shall go through image pre-processing. After that, data splitting process is being doing and the dataset will be prepared for the transfer learning techniques. Following the feature extraction, the classifier was used to classify the dataset images in order to produce the best result of accuracy in detecting ADHD among children.

Following the completion of all testing and training result, the performance metrics are observed to decide whether the results achieved with the transfer learning model and classifier model used is satisfied or not satisfied with the percentage of accuracy. If the outcome result is satisfied, the performance evaluation will be used to identify which model has the best results or pipelines. The classification accuracy must be above 70% in order to proof the selected model can obtain a satisfied outcome in detecting ADHD among children using the fMRI images dataset for the research.

3.3 Project Requirement

In this project, the model should be able to detect the ADHD symptoms in children based on the fMRI images that is provided from the dataset collection. Into developing the model, children fMRI scan that detects ADHD and Normal children is required. The model will be trained and produce the best model that is able to detect the ADHD among children.

This part will list all the software that will be used to develop the Early Detection of ADHD among children detection model:

Table 3.1 Software items

Software	**Purpose**
Microsoft Word 360	To create and compile all documents
Google Browser	To Google Colab notebooks for the development of the model
Python	To develop and deploy the model and testing
Jupyter	To develop and deploy the model

Table 3.2 Hardware items

Hardware	**Purpose**
Personal Laptop	To do all the processes related to the project

3.4 Propose Design

Figure 3.3 Research Methodology Process

1. Dataset Collection

Purpose

Preparing collection dataset in training phase and testing phase of model classifier. The process is searching related dataset from different websites and articles which uses the dataset of detecting ADHD.

2. Select Machine Learning Model

Purpose

Understanding the concept and process require in detection of ADHD in children. Evaluate all methods, processes and algorithms applied by previous related researchers and listing out the performance, effectiveness and accuracy of each algorithm which successfully in detecting and analyzing the result. Selecting the machine learning model to be used for the research. The process is to search for the relevant literature and study all journal related to the detection of ADHD. There is a total of 15 case studies has been studied and reviewed.

Output

Select the most suitable and accurate case study as a reference for this research topic and select machine learning model that it suitable for the research process.

3. Machine Learning Model Development

Purpose

To propose and develop a classification algorithm that is capable to successfully detect and predict the detection of ADHD, increase the performance and accuracy of prediction system. The input dataset is collected and the data is pre-processing for the algorithm to recognize.

Output

New classification algorithm that detects and analyse the dataset for the level of accuracy of the classification model.

4. Evaluate

Purpose

Evaluate whether the classification algorithm or model is capable of predicting the ADHD among children through the dataset collected.

3.5 Data Design

The Figure 3.4 shows the dataset that will be used for the detection of ADHD among children. The fMRI data used for this research is retrieved from ADHD-200 competition. The data provided by the competition consist of fMRI data and the subject information such as age, gender and IQ. The dataset was collected and contributed by eight different imaging sites. For the development and evaluation of the proposed project for this PSM, three imaging sites datasets were used: Brown, NeuroImage (NI), and Kennedy Krieger Institute (KKI). Figure 3.5, 3.6 and 3.7 shows the dataset that has been downloaded from the site from three imaging sites. Each sites have a folder with a number of subjects and each subjects have two fMRI images folder anat and rest which is as shown in figure 3.8. The example of fMRI images for anat and rest is shown in figure 3.9 and 3.10. These figures are the fMRI images that has been collected from different imaging sites under the ADHD-200 competition.

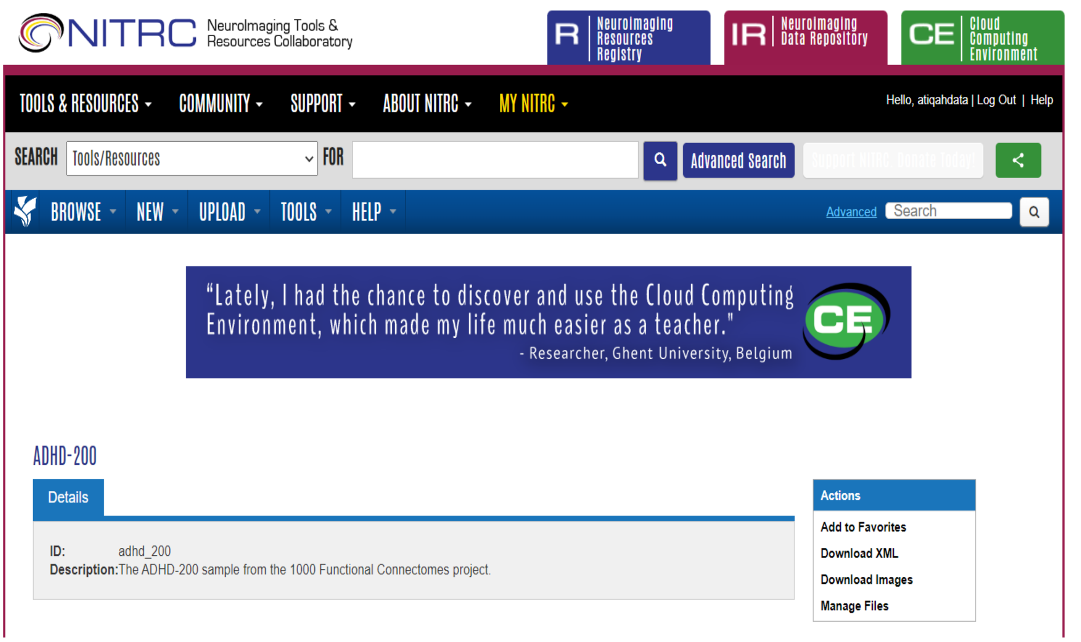

Figure 3.4 ADHD-200 dataset

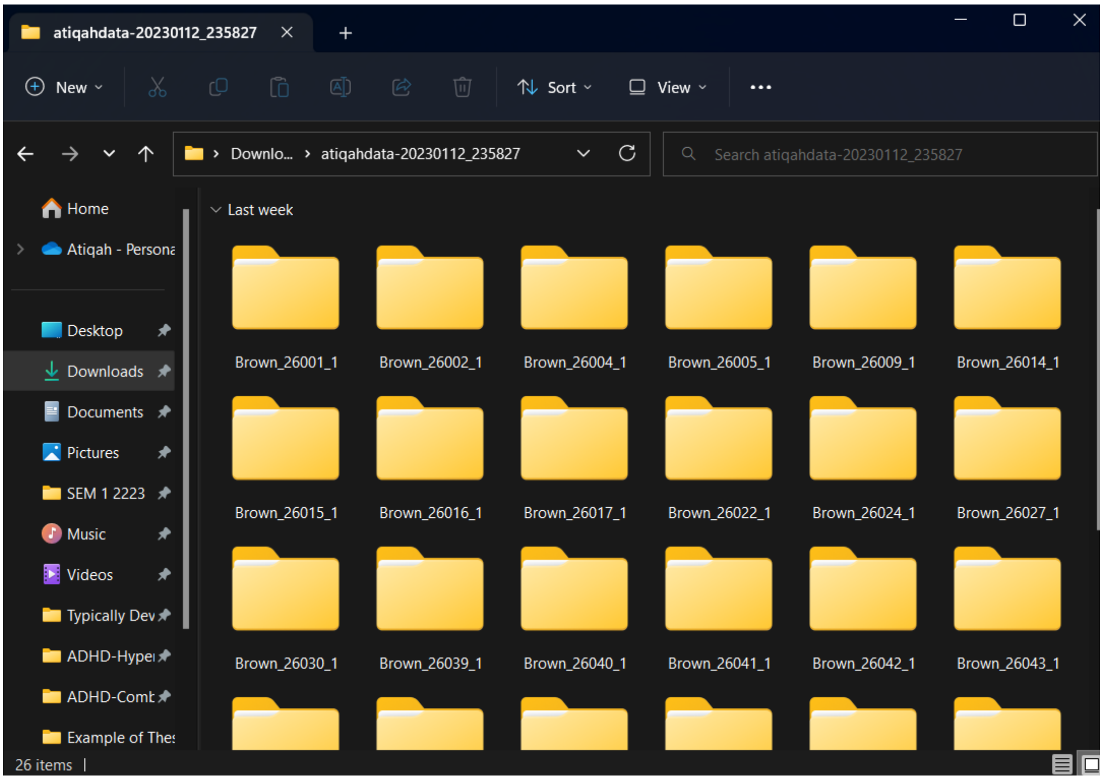

Figure 3.5 Brown image set folder

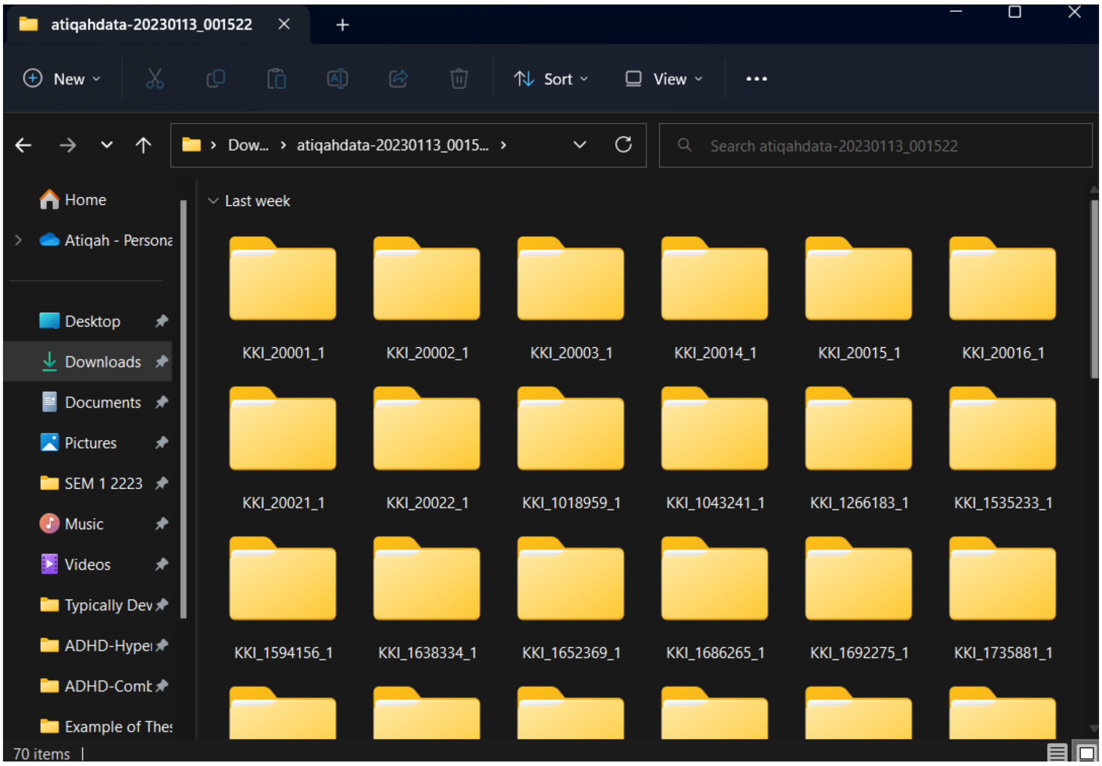

Figure 3.6 KKI image set folder

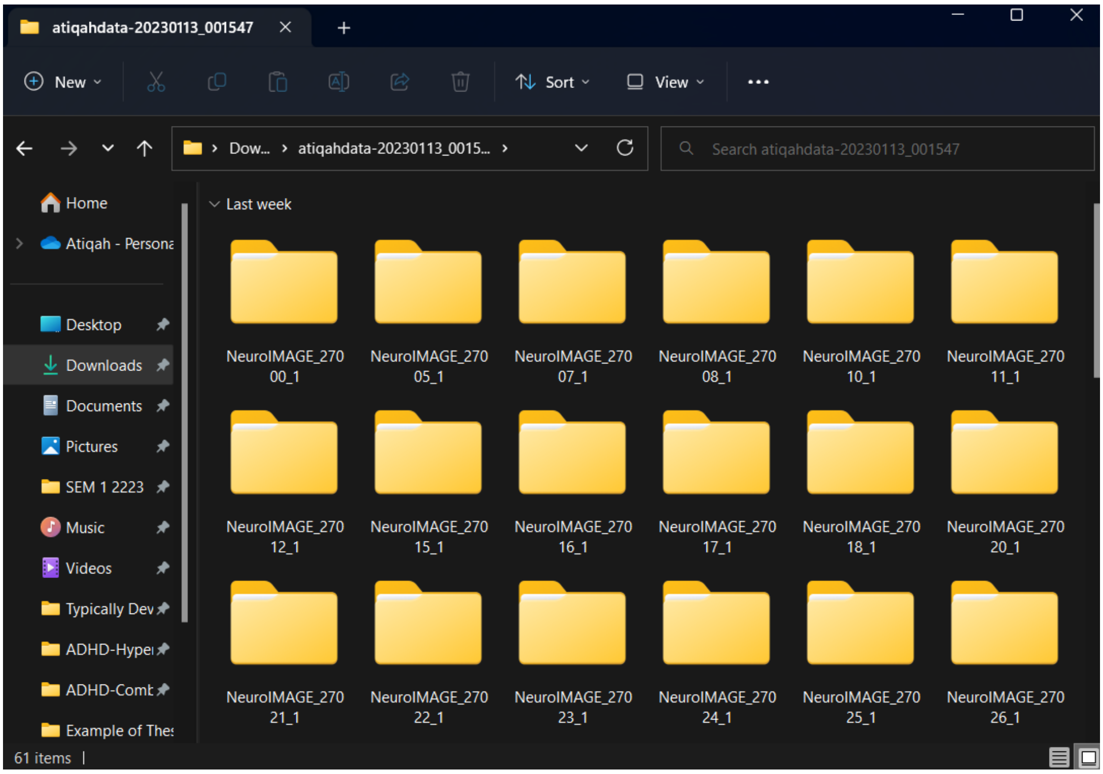

Figure 3.7 Neuroimage image set folder

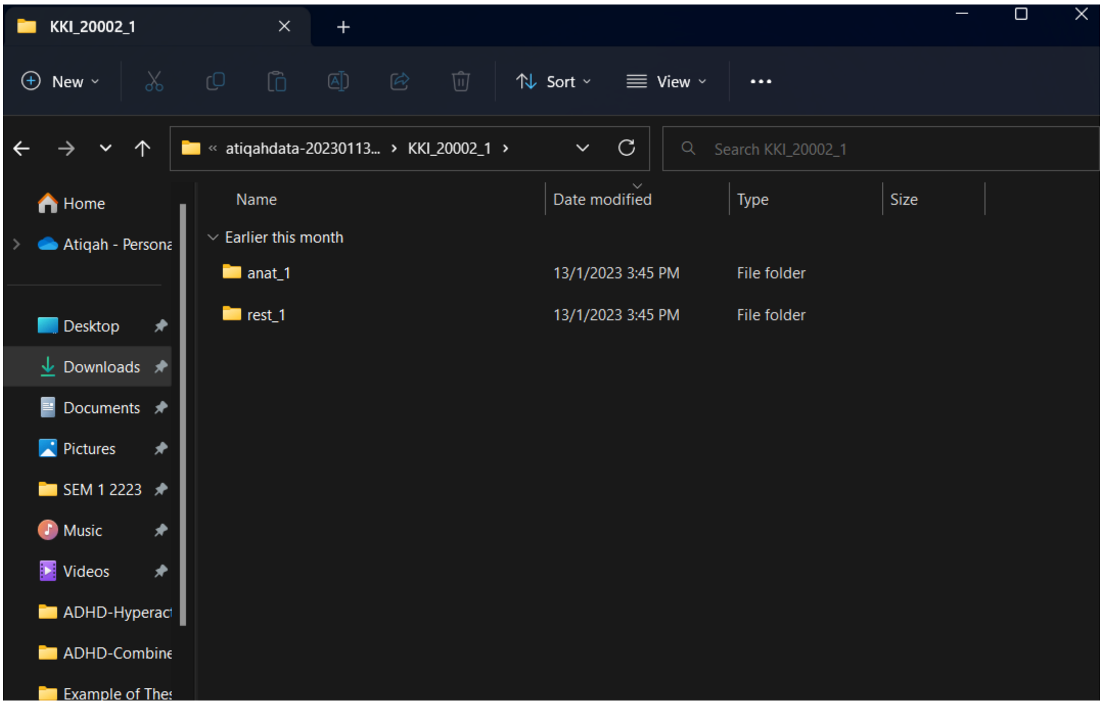

Figure 3.8 fMRI image folder

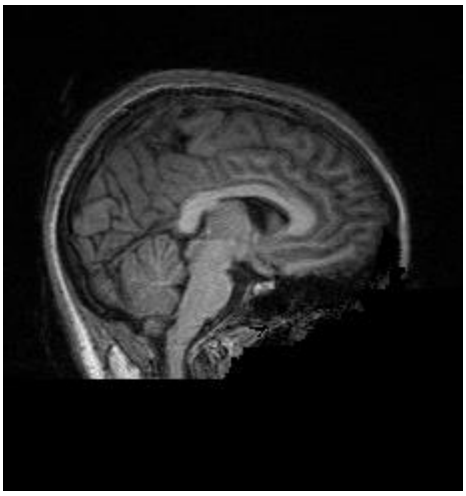

Figure 3.9 Anat fMRI image

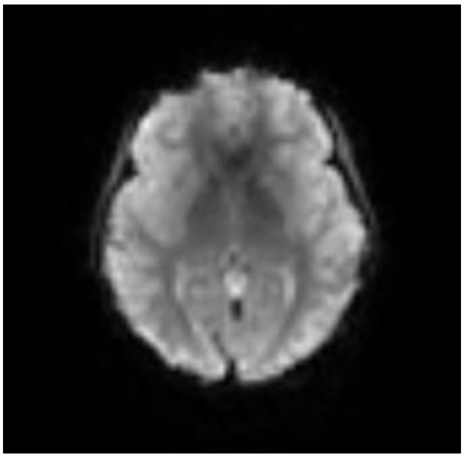

Figure 3.10 Rest fMRI image

3.6 Proof of Initial Concept

The proof of initial concept starts with data collection, selection on machine learning and evaluating the performance that is achieved from the fMRI dataset that had been used. By conducting the python with the Machine Learning (ML) techniques, the result for the Classification Accuracy (CA), F1, Precision and Recall score of the ML method were achieved.

Firstly, data organization on the dataset had been done. For the current dataset that had been collected, 32 subjects have been chosen as Normal/Typically Developing children fMRI images. Meanwhile, another categorization of ADHD such as ADHD-Hyperactive Impulsive, ADHD Combined, ADHD Inattentive has been categorized into one folder of ADHD.

In transfer learning, it contains several types of models and three of the models will be selected use for this research which is Inception v3, VGG-16 and VGG-19 applied. Inception v3 model is an image recognition model trained on ImageNet. The model is the culmination of many ideas developed by multiple researchers over the years. Next, VGG-16 is a convolutional neural network 16-layer image recognition model trained on ImageNet. Moreover, VGG-19 is a convolutional neural network 19-layers deep which can classify images into many object categories.

Different classifier was used such as Naïve Bayes classifier, K-Nearest Neighbor (K-NN) classifier, Support Vector Machine (SVM) classifier, and Random Forest (RF) classifier. The fMRI dataset was trained with different Transfer Learning Model and classifier to achieve the result of the models. The results applying inception v3 transfer learning model are shown below in Table 3.3.

Table 3.3 Result for Inception v3 TL-Model

Transfer Learning Model: Inception v3				
Model	**Accuracy**	**F1**	**Precision**	**Recall**
Naïve Bayes	0.81	0.81	0.82	0.81
k-NN	0.81	0.81	0.81	0.81
SVM	0.83	0.83	0.83	0.83
Random Forest	0.83	0.83	0.83	0.83

The SVM and Random Forest scores 0.83 which is the highest value among the Naïve Bayes that scores 0.81 and k-NN that scores 0.81 too. In the F1 score, SVM and Random Forest scores 0.83, while both Naïve Bayes and k-NN obtained 0.81 as the score. Besides that, the Random Forest and SVM perform the better performance in term of precision as it achieved 0.83 compared to Naïve Bayes score 0.82 and k-NN that score 0.81. Among the four ML methods, SVM and Random Forest achieved better result in recall score as it performs highest value, 0.83 compared to other methods.

The results applying VGG-16 Transfer Learning Model are shown below in Table 3.4. The tables shows that accuracy, F1, Precision and Recall that's achieved by each model.

Table 3.4 Result for VGG-16 TL-Model

Transfer Learning Model: VGG-16				
Model	**Accuracy**	**F1**	**Precision**	**Recall**
Naïve Bayes	0.77	0.76	0.77	0.77
k-NN	0.78	0.78	0.80	0.78
SVM	0.81	0.81	0.81	0.81
Random Forest	0.81	0.81	0.81	0.81

The SVM and Random Forest scores 0.81 which is the highest value among the Naïve Bayes that scores 0.77 and k-NN that scores 0.78 for the accuracy. In the F1 score, SVM and Random Forest scores 0.81, while both Naïve Bayes and k-NN obtained 0.76 and 0.78 as the score. Besides that, the Random Forest and SVM perform the better performance in term of precision as it achieved 0.81 compared to Naïve Bayes score 0.77 and k-NN that score 0.80. Among the four ML methods, SVM and Random Forest achieved better result in recall score as it performs highest value, 0.81 compared to other methods.

The results applying VGG-19 Transfer Learning Model are shown below in table 3.5. The tables shows that accuracy, F1, Precision and Recall that's achieved by each model.

Table 3.5 Result for VGG-19 TL-Model

Transfer Learning Model: VGG-19				
Model	**Accuracy**	**F1**	**Precision**	**Recall**
Naïve Bayes	0.81	0.81	0.81	0.81
k-NN	0.84	0.84	0.84	0.84
SVM	0.87	0.87	0.87	0.87
Random Forest	0.89	0.89	0.91	0.89

For the accuracy, Random Forest scores 0.89 which is the highest value among the Naïve Bayes, 0.81, k-NN, 0.844 and SVM, 0.87. In the F1 score, Random Forest score 0.89 which is highest than other models. Besides that, the Random Forest perform the better performance in term of precision as it achieved 0.91 compared to SVM that score 0.87, Naïve Bayes score 0.814 and k-NN that score 0.84. Among the four ML methods, Random Forest achieved better result in recall score as it performs highest value, 0.89 compared to other models.

It can be concluded that for both inception v3 and VGG-16 as the Transfer Learning Model for all four models, Support Vector Machine (SVM) classifier, and Random Forest (RF) classifier achieved the highest score in accuracy. Meanwhile, for the VGG-19 Transfer Learning Model, Random Forest achieved the highest score of 0.89 that other models. Since the accuracy of the fMRI dataset achieved above 50% even with the minimum amount of dataset, it can be stated that all four models can be used for the future use with bigger data collection of the children fMRI dataset. However, unbalanced data and the small size of the children ADHD database can affect the result of accuracy therefore for the future research, we will use more fMRI data from the ADHD-200 dataset.

3.7 Testing/Validation Plan

To test the functionality of the model, it must go through many processes such as Data Acquisition, Data Pre-Processing, Data Splitting, Performance Metrics and Classification Accuracy of the model to detect the children with ADHD or Normal.

3.7.1 Data Collection and Acquisition

Data collection and data acquisition play crucial roles in understanding and analyzing the ADHD 200 dataset, which consists of functional magnetic resonance imaging (fMRI) images. The ADHD 200 dataset is a publicly available neuroimaging dataset aimed at advancing research on Attention-Deficit/Hyperactivity Disorder (ADHD) and related conditions.

The data collection process is mentioned briefly in this section. Data Acquisition is a step-in machine learning algorithm. It is used to collect data on how the model performs on the selected dataset. Data acquisition is a process to display, store and analyse the existing dataset that has been collected and making sure the data that has been collected can be used for the model. Images of brain scan for fMRI method is a suitable representation to as it may convey size, shape, and some other characteristics of the human brain. The source of the dataset used in this study is from ADHD-200 Competition. The Anat fMRI images and right side of the brain scan is used for this research. The sample of Anat image data is shown in Figure 3.11

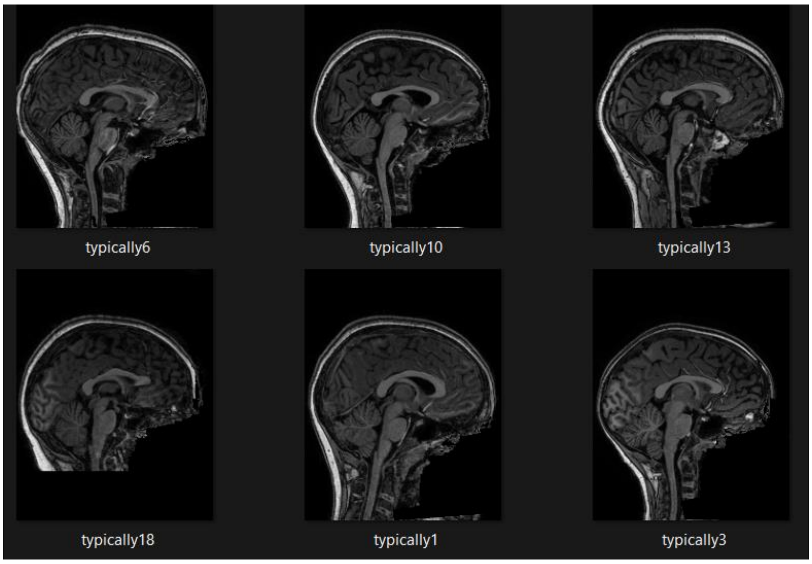

Figure 3.11 Anat fMRI image

For this research, a meticulous effort was undertaken to assemble a comprehensive dataset of fMRI images, with 566 images being diligently collected and curated. The primary focus of this data collection was centred around the task of image classification, particularly in the domain of ADHD diagnosis.

The dataset was thoughtfully composed by gathering 226 fMRI images specifically depicting individuals diagnosed with ADHD. These images were chosen carefully, considering the various manifestations and characteristics associated with the disorder. As many as 226 fMRI images were also included to represent individuals without ADHD, serving as the normal class within the dataset.

Moreover, to strengthen the robustness of the classification models and validate their performance, a separate subset of data was reserved for validation purposes. This validation set consisted of 57 fMRI images for individuals with ADHD, ensuring adequate representation of the disorder. Likewise, an equivalent number of 57 fMRI images were gathered to represent the neurotypical group in the validation set, enabling a fair and unbiased assessment of the classification models.

By carefully curating this comprehensive dataset comprising a total of 566 fMRI images, encompassing training and validation sets, this research explores the potential of image classification techniques in accurately discerning individuals with ADHD and those with Normal. Including a diverse range of images capturing the distinct characteristics associated with ADHD aims to enhance the efficacy and generalizability of the classification models developed within this study.

3.7.2 Data Pre-Processing

Data pre-processing is a crucial step in data mining that transforms raw data into a format that can be easily understood and analyzed. In practice, data often contains errors, inconsistencies, or discrepancies in codes or names. Analyzing the data before pre-processing can greatly enhance the accuracy of subsequent analyses. In the context of computer vision, where images or videos serve as input, digital image processing techniques are employed to improve image data quality by reducing unwanted distortions and enhancing important image characteristics.

During the data pre-processing phase, all images need to undergo resizing, feature augmentation, and channel adjustment. Resizing ensures that the image dimensions are fixed at 244x244 (height and width), aligning with the input dimension requirements of the VGG-16 and VGG-19 models. Additionally, features are appended to the image data and labeled to indicate the respective classes they belong to. This labeling helps the system correctly identify and categorize the loaded images under their respective classes, facilitating accurate classification.

3.7.3 Feature extraction through Transfer Learning (TL)

Transfer Learning (TL) is a highly prevalent approach in computer vision that facilitates the effective creation of precise models. It involves harnessing the knowledge acquired from training on one dataset and applying it to another dataset within the same domain. By utilizing pre-trained models on source data, the performance of the models can be enhanced through further training on target data.

During the transfer learning process, two renowned models, VGG-16 and VGG-19, developed by the Visual Geometric Group at Oxford University, are chosen from various alternatives. Furthermore, the Inception V3 model is employed for feature extraction in image analysis research.

3.7.3.1 VGG-16

Visual Geometric Group-16 (VGG-16) is a specific transfer learning model comprising 16 layers. It is one of the models introduced by the Visual Geometric Group at Oxford University. These 16 layers consist of 13 convolution layers and three fully connected layers following the last pooling layer. Among these layers, 13 are equipped with trainable parameters, while others, such as the maximum pooling layers, do not contain trainable parameters. (Khandelwal, 2020). The illustration is shown in Figure 3.12.

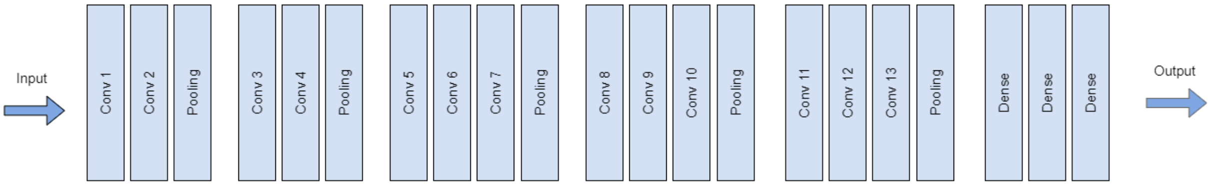

Figure 3.12 Structural of model VGG-16

3.7.3.2 VGG-19

The Visual Geometric Group-19 (VGG-19) is a transfer learning model that shares similarities with VGG-16 but comprises a network with 19 layers. In comparison to VGG-16, VGG-19 has three additional convolutional layers with trainable weights. However, the number of fully connected layers and maximum pooling layers remains the same as in the VGG-16 model (Yang, Zheng, & Merkulov, 2018). Figure 3.13 displays a structural representation of VGG19.

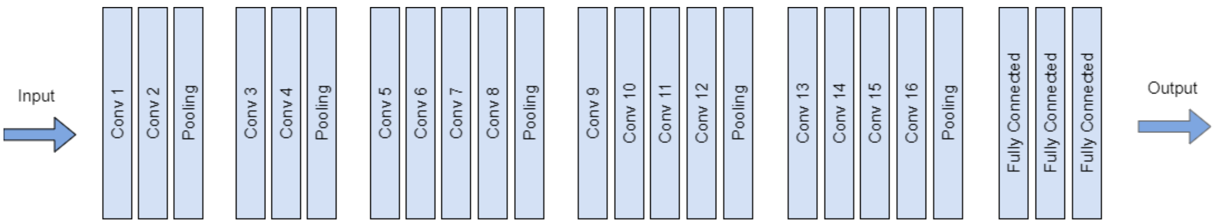

Figure 3.13 Structural of model VGG-19

3.7.3.3 Inception v3

Inception V3 is a deep convolutional neural network (CNN) architecture that was introduced by Google researchers as part of the Inception series of models. It was designed to excel at image classification and object recognition tasks. Inception V3 builds upon the success of its predecessors, Inception and Inception V2, by introducing several key improvements.

One of the notable features of Inception V3 is its use of inception modules. These modules consist of parallel convolutional layers with different filter sizes, allowing the model to capture features at multiple scales. By incorporating 1x1, 3x3, and 5x5 convolutions in parallel, the network can efficiently capture both local and global information within the image.

The architecture of Inception V3 includes multiple stacked inception modules, interleaved with max pooling and down-sampling layers. Towards the end of the network, global average pooling is applied, followed by fully connected layers and a softmax activation for classification (R Tamilarasi & S Gopinathan, 2021). Figure 3.14 display the illustration of Inception V3 (Anas Brital, 2021).

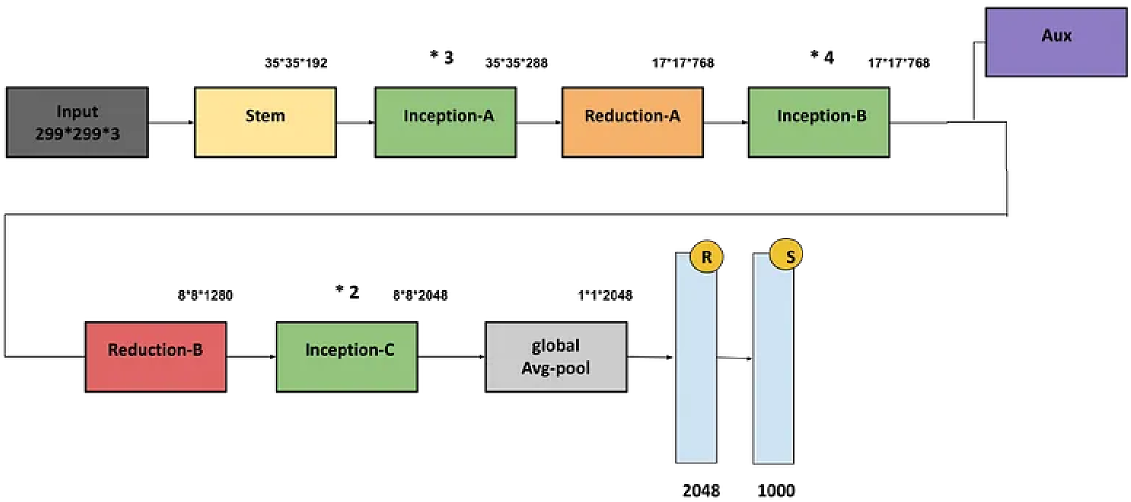

Figure 3.14 Illustration of model Inception V3

3.7.4 Classification

This research aims to classify the fMRI images into different classes, making supervised learning a suitable approach for implementing the classification system. Supervised learning involves using a labelled training dataset to identify patterns and make predictions for new, unlabelled data (Howard, 2019). In essence, supervised learning enables the classification process, allowing the system to use observed data for training and accurately categorizing test data into the appropriate class or group. As supervised learning offers classification capabilities, there are several classifiers that can aid in the classification system. The main five (5) classifiers that will is selected to conduct with this research are k – Nearest Neighbour (k-NN), Support Vector Machine (SVM), Random Forest (RF), Naïve Bayes and Logistic Regression.

3.7.4.1 k – Nearest Neighbour (k-NN)

K-nearest neighbour (k-NN) is a simple algorithm that makes it easy to implement supervised learning. The k-NN algorithm can solve both the classification and regression problems in supervised learning. This algorithm assumes similar things that exist nearby or similar things near each other. K-NN works can be done by finding the distance between a query and all the examples in the data by selecting the specified number of samples, the k, closest to the query (Harrison, 2018). k-NN classifier has two (2) types of parameters that can be tuned: the amount or number of k and the measurement method for the distance.

The k value has few thoughts while picking; the first thought is that there is no physical or biological way to determine the best value of k; it needs to try an error and check with the result. Secondly, a low value of k may be noisy and subject to the effect of outliers (Starmer, 2017). After that, the measurement method that is selected for use in this research are Euclidean, Manhattan, and Chebyshev measurements or distance. This measurement method will affect the final decision on the result of classification. This is because the distance may be slightly far from the actual group and be predicted to another group.

Euclidean measurement or distance is a distance between two points in the length of a line segment between the two points. This distance can be calculated from the Cartesian coordinate of the points using the Pythagorean Theorem (O'Neill, 2018). Equation 1, shown below, is the formula for the Euclidean distance.

$$d(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2}$$

Equation 1

Where, p and q is the coordinate of the point, and d(p,q) is the function of Euclidean distance.

Manhattan measurement or distance is a distance between two points measured along the axes at right angles. This distance measurement is calculated by the sum of the absolute difference of the Cartesian coordinate of the two points (Szabo, 2015). The formulation of the Manhattan distance is shown in Equation 2.

$$d_1(p,q) = \|p - q\|_1 = \sum_{i=1}^{n} |p_i - q_i|$$

Equation 2

Where, p and q is the coordinate of the point, and d1(p,q) is the function of Manhattan distance.

Chebyshev measurement or distance is a metric defined on the vector space. In other words, the distance between two vectors is the greatest of their differences along any coordinate dimension (Cantrell, 2000). The formula of the Chebyshev distance is shown in Equation 3.

$$D_{Chebyshev}(x,y) := \max_i(|x_i - y_i|)$$

Equation 3

Where, x and y is the coordinate of the point, and DChebyshev(x,y) is the function of Chebyshev distance.

3.7.4.2 Support Vector Machine (SVM)

Support Vector Machine (SVM) is a technique that represents an algorithm used for classification and regression analysis in supervised learning to analyze the data. The classification conducted under this classifier will be performed by finding the hyper-plane or line that can differentiate the two (2) classes or categories, as shown in Figure 3.10. As the hyper-plane or line is found, SVM will separate between the two classes by the maximized margin (Ray, 2017). In other words, SVM is a classifier used to find the hyper-plane to achieve the best separation of features into different classes (Yadav, 2018).

The SVM classifier is famous for its kernel trick. This kernel is a method for SVM to compute the dot product of two (2) vectors for the features. Generally, the kernel defines two (2) vectors, which are x and y (Yadav, 2018). The main kernels for the SVM are linear, sigmoid, and radian basic functions (RBF) (Editor of Cesar Souza, 2010).

Linear kernel is a useful kernel to deal with the large data vector and the equation of this kernel is shown in Equation 4.

$$k(x, y) = x \cdot y$$

Equation 4

Where, x and y are the vectors of the feature space.

Secondly, the Equation 5 outlined below was the formula to calculate the sigmoid kernel which is normally used for the neural networks.

$$k(x, y) = tanh(\alpha x \cdot y + c)$$

Equation 5

Where, k(x,y) is the function of the linear kernel , x and y are the vectors of the feature space, α is the slope value, and c is the inverse of the strength of regularisation.

Lastly, RBF or Gaussian kernel is the function that plays the trick on the distance from an origin or some point. This means it will use d the values that is depends on the distance outlined. The formula of the RBF kernel is shown in Equation 6.

$$k(x,x') = exp(-\frac{\|x-x\prime\|^2}{2\sigma^2})$$

Equation 6

Where, k(x,x') is the function of the linear kernel, x and x' are the vectors of the feature.

3.7.4.3 Random Forest (RF)

Random Forest is a scalable and straightforward supervised learning algorithm that generates the perfect result even though without changing any hyper-parameter, which is a practical and helpful algorithm that could be able to use in both classification and regression problems (Niklas Donges, 2019). Random forests are a combination of tree predictors. In this method, each tree depends on the random vector values independently sampled for all trees in the forest and with the same distribution. The generalization error of a tree classifier forest depends on the intensity and the connection between the individual trees in the forest (Breiman, 2001).

The random forest has similar hyperparameters as a decision tree or a classifier for bagging. In a random forest, combining a decision tree with a classifier for bagging is unnecessary. This is because the classifier class of random forest can be easily used. Other than that, while increasing the trees, Random Forest adds additional randomness to the model. In a random forest, only the random subset of the feature will be considered by the algorithm for splitting nodes, which can create more trees randomly by using random thresholds for each feature instead of looking for the best possible thresholds (Niklas Donges, 2019).

The random forest contains many individual decision trees that act like an ensemble. Each tree in the random forest will spit out a class prediction, and the class with the highest votes will become the model's prediction. Many relatively uncorrelated models working like committees will outperform any of the 75 individual constituent models. The requirements for executing well in a random forest must be some signal. Hence the model constructed using those features performs better than random guessing, and the prediction created by the peach trees must have a low correlation with each other (Yiu, 2019).

3.7.4.4 Naïve Bayes

The Naive Bayes Classifier is a probabilistic method of ML used for classification by the Bayes Theorem. Simply put, an NB classification assumes that there is no relation between a particular feature's existence in a class and the presence of any other feature. Highly sophisticated classification methods. In particular, the redundant data removal technique based on the NB algorithm is recognized even by highly sophisticated classification methods (Hema, Sankar, & Sandhya, 2018). The following example can rewrite the theorem of Bayes:

$$P(x \mid x') = \frac{P(x' \mid x)P(x)}{P(x')}$$

Equation 7

Where $P(x|x')$ represents the posterior likelihood of target features, x is given by predictor x′. In contrast, another $P(x'|x)$ is the likelihood of predictor given by target features. The $P(x)$ is the prior likelihood of features, whereas $P(x')$ is the prior likelihood of predictor. The NB algorithm is used to predict the probability that the pallet level will be specified, depending on the different features that the pallet will have, and that the pallet will belong to the group with the highest probability.

3.7.4.5 Logistic Regression

Logistic regression is a statistical learning algorithm used for binary classification tasks in machine learning. It models the relationship between input features and a binary target variable, estimating the probability of an instance belonging to a specific class.

The logistic regression model uses the logistic function (also known as the sigmoid function) to transform a linear combination of input features into a value between 0 and 1, representing the estimated probability. If the probability is above a threshold (commonly 0.5), the instance is classified as the positive class; otherwise, it is classified as the negative class.

The formula for logistic regression can be expressed as follows:

$$P(y=1|X) = 1 / (1 + e^{(-z)})$$

Equation 8

where:

$P(y=1|X)$ represents the probability of the instance belonging to the positive class given the input features X.

z is the linear combination of input features and their corresponding weights, along with an intercept term:

$$z = \beta 0 + \beta 1 * x1 + \beta 2 * x2 + ... + \beta n * xn$$

Equation 9

In the equation above:

β0 represents the intercept term or bias.

β1, β2, ..., βn represent the coefficients or weights associated with the input features x1, x2, ..., xn, respectively.

During the training phase, the logistic regression model estimates the optimal values for the coefficients (β0, β1, β2, ..., βn) through a process called parameter estimation. Common methods for estimating these parameters include maximum likelihood estimation or gradient descent, which minimize the difference between the predicted probabilities and the true class labels in the training data.

The logistic regression model can make predictions by applying the learned coefficients to the input features of new, unseen instances. The resulting probability can be used for classification by applying a threshold, as mentioned earlier.

Logistic regression is a widely used classification algorithm due to its simplicity, interpretability, and efficiency. However, it assumes a linear relationship between the input features and the log-odds of the target variable, which may limit its ability to capture complex nonlinear relationships. It is often used as a baseline model or in scenarios where interpretability is important.

3.7.5 Performance Metrics

Performance metrics is the standard metrics use to evaluate the efficiency of the model in the supervised learning especially in the classification system. The performance metrics is able to show the value, so the researchers can determine the suitable pipeline of the model in the research project. In this study, the confusion matrix or contingency table plays an important role as it helps visualize the data required for the performance metrics calculations. Table 3.6 it shows the confusion matrix.

Table 3.6 Confusion matrix

		Predict	
		Class 1	Class 2
Actual	Class 1	True Positive	True Negative
	Class 2	False Positive	False Negative

3.7.5.1 Classification Accuracy

When utilising a cross-validation method or an independent test set, classification accuracy is simply the percentage of correctly assigned classes. The classification accuracy formula displayed in Equation 10.

$$Classification\ Accuracy = \frac{Tp+TN}{Tp+Fp+Fn+Tn}$$

Equation 10

3.7.5.2 Precision

Precision is the ratio of correctly predicted +ve observations to the total predicted +ve observations. Precision formula was show in Equation 11.

$$Precision = \frac{Tp}{Tp+Fp}$$

Equation 11

3.7.5.3 Recall and Sensitivity

Recall is the ratio of correctly predicted +ve observations to all observations in actual +ve.

Recall formula was show in Equation 12.

$$Recall = \frac{Tp}{Tp+Fn}$$

Equation 12

3.7.5.4 F1 Score

The combination of precision and recall which are relative to a specific positive class. It is mainly used to compare two models with low precision and high recall or the other way around through harmonic mean. It aids to classify the model with a higher number of actual negative values, through this support it could obtain the best F1-score. The best FL score would be at 1, whereas the worst will be at 0. Equation 13 represents F1 score calculation.

$$F1\,Score = 2\ \times \frac{(Precision \times recall))}{(precision+recall)}$$

Equation 13

3.8 Potential Use of Proposed Solution

This proposed solution aim is to help the doctors to detect the children whether they have ADHD by using the system that is proposed to detect the fMRI images of the children brain. This solution can help the doctors to detect the children with ADHD using the fMRI images that had been used for the physical test and applying the machine learning to the system. This will help to raise the awareness of the people as ADHD is one of the most commonly known disorders in children of school age, but the lot of parents are unaware of its influence as one of the existing mental disorders that might influence or develop in their children at an early age.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

Chapter 4 presents the development, implementation, and results of an image classification system for functional Magnetic Resonance Imaging (fMRI) data to aid in the early detection of attention deficit hyperactivity disorder (ADHD) among children. Early detection of ADHD is critical for effective management and treatment of the disorder. In recent years, fMRI has emerged as a powerful tool for identifying biomarkers of ADHD. In this chapter, we describe our approach to developing an image classification system using machine learning techniques to analyze fMRI data and predict the likelihood of ADHD in children. We also present the results of our analysis, which demonstrate the potential of our approach for improving early detection and treatment of ADHD.

4.2 Manual Hyper-parameter Setting

Four manual hyper-parameter settings that were optimised are used in this study. The chosen parameter for the classifier, k-nearest neighbour (k-NN), is given in Table 4.1, the support vector machine (SVM), in Table 4.2, random forest shown in Table 4.3 and logistic regression shown in Table 4.4. Other than that, there will been having three (3) type of the model to be use in this research, which is Inception v3 model, VGG-16 model and VGG-19 model it will be show in Table 4.5.

Table 4.1 Setting for k-NN Classifier

Parameter	**Tune Condition/Setting**
No. of Neighbour, k	1-20
Metrics/Distance	Euclidean, Manhattan, Chebyshev

Table 4.2 Setting for SVM Classifier

Parameter	Tune Condition/Setting
Kernel	Linear, RBF, Sigmoid
Kernel Parameter	Degree, d=2,3; Gamma =0.1,1,10; Cost, c=0.1,1,10

Table 4.3 Setting for Random Forest Classifier

Parameter	Tune Condition/Setting
Number of trees	100 trees
Limit depth of individual trees	10
Do not split subset smaller than	5

Table 4.4 Setting for Logistic Regression Classifier

Parameter	Tune Condition/Setting
Regularization type	Ridge (L2)
Strength	C=1

Table 4.5 Feature extraction model

Model	Transfer Learning Model
Model 1	Inception v3
Model 2	VGG-16
Model 3	VGG-19

Three metrics and 20 different values of the neighbouring number, k, serve as the hyper-parameters for k-NN classifier. Each metric will be tested with 20 k-values, which translates to 20 sets of hyper-parameters tuning for each metric, since the model with this classifier will test output results with the hyper-parameter of combining one metric to one k-value one at a time. In order to choose the optimal hyper-parameter for the models, a total of 60 sets of hyper-parameters will be tweaked, tested, and optimised.

Three metrics have also been chosen for this experiment's utilisation with the SVM classifier. Moreover, multiple hyper-parameters are taken into account, including cost (c), gamma (gamma), and degree (d). Each of the three (3) metrics or kernels will be tested using a different combination of cost, gamma, and degree. In order to choose the optimal hyper parameter for the models with this classifier, a total of 54 sets of hyper parameters will be tuned, tested, and optimised. This means that there will be 18 sets of parameters to be tweaked for each kernel.

For the Random Forest classifier there is 3 parameters need to be set. The first parameter is the number of trees which mean that the number of trees that set inside the forest is 100 trees. The limit depth of individual trees in random forest which mean that the longest path between the root node and the leaf node have been set it equal to 10 maximums. Next, the minimum sample split was set as 5 which mean that the minimum number of samples required to split an internal node.

Two metrics need to be set for the Logistic Regression classifier. The first parameter is the regularization type. Regularization is used to reduce the complexity of the prediction function by imposing a penalty. For this research, Ridge (L2) was chosen as the regularization type. The parameter C is the inverse of regularization strength in Logistic Regression. For the strength of the regularization, C is set to 1.

4.3 Model Performance and Selection

4.3.1 k-NN Classifier

4.3.1.1 Classification Accuracy

In this k-NN classifier section, the best pipeline model will be selected and compare the classification accuracy. From the Table 4.6, it shows the classification accuracy of the different transfer learning model with the k-NN classifier. The average classification accuracy graph is shown in Figure 4.1.

Table 4.6 Classification accuracy of the Models

Model	**Classification Accuracy (%)**		
	Training	**Testing**	**Average**
Inception v3 + k-NN	88	80	84
VGG-16 + k-NN	87	79	83
VGG-19 + k-NN	88	82	85

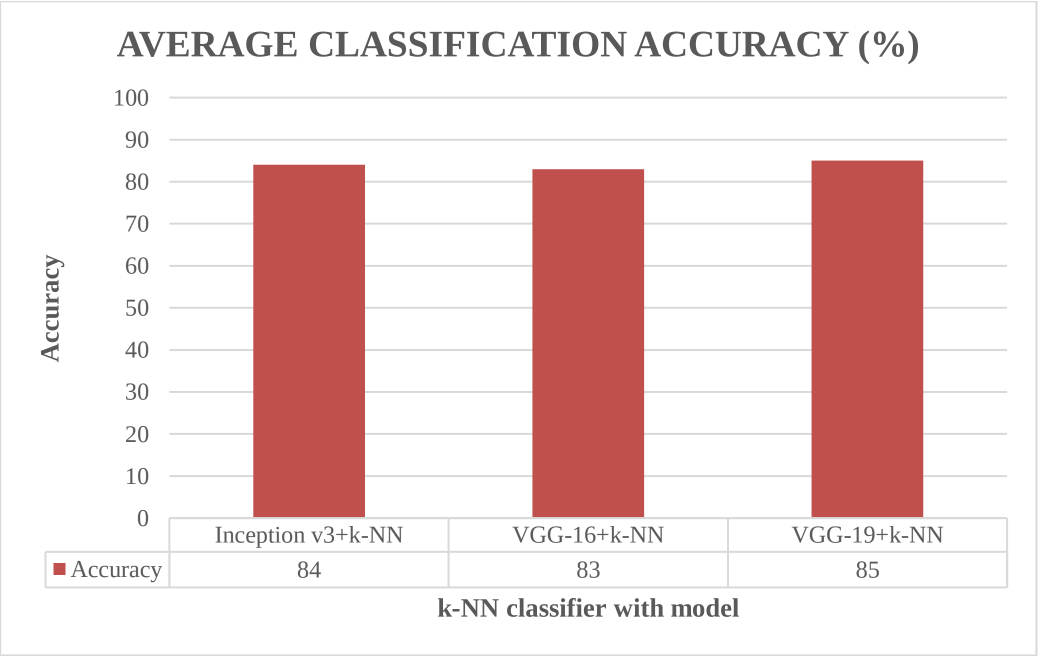

Figure 4.1 Average Classification Accuracy of k-NN classifier with model

The combination of the Inception V3 model with the k-Nearest Neighbors (k-NN) algorithm, the VGG-16 model with k-NN, and the VGG-19 model with k-NN showcased highly promising performance in the classification task of ADHD detection among children. These models leverage the power of deep learning architectures and the intuitive k-NN algorithm to achieve accurate predictions.

During the training phase, all three models demonstrated remarkable training accuracy. The VGG-16 + k-NN model achieved a training accuracy of 87%, indicating its ability to learn intricate patterns and extract relevant features from the training data. Similarly, the Inception V3 + k-NN model and the VGG-19 + k-NN model both achieved a training accuracy of 88%, signifying their effectiveness in capturing meaningful information from the dataset.

To evaluate the models' generalization capabilities, they were tested on unseen data. Even in this scenario, the models exhibited robustness and maintained commendable accuracy. The Inception V3 + k-NN model achieved a testing accuracy of 80%, implying its ability to accurately classify new instances. Likewise, the VGG-16 + k-NN model achieved a testing accuracy of 79% and the VGG-19 + k-NN model attained testing accuracies of 82%, demonstrating their reliability in predicting the ADHD status of children from unseen data.

When considering the overall performance of the models, taking into account both the training and testing accuracies, the average accuracy across the three models stood at a highly respectable 85%. This average accuracy showcases the consistent and reliable classification capabilities of the Inception V3 + k-NN, VGG-16 + k-NN, and VGG-19 + k-NN models in detecting ADHD among children.

These findings highlight the potential of machine learning techniques, particularly the combination of deep learning models like Inception V3 and VGG-16/VGG-19 with the intuitive k-NN algorithm, in aiding doctors and healthcare professionals in the early detection of ADHD. By leveraging the power of these models, medical practitioners can potentially improve the accuracy and efficiency of ADHD diagnosis, leading to timely interventions and improved outcomes for children.

4.3.1.2 Confusion Matrix

Training Dataset

In Figure 4.2, the confusion matrix for the training dataset, using the Inception V3 + k-NN model, revealed that out of the 226 instances, 217 were correctly classified as ADHD (true positive), while 13 instances that were predicted as normal were actually ADHD (false positive). Additionally, the model accurately identified 187 instances as normal (true negative) out of the 236, but misclassified 39 instances as ADHD when they were actually normal (false negative). These results highlight the model's ability to accurately detect ADHD cases, while also emphasizing the importance of minimizing false positive and false negative predictions to ensure reliable early detection and proper intervention for children.

		Predict	
		adhd	normal
Actual	adhd	217	13
	normal	39	187

Figure 4.2 Confusion matrix of k-NN classifier with Inception v3 training dataset

In Figure 4.3, the confusion matrix for the training dataset, using the VGG-16 + k-NN model, indicated that out of the 226 instances, 213 were correctly classified as ADHD (true positive), while 13 instances that were predicted as normal were actually ADHD (false positive). Furthermore, the model accurately identified 182 instances as normal (true negative) out of the 226, but misclassified 44 instances as ADHD when they were actually normal (false negative).

		Predict	
		adhd	normal
Actual	adhd	213	13
	normal	44	182

Figure 4.3 Confusion matrix of k-NN classifier with VGG-16 training dataset

In Figure 4.4, the confusion matrix obtained from training the VGG-19 + k-NN model on the dataset revealed that it correctly identified 201 out of 226 instances as ADHD (true positive), while 16 instances that were predicted as normal turned out to be ADHD (false positive). Additionally, the model accurately classified 186 instances as normal (true negative) out of the 226, but misclassified 40 instances as ADHD when they were actually normal (false negative).

		Predict	
		adhd	normal
Actual	adhd	201	16
	normal	40	186

Figure 4.4 Confusion matrix of k-NN classifier with VGG-19 training dataset

Testing Dataset

Figure 4.5 shows the confusion matrix for the testing dataset, utilizing the Inception V3 + k-Nearest Neighbors (k-NN) model, unveiled that out of the 57 instances, 45 were correctly classified as ADHD (true positive), while 12 instances that were predicted as normal were actually ADHD (false positive). Moreover, the model accurately identified 47 instances as normal (true negative) out of the 57, but misclassified 10 instances as ADHD when they were actually normal (false negative).

		Predict	
		adhd	**normal**
Actual	**adhd**	**45**	**12**
	normal	**10**	**47**

Figure 4.5 Confusion matrix of k-NN classifier with Inception v3 testing dataset

In Figure 4.6, the confusion matrix for the testing dataset, using the VGG-16 + k-NN model, revealed that out of the 57 instances, 48 were correctly classified as ADHD (true positive), while 9 instances that were predicted as normal were actually ADHD (false positive). Additionally, the model accurately identified 42 instances as normal (true negative) out of the 57, but incorrectly classified 15 instances as ADHD when they were actually normal (false negative).

		Predict	
		adhd	normal
Actual	adhd	48	9
	normal	15	42

Figure 4.6 Confusion matrix of k-NN classifier with VGG-16 testing dataset

Upon evaluating the testing dataset using the VGG-19 + k-NN model, the confusion matrix showcased notable findings. Figure 4.7 shows that among the 57 instances, 49 were accurately classified as ADHD, indicating a successful identification of true positives. However, the model misclassified 8 instances as normal when they were, in fact, ADHD, reflecting false positives. On the other hand, the model effectively identified 45 instances as normal, representing true negatives, out of the total 57. Nevertheless, there were 12 instances where the model wrongly categorized them as ADHD when they were actually normal, resulting in false negatives. These results underscore the significance of accurately identifying ADHD cases while emphasizing the need for minimizing false positive and false negative predictions to enhance the reliability of early detection in children.

		Predict	
		adhd	normal
Actual	adhd	49	8
	normal	12	45

Figure 4.7 Confusion matrix of k-NN classifier with VGG-19 testing dataset

4.3.2 SVM Classifier

4.3.2.1 Classification Accuracy

In this SVM classifier section, the best pipeline model will be selected and compare the classification accuracy. From the Table 4.7, it shows the classification accuracy of the different transfer learning model with the SVM classifier. The average classification accuracy graph is shown in Figure 4.8.

Table 4.7 Classification accuracy of the SVM Model

Model	Classification Accuracy (%)		
	Training	Testing	Average
Inception v3 + SVM	93	82	87
VGG-16 + SVM	93	84	88
VGG-19 + SVM	92	80	86

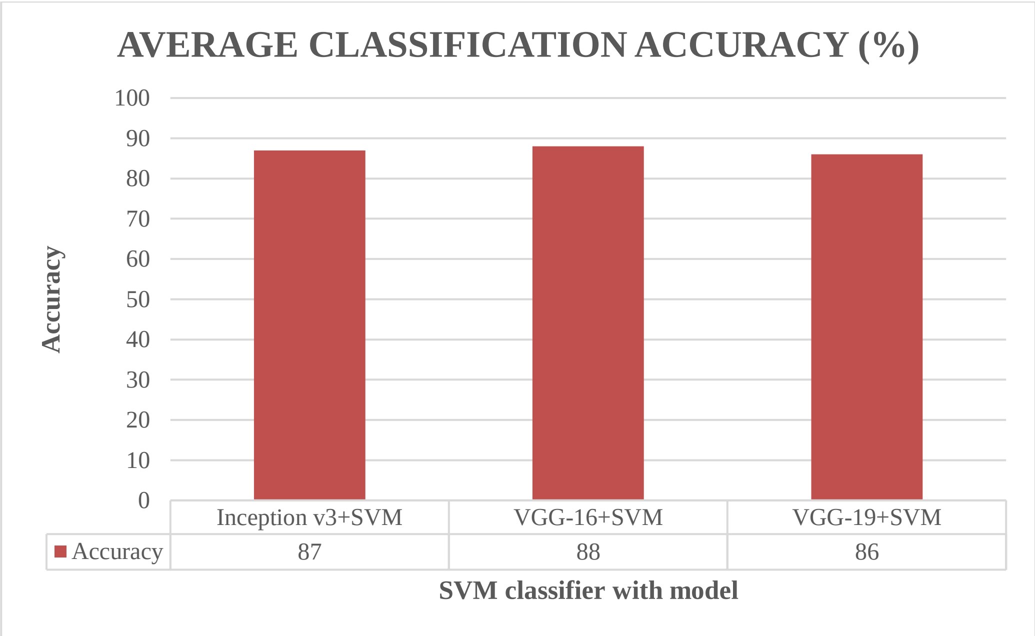

Figure 4.8 Average Classification Accuracy of SVM classifier with model

Among the various models that were evaluated, the combination of Inception V3 with Support Vector Machine (SVM) demonstrated highly impressive performance in the task of ADHD detection among children. This model exhibited exceptional training accuracy, achieving an impressive score of 93%, indicating its ability to capture the underlying patterns and characteristics of the training dataset effectively. During the testing phase, the Inception V3 + SVM model achieved a respectable accuracy of 82%. This suggests that the model was successful in generalizing its learnings to unseen data, thus demonstrating its robustness in real-world scenarios. The overall average accuracy of 87% further emphasizes the consistency and reliability of this model in accurately classifying ADHD cases.

Similarly, the VGG-16 + SVM model showcased strong performance throughout the evaluation process. With a training accuracy of 93%, it demonstrated the model's ability to capture the nuances and intricacies of the ADHD dataset. During the testing phase, the model achieved a high level of accuracy, achieving a commendable score of 84%, indicating its capacity to accurately classify new and unseen instances. The average accuracy of 88% further reinforces the reliability and effectiveness of the VGG-16 + SVM model in identifying ADHD cases early on.

Additionally, the VGG-19 + SVM model exhibited remarkable results, further highlighting the potential of this model in ADHD detection. With a training accuracy of 92%, it showcased its ability to learn and extract meaningful features from the training dataset. During the testing phase, the model achieved an accuracy of 80%, indicating its ability to generalize its learnings to new and unseen data. The average accuracy of 86% further solidifies the efficacy and consistency of the VGG-19 + SVM model in accurately identifying ADHD cases.

The exceptional performance of these models, including Inception V3 + SVM, VGG-16 + SVM, and VGG-19 + SVM, underscores their significant potential for aiding medical professionals in the early detection of ADHD among children. By leveraging the power of machine learning and combining it with advanced image analysis techniques, these models offer a promising approach to improving the accuracy and efficiency of ADHD diagnosis.

4.3.2.2 Confusion Matrix

Training Dataset

In Figure 4.9, the confusion matrix for the training dataset, using the Inception v3 + SVM model, revealed that out of the 226 instances, 198 were correctly classified as ADHD (true positive), while 28 instances that were ADHD were predicted as normal (false positive). Additionally, the model accurately identified 223 instances as normal (true negative) out of the 226, but misclassified 3 instances as ADHD when they were actually normal (false negative). These results highlight the model's ability to accurately detect ADHD cases, while also emphasizing the importance of minimizing false positive and false negative predictions to ensure reliable early detection and proper intervention for children.

		Predict	
		adhd	**normal**
Actual	**adhd**	198	28
	normal	3	223

Figure 4.9 Confusion matrix of SVM classifier with Inception v3 training dataset

In Figure 4.10, the confusion matrix for the training dataset, using the VGG-16 + SVM model, indicated that out of the 226 instances, 198 were correctly classified as ADHD (true positive), while 28 instances that were predicted as normal were actually ADHD (false positive). Furthermore, the model accurately identified 223 instances as normal (true negative) out of the 226, but misclassified 3 instances as ADHD when they were actually normal (false negative).

		Predict	
		adhd	**normal**
Actual	**adhd**	**198**	**28**
	normal	**3**	**223**

Figure 4.10 Confusion matrix of SVM classifier with VGG-16 training dataset

In Figure 4.11, the confusion matrix for the training dataset, using the VGG-19 + SVM model, revealed that out of the 226 instances, 198 were correctly classified as ADHD (true positive), while 28 instances that were ADHD were predicted as normal (false positive). Additionally, the model accurately identified 220 instances as normal (true negative) out of the 226, but misclassified 6 instances as ADHD when they were actually normal (false negative).

		Predict	
		adhd	**normal**
Actual	**adhd**	**198**	**28**
	normal	**6**	**220**

Figure 4.11 Confusion matrix of SVM classifier with VGG-19 training dataset

Testing Dataset

The confusion matrix for the testing dataset, utilizing the Inception V3 + SVM model, Figure 4.12 unveiled that out of the 57 instances, 46 were correctly classified as ADHD (true positive), while 11 instances that were predicted as normal were actually ADHD (false positive). Moreover, the model accurately identified 48 instances as normal (true negative) out of the 57, but misclassified 9 instances as ADHD when they were actually normal (false negative).

		Predict	
		adhd	**normal**
Actual	**adhd**	**46**	**11**
	normal	**9**	**48**

Figure 4.12 Confusion matrix of SVM classifier with Inception v3 testing dataset

When examining the testing dataset using the VGG-16 + SVM, the confusion matrix provided insightful results. Figure 4.13 shows that out of the 57 instances, 46 were accurately labelled as ADHD, representing true positives. However, the model erroneously classified 11 instances as normal when they were actually ADHD, indicating false positives. Conversely, the model correctly identified 50 instances as normal, reflecting true negatives, out of the total 57. Nonetheless, there were 7 instances where the model misclassified them as ADHD when they were, in fact, normal, resulting in false negatives.

		Predict	
		adhd	**normal**
Actual	**adhd**	**46**	**11**
	normal	**7**	**50**

Figure 4.13 Confusion matrix of SVM classifier with VGG-16 testing dataset

Upon evaluating the testing dataset using the VGG-19 + SVM model, the confusion matrix showcased notable findings. Figure 4.14 shows that among the 57 instances, 43 were accurately classified as ADHD, indicating a successful identification of true positives. However, the model misclassified 14 instances as normal when they were, in fact, ADHD, reflecting false positives. On the other hand, the model effectively identified 49 instances as normal, representing true negatives, out of the total 57. Nevertheless, there were 8 instances where the model wrongly categorized them as ADHD when they were actually normal, resulting in false negatives.

		Predict	
		adhd	normal
Actual	adhd	43	14
	normal	8	49

Figure 4.14 Confusion matrix of SVM classifier with VGG-19 testing dataset

4.3.3 Random Forest Classifier

4.3.3.1 Classification Accuracy

In this Random Forest classifier section, the best pipeline model will be selected and compare the classification accuracy. From the Table 4.8, it shows the classification accuracy of the different transfer learning model with the Random Forest classifier. The average classification accuracy graph is shown in Figure 4.15.

Table 4.8 Classification accuracy of the Random Forest Model

Model	Classification Accuracy (%)		
	Training	Testing	Average
Inception v3 + Random Forest	91	80	85
VGG-16 + Random Forest	93	82	87
VGG-19 + Random Forest	91	80	85

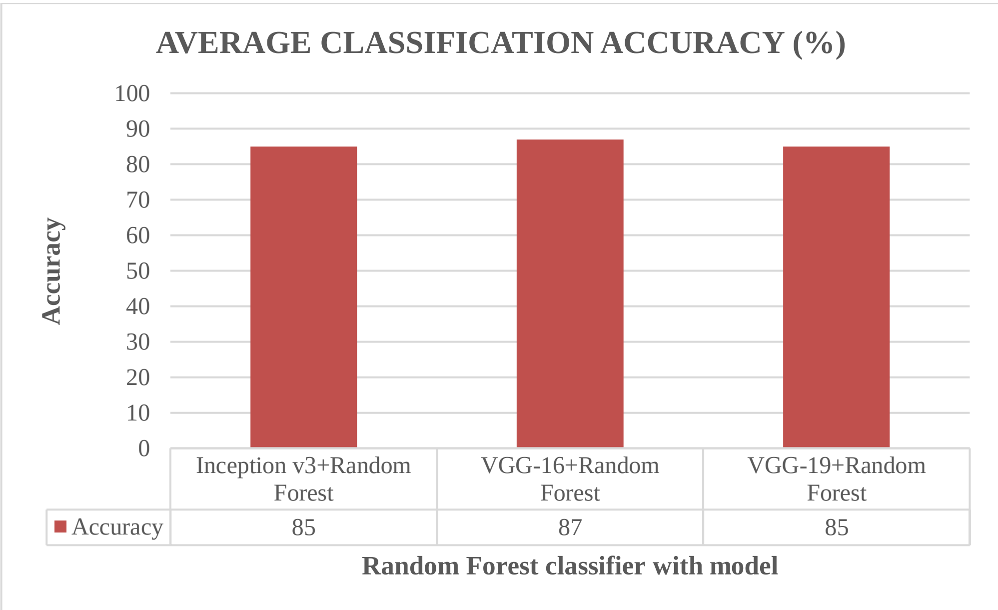

Figure 4.15 Average Classification Accuracy of Random Forest classifier with model

Upon evaluating the performance of the models, it was observed that the combination of Inception V3 with the Random Forest algorithm yielded impressive results. The Inception V3 + Random Forest model demonstrated exceptional training accuracy, reaching an impressive 91%. During the testing phase, it exhibited a satisfactory accuracy of 80%, contributing to an overall average accuracy of 85%. These findings suggest that the Inception V3 + Random Forest model shows promise in accurately classifying ADHD-related data.

Similarly, the VGG-16 + Random Forest model showcased commendable performance with a training accuracy of 93%. It continued to perform well during the testing phase, achieving an accuracy of 82%. Consequently, the model achieved an average accuracy of 87%. These results indicate the efficacy of the VGG-16 + Random Forest model in accurately categorizing ADHD-related data, showcasing its potential as a valuable tool for early detection.

Furthermore, the VGG-19 + Random Forest model exhibited notable accuracy in the training phase, achieving a training accuracy of 91%. During the testing phase, it maintained a high level of performance with an accuracy of 80%. The model's average accuracy stood at an impressive 85%, reinforcing its effectiveness in accurately classifying ADHD-related data.

The obtained results from these models demonstrate their potential to serve as reliable tools for aiding doctors in the early detection of ADHD among children. Leveraging machine learning techniques such as Random Forest in conjunction with established models like Inception V3, VGG-16, and VGG-19, can provide doctors with valuable insights and support in accurately identifying ADHD cases. These findings underscore the significance of machine learning in assisting medical professionals and enhancing the efficiency and accuracy of ADHD detection in children.

4.3.3.2 Confusion Matrix

Training Dataset

Figure 4.16 shows the confusion matrix for the training dataset, utilizing the Inception v3 + Random Forest model, unveiled that 198 out of 226 instances were correctly classified as ADHD (true positive), while 28 instances that were predicted as normal were actually ADHD (false positive). The model accurately identified 217 instances as normal (true negative) but misclassified 9 instances as ADHD when they were actually normal (false negative).

		Predict	
		adhd	**normal**
Actual	**adhd**	**198**	**28**
	normal	**9**	**217**

Figure 4.16 Confusion matrix of Random Forest classifier with Inception v3 training dataset

The analysis of the training dataset using the VGG-16 + Random Forest model yielded insightful results in the form of a confusion matrix. Figure 4.17 shows that among the 226 instances examined, a noteworthy outcome was observed. The model demonstrated impressive performance by correctly identifying 198 cases as ADHD, showcasing its ability to achieve true positive results. However, it also exhibited a small number of misclassifications, with 28 instances being falsely labelled as normal when they were actually ADHD (false positive). On the other hand, the model excelled in accurately identifying 221 instances as normal (true negative), indicating its proficiency in recognizing non-ADHD cases. Nevertheless, the model did encounter a few instances where it incorrectly identified them as ADHD, when in fact they were normal (false negative), resulting in a total of 5 such misclassifications.

		Predict	
		adhd	**normal**
Actual	**adhd**	**198**	**28**
	normal	**5**	**221**

Figure 4.17 Confusion matrix of Random Forest classifier with VGG-16 training dataset

Upon analyzing the training dataset using the VGG-19 + Random Forest model, the resulting confusion matrix revealed crucial insights. Figure 4.18 shoes among the 226 instances examined, the model exhibited commendable performance by accurately classifying 198 cases as ADHD, demonstrating its ability to achieve true positive results. However, there were instances where the model falsely labelled 28 cases as normal when they were actually ADHD (false positive). Conversely, the model excelled in correctly identifying 216 instances as normal (true negative), indicating its proficiency in recognizing non-ADHD cases. Nonetheless, the model encountered a few misclassifications where it erroneously identified 10 instances as ADHD when they were, in fact, normal (false negative).

		Predict	
		adhd	normal
Actual	adhd	198	28
	normal	10	216

Figure 4.18 Confusion matrix of Random Forest classifier with VGG-19 training dataset

Testing Dataset

Figure 4.19 shows the confusion matrix for the testing dataset, utilizing the Inception v3 + Random Forest model, unveiled that 46 out of 57 instances were correctly classified as ADHD (true positive), while 11 instances that were predicted as normal (false positive). The model accurately identified 46 instances as normal (true negative) but misclassified 11 instances as ADHD (false negative).

		Predict	
		adhd	**normal**
Actual	**adhd**	46	11
	normal	11	46

Figure 4.19 Confusion matrix of Random Forest classifier with Inception v3 testing dataset

Figure 4.20 shows the confusion matrix for the testing dataset, using the VGG-16 + Random Forest model, revealed that out of 57 instances, 48 were correctly classified as ADHD (true positive), while 9 instances were falsely predicted as normal (false positive). The model accurately identified 46 instances as normal (true negative) but misclassified 11 instances as ADHD (false negative).

		Predict	
		adhd	**normal**
Actual	**adhd**	**48**	**9**
	normal	**11**	**46**

Figure 4.20 Confusion matrix of Random Forest classifier with VGG-16 testing dataset

Upon evaluating the testing dataset using the VGG-19 + Random Forest model, a comprehensive confusion matrix was obtained, providing valuable insights. Figure 4.21 shows that out of the 57 instances examined, the model demonstrated proficiency by correctly classifying 45 cases as ADHD, indicating its ability to achieve true positive results. However, there were instances where the model falsely identified 12 cases as normal when they were actually ADHD (false positive). On the other hand, the model excelled in accurately recognizing 47 instances as normal (true negative), showcasing its competence in identifying non-ADHD cases.

		Predict	
		adhd	**normal**
Actual	**adhd**	**45**	**12**
	normal	**10**	**47**

Figure 4.21 Confusion matrix of Random Forest classifier with VGG-19 testing dataset

4.3.4 Naïve Bayes Classifier

4.3.4.1 Classification Accuracy

In this Naïve Bayes classifier section, the best pipeline model will be selected and compare the classification accuracy. From the Table 4.9, it shows the classification accuracy of the different transfer learning model with the Naïve Bayes classifier. The average classification accuracy graph is shown in Figure 4.22.

Table 4.9 Classification accuracy of the Naïve Bayes Model

Model	Classification Accuracy (%)		
	Training	Testing	Average
Inception v3 + Naïve Bayes	86	82	84
VGG-16 + Naïve Bayes	89	80	84
VGG-19 + Naïve Bayes	92	82	87

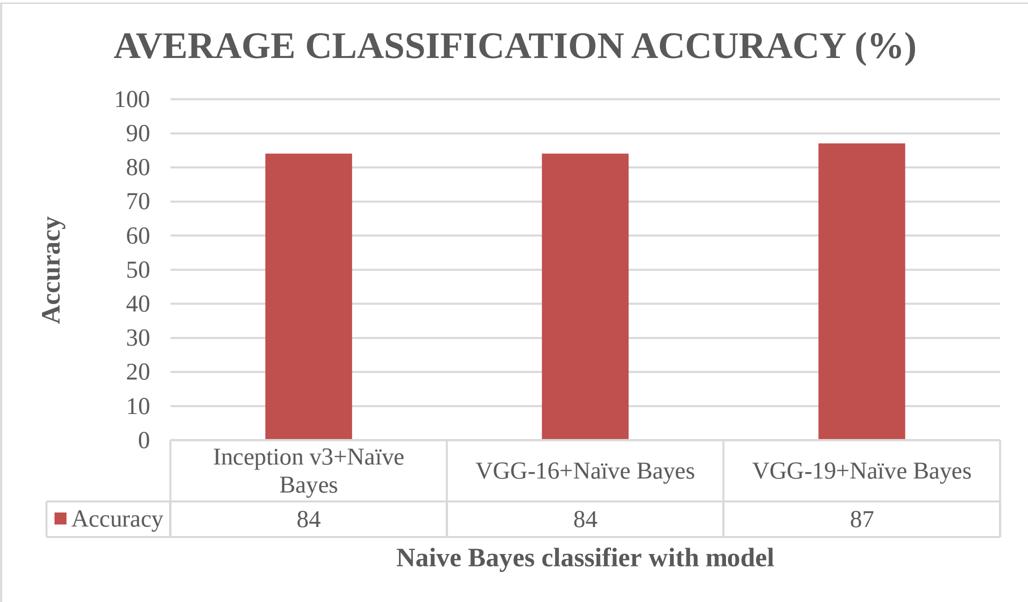

Figure 4.22 Average Classification Accuracy of Naïve Bayes classifier with model

Upon evaluating the performance of the models, it was observed that the combination of Inception V3 with the Naïve Bayes algorithm yielded impressive results. The Inception V3 + Naïve Bayes model demonstrated exceptional training accuracy, reaching 86%. During the testing phase, it exhibited a satisfactory accuracy of 82%, contributing to an overall average accuracy of 84%.

Similarly, the VGG-16 + Naïve Bayes model showcased commendable performance with a training accuracy of 89%. It continued to perform well during the testing phase, achieving an accuracy of 80%. Consequently, the model achieved an average accuracy of 84%.

Furthermore, the VGG-19 + Naïve Bayes model exhibited notable accuracy in the training phase, achieving a training accuracy of 92%. During the testing phase, it maintained a high level of performance with an accuracy of 82%. The model's average accuracy stood at an impressive 87%, reinforcing its effectiveness in accurately classifying ADHD-related data.

4.3.4.2 Confusion Matrix

Training Dataset

The confusion matrix for the training dataset, utilizing the Inception v3 + Naïve Bayes model, Figure 4.23 unveiled that 191 out of 226 instances were correctly classified as ADHD (true positive), while 35 instances that were predicted as normal were actually ADHD (false positive). The model accurately identified 198 instances as normal (true negative) but misclassified 28 instances as ADHD when they were actually normal (false negative).

		Predict	
		adhd	normal
Actual	adhd	191	35
	normal	28	198

Figure 4.23 Confusion matrix of Naïve Bayes classifier with Inception v3 training dataset

Upon analyzing the training dataset using the VGG-16 + Naïve Bayes model, the resulting confusion matrix revealed crucial insights. Figure 2.24 shows that among the 226 instances examined, the model exhibited commendable performance by accurately classifying 206 cases as ADHD, demonstrating its ability to achieve true positive results. However, there were instances where the model falsely labelled 20 cases as normal when they were actually ADHD (false positive). Conversely, the model excelled in correctly identifying 195 instances as normal (true negative), indicating its proficiency in recognizing non-ADHD cases. Nonetheless, the model encountered a few misclassifications where it erroneously identified 31 instances as ADHD when they were, in fact, normal (false negative).

		Predict	
		adhd	**normal**
Actual	**adhd**	**206**	**20**
	normal	**31**	**195**

Figure 4.24 Confusion matrix of Naïve Bayes classifier with VGG-16 training dataset

The confusion matrix for the training dataset, utilizing the VGG-19 + Naïve Bayes model, unveiled that 206 out of 226 instances were correctly classified as ADHD (true positive), while 20 instances that were predicted as normal were actually ADHD (false positive). The model accurately identified 199 instances as normal (true negative) but misclassified 27 instances as ADHD when they were actually normal (false negative).

		Predict	
		adhd	normal
Actual	adhd	206	20
	normal	27	199

Figure 4.25 Confusion matrix of Naïve Bayes classifier with VGG-19 training dataset

Testing Dataset

The Figure 4.26 shows the confusion matrix for the testing dataset, utilizing the Inception v3 + Naïve Bayes model, unveiled that 45 out of 57 instances were correctly classified as ADHD (true positive), while 12 instances that were predicted as normal (false positive). The model accurately identified 49 instances as normal (true negative) but misclassified 8 instances as ADHD (false negative).

		Predict	
		adhd	**normal**
Actual	**adhd**	**45**	**12**
	normal	**8**	**49**

Figure 4.26 Confusion matrix of Naïve Bayes classifier with Inception v3 testing dataset

The confusion matrix for the testing dataset, using the VGG-16 + Naïve Bayes model, Figure 4.27 revealed that out of 57 instances, 46 were correctly classified as ADHD (true positive), while 11 instances were falsely predicted as normal (false positive). The model accurately identified 45 instances as normal (true negative) but misclassified 12 instances as ADHD (false negative).

		Predict	
		adhd	**normal**
Actual	**adhd**	**46**	**11**
	normal	**12**	**45**

Figure 4.27 Confusion matrix of Naïve Bayes classifier with VGG-16 testing dataset

Upon evaluating the testing dataset using the VGG-19 + Naïve Bayes model, a comprehensive confusion matrix was obtained, providing valuable insights. Figure 4.28 shows that out of the 57 instances examined, the model demonstrated proficiency by correctly classifying 44 cases as ADHD, indicating its ability to achieve true positive results. However, there were instances where the model falsely identified 13 cases as normal when they were actually ADHD (false positive). On the other hand, the model excelled in accurately recognizing 43 instances as normal (true negative), showcasing its competence in identifying non-ADHD cases.

		Predict	
		adhd	**normal**
Actual	**adhd**	**44**	**13**
	normal	**14**	**43**

Figure 4.28 Confusion matrix of Naïve Bayes classifier with VGG-19 testing dataset

4.3.5 Logistic Regression Classifier

4.3.5.1 Classification Accuracy

In this Logistic Regression classifier section, the best pipeline model will be selected and compare the classification accuracy. From the Table 4.10, it shows the classification accuracy of the different transfer learning model with the Logistic Regression classifier. The average classification accuracy graph is shown in Figure 4.29.

Table 4.10 Classification accuracy of the Logistic Regression Model

Model	Classification Accuracy (%)		
	Training	**Testing**	**Average**
Inception v3 + Logistic Regression	93	82	87
VGG-16 + Logistic Regression	92	83	87
VGG-19 + Logistic Regression	92	82	87

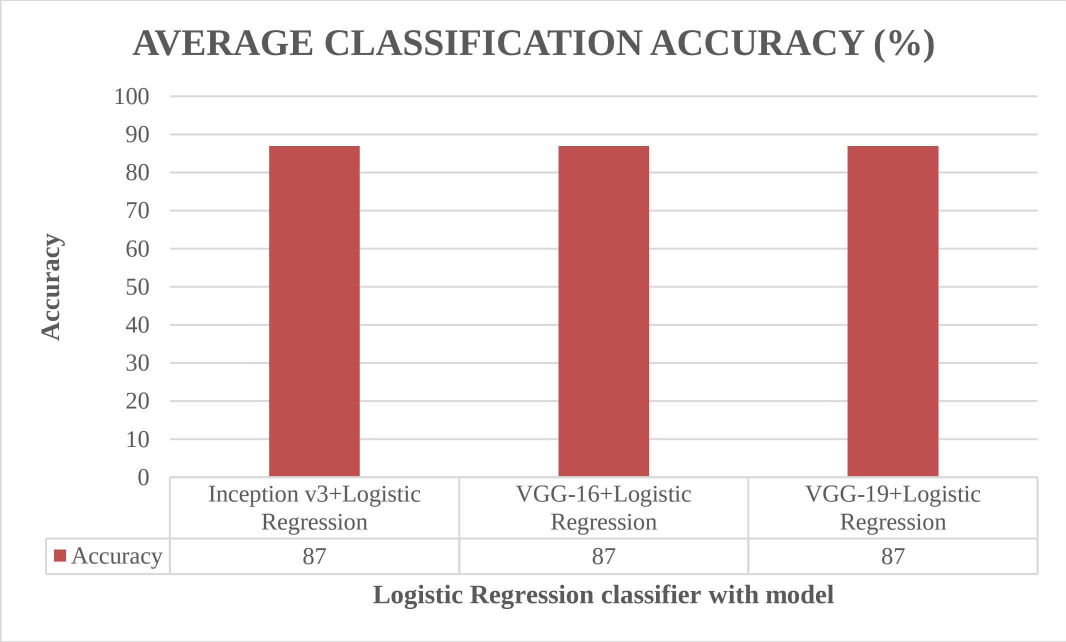

Figure 4.29 Average Classification Accuracy of Logistic Regression classifier with model

Upon evaluating the performance of the models, it was observed that the combination of Inception V3 with the Logistic Regression algorithm yielded impressive results. The Inception V3 + Logistic Regression model demonstrated exceptional training accuracy, reaching an impressive 93%. During the testing phase, it exhibited a satisfactory accuracy of 82%, contributing to an overall average accuracy of 87%.

Similarly, the VGG-16 + Logistic Regression model showcased commendable performance with a training accuracy of 92%. It continued to perform well during the testing phase, achieving an accuracy of 83%. Consequently, the model achieved an average accuracy of 87%.

Furthermore, the VGG-19 + Logistic Regression model exhibited notable accuracy in the training phase, achieving a training accuracy of 92%. During the testing phase, it maintained a high level of performance with an accuracy of 82%. The model's average accuracy stood at an impressive 87%, reinforcing its effectiveness in accurately classifying ADHD-related data.

4.3.5.2 Confusion Matrix

Training Dataset

The confusion matrix for the training dataset, utilizing the Inception v3 + Logistic Regression model, Figure 4.30 unveiled that 19 out of 226 instances were correctly classified as ADHD (true positive), while 27 instances that were predicted as normal were actually ADHD (false positive). The model accurately identified 220 instances as normal (true negative) but misclassified 6 instances as ADHD when they were actually normal (false negative).

		Predict	
		adhd	**normal**
Actual	**adhd**	**199**	**27**
	normal	**6**	**220**

Figure 4.30 Confusion matrix of Logistic Regression classifier with Inception v3 training dataset

The Figure 4.31 shows that upon analyzing the training dataset using the VGG-16 + Logistic Regression model, the resulting confusion matrix revealed crucial insights. Among the 226 instances examined, the model exhibited commendable performance by accurately classifying 198 cases as ADHD, demonstrating its ability to achieve true positive results. However, there were instances where the model falsely labelled 28 cases as normal when they were actually ADHD (false positive). Conversely, the model excelled in correctly identifying 223 instances as normal (true negative), indicating its proficiency in recognizing non-ADHD cases. Nonetheless, the model encountered a few misclassifications where it erroneously identified 3 instances as ADHD when they were, in fact, normal (false negative).

		Predict	
		adhd	**normal**
Actual	**adhd**	**198**	**28**
	normal	**3**	**223**

Figure 4.31 Confusion matrix of Logistic Regression classifier with VGG-16 training dataset

Figure 4.32 shows the confusion matrix for the training dataset, utilizing the VGG-19 + Logistic Regression model, unveiled that 198 out of 226 instances were correctly classified as ADHD (true positive), while 28 instances that were predicted as normal were actually ADHD (false positive). The model accurately identified 218 instances as normal (true negative) but misclassified 8 instances as ADHD when they were actually normal (false negative).

		Predict	
		adhd	**normal**
Actual	**adhd**	**198**	**28**
	normal	**8**	**218**

Figure 4.32 Confusion matrix of Logistic Regression classifier with VGG-19 training dataset

Testing Dataset

The confusion matrix for the testing dataset, utilizing the Inception v3 + Logistic Regression model, Figure 4.33 unveiled that 46 out of 57 instances were correctly classified as ADHD (true positive), while 11 instances that were predicted as normal (false positive). The model accurately identified 48 instances as normal (true negative) but misclassified 9 instances as ADHD (false negative).

		Predict	
		adhd	normal
Actual	adhd	46	11
	normal	9	48

Figure 4.33 Confusion matrix of Logistic Regression classifier with Inception v3 testing dataset

The confusion matrix for the testing dataset, using the VGG-16 + Logistic Regression model, Figure 4.34 revealed that out of 57 instances, 47 were correctly classified as ADHD (true positive), while 10 instances were falsely predicted as normal (false positive). The model accurately identified 48 instances as normal (true negative) but misclassified 9 instances as ADHD (false negative).

		Predict	
		adhd	**normal**
Actual	**adhd**	**47**	**10**
	normal	**9**	**48**

Figure 4.34 Confusion matrix of Logistic Regression classifier with VGG-16 testing dataset

Upon evaluating the testing dataset using the VGG-19 + Logistic Regression model, a comprehensive confusion matrix was obtained, providing valuable insights. Figure 4.35 shows that out of the 57 instances examined, the model demonstrated proficiency by correctly classifying 45 cases as ADHD, indicating its ability to achieve true positive results. However, there were instances where the model falsely identified 12 cases as normal when they were actually ADHD (false positive). On the other hand, the model excelled in accurately recognizing 48 instances as normal (true negative), showcasing its competence in identifying non-ADHD cases.

		Predict	
		adhd	**normal**
Actual	**adhd**	**45**	**12**
	normal	**9**	**48**

Figure 4.35 Confusion matrix of Logistic Regression classifier with VGG-19 testing dataset

4.4 Best Model Selection

Table 4.11 Highest Classification accuracy of the models

Model	Classification Accuracy (%)		
	Training	Testing	Average
VGG-19 + k-NN	88	82	85
VGG-16 + SVM	93	84	88
VGG-16 + Random Forest	93	82	87
VGG-19 + Naïve Bayes	92	82	87
Inception v3 + Logistic Regression	93	82	87

During the research testing and validation phase, multiple classifiers with different models were employed to obtain the highest classification accuracy in detecting ADHD among children using fMRI images. Among these classifiers, the VGG-19 + k-Nearest Neighbors (k-NN) model emerged with an impressive average accuracy of 85%. On the other hand, for the Support Vector Machine (SVM) classifier, the transfer learning model VGG-16 + SVM demonstrated a notable average accuracy of 88%, indicating its effectiveness in accurately classifying ADHD cases. Additionally, the VGG-16 + Random Forest model, VGG-19 + Naïve Bayes model, Inception v3 + Logistic Regression achieved a commendable average accuracy of 87%, further showcasing its potential in ADHD detection. These results highlight the importance of exploring different classifiers and models to identify the most suitable approach for accurately diagnosing ADHD using fMRI images.

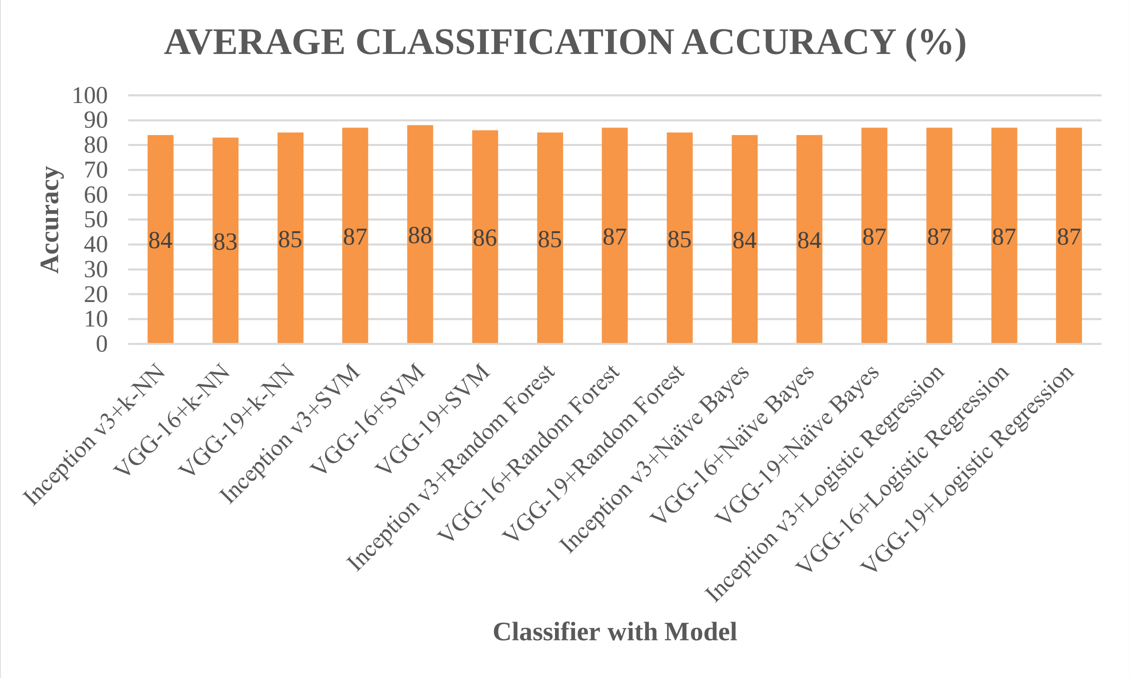

Figure 4.36 Average classification accuracy of all models

The chart Figure 4.36 provided showcases the classification accuracy (CA) of different classifiers in the context of detecting ADHD using fMRI images. Among these classifiers, the VGG-16 + SVM model stands out with a remarkable classification accuracy of 88%. This high accuracy underscores the potential of the VGG-16 + SVM model in effectively identifying ADHD cases through the analysis of fMRI images.

The VGG-16 + SVM model combines the power of the VGG-16 convolutional neural network and the support vector machine algorithm to leverage the unique patterns and features present in fMRI images for accurate ADHD detection. With its deep learning capabilities, the VGG-16 model is adept at extracting intricate visual features from the fMRI images, while the SVM algorithm excels in creating a robust decision boundary to distinguish between ADHD and non-ADHD cases.

By utilizing fMRI images, which provide insights into brain activity and connectivity, the VGG-16 + SVM model can effectively capture subtle differences and abnormalities associated with ADHD. It leverages the inherent characteristics of fMRI data to identify specific brain regions and activation patterns that are indicative of ADHD, enabling early detection and intervention.

The exceptional classification accuracy achieved by the VGG-16 + SVM model signifies its potential as a valuable tool in assisting clinicians and researchers in the diagnosis and management of ADHD. With its ability to process and analyze large volumes of fMRI data, this model offers an automated and objective approach to ADHD detection, reducing subjectivity and enhancing the efficiency and reliability of diagnosis.

The successful utilization of the VGG-16 + SVM model in detecting ADHD with fMRI images opens up new possibilities for improving the accuracy and accessibility of ADHD assessments. Its integration into clinical practice has the potential to facilitate early identification, leading to timely interventions and improved outcomes for individuals affected by ADHD.

Table 4.12 Performance Metric between 2 classes for VGG-16+SVM model

Dataset	Target	Performance Metric				
		CA	F1 Score	Precision	Recall	Sample Size
Train	ADHD	0.93	0.93	0.98	0.88	226
	Normal	0.93	0.93	0.89	0.99	226
Test	ADHD	0.84	0.83	0.87	0.81	57
	Normal	0.84	0.85	0.82	0.88	57

Table 4.13 Performance Metric for VGG-16+SVM model

Dataset	Performance Metric				
	CA	F1 Score	Precision	Recall	Sample Size
Train	0.93	0.93	0.94	0.93	452
Test	0.84	0.84	0.84	0.84	114

The table provides a comprehensive overview of the performance metrics for the selected model, VGG-16 + SVM. It highlights key measures such as classification accuracy (CA), F1 score, precision, recall, and the sample size for each dataset. These metrics offer valuable insights into the model's effectiveness in accurately classifying ADHD cases using the VGG-16 architecture combined with the SVM algorithm. By considering multiple performance indicators, the table allows for a more comprehensive evaluation of the model's overall performance and its ability to successfully detect ADHD using the provided datasets. For others non-selected models, the performance report can be referring on APPENDIX A.

4.5 Summary

In conclusion, in the context of image classification, transfer learning models such as VGG-16, VGG-19, and Inception V3 were evaluated. These models showed promising performance in accurately classifying ADHD and normal cases using fMRI images. The best pipeline of the model was VGG-16 with SVM classifier which have the highest classification accuracy in the training dataset and testing dataset. The average of the classification accuracy is 88%. In addition, this model has the good result in the performance metric. The discussed models and their performance in classifying ADHD among children using fMRI images highlight their potential for aiding in the early detection of ADHD. Overall, the results indicate that machine learning models, particularly those leveraging transfer learning techniques, can contribute to the early detection of ADHD among children, thereby supporting improved diagnosis and intervention strategies.

CHAPTER 5

CONCLUSION

5.1 Introduction

This chapter is about the conclusion and recommendation on future works on this research. The conclusion will state in Section 5.2 based on the result and discussion done in Chapter 4 and also the objective and aim of the research. Then the recommendation and the future works that may do for this research will be also outlined well in Section 5.3.

5.2 Conclusion

In conclusion, fMRI image classification has emerged as a promising approach for detecting ADHD among children. The use of machine learning models, such as VGG-16, VGG-19, and Inception V3, combined with various algorithms such as k-Nearest Neighbors (kNN), Support Vector Machines (SVM), Random Forest, Naïve Bayes and Logistic Regression has shown encouraging results in accurately classifying ADHD and normal cases based on fMRI images.

Numerous research studies have been conducted to explore and refine the methods and tools for ADHD detection, demonstrating the ongoing efforts in this field. The application of transfer learning, which leverages pre-trained models, has proven to be effective in improving classification accuracy. The findings highlight the potential of fMRI image classification as a valuable tool for early detection of ADHD. By utilizing these techniques, healthcare professionals, parents, and teachers can identify children at risk and provide timely interventions and support. However, further research and refinement are needed to enhance the accuracy, reliability, and generalizability of these models for real-world clinical applications.

In this research, we conducted an analysis using a dataset of 566 fMRI images obtained from the ADHD-200 competition. The dataset comprised two classes, namely ADHD and Normal, allowing us to explore the potential of fMRI image classification for detecting ADHD among children.

The research methodology involved several key steps. Firstly, we focused on data collection and acquisition, ensuring the inclusion of a diverse set of fMRI images representing both ADHD and Normal cases. Next, rigorous data pre-processing techniques were employed to enhance the quality and consistency of the dataset.

To evaluate the performance of our models, we divided the dataset into training and testing sets using an 80:20 split. This ensured that the models were trained on a substantial portion of the data while maintaining a robust evaluation on unseen instances.

Feature extraction was a critical step in our methodology, and we leveraged the power of transfer learning to extract meaningful features from the fMRI images. This approach allowed us to utilize pre-trained models, such as VGG-16, VGG-19, and Inception V3, to capture relevant patterns and characteristics.

Classification algorithms were then applied to classify the fMRI images into the ADHD and Normal classes. We employed various algorithms, including k-Nearest Neighbors (kNN), Support Vector Machines (SVM), Random Forest, Naïve Bayes, and Logistic Regression to evaluate their performance in accurately categorizing the images.

To assess the effectiveness of the models, we utilized performance metrics such as classification accuracy (CA), F1 score, precision, recall, and sample size. These metrics provided valuable insights into the models' ability to correctly identify ADHD and Normal cases.

Through this research, our primary objective was to make a valuable contribution to the expanding realm of knowledge regarding fMRI image classification for the purpose of detecting ADHD. The meticulous and comprehensive methodology adopted in this study ensured a robust and rigorous analysis, serving as a solid foundation for future advancements in this field. The progress made in fMRI image classification for ADHD detection represents a substantial leap forward in comprehending and addressing this neurodevelopmental disorder, ultimately paving the way for early intervention strategies and enhanced outcomes for children affected by ADHD.

From the research endeavours in the realm of ADHD detection using image classification have demonstrated the efficacy of various models and techniques. Among them, the VGG-16 + SVM model emerged as the most successful in accurately identifying ADHD cases. With a remarkable average classification accuracy (CA) of 88% of training and testing dataset, this model showcased its potential as a reliable tool for ADHD detection using fMRI images. The utilization of transfer learning with the VGG-16 architecture, coupled with the SVM classifier, yielded outstanding results in terms of classification accuracy. This achievement highlights the significance of leveraging advanced machine learning techniques for improved ADHD diagnosis and underscores the importance of image classification in aiding early detection efforts. With further advancements in this field, the VGG-16 + SVM model holds promise for enhancing clinical decision-making and facilitating timely interventions to benefit children with ADHD.

In conclusion, the three (3) objectives have well achieved as following,

1) To study ADHD among children and machine learning model.

The feature of the fMRI image is well extracted with the transfer learning model with the Inception v3, VGG-16 and VGG-19 models as the features can be clearly works for the classifier kNN, SVM, Random Forest, Naïve Bayes and Logistic Regression. The most suitable which give high accuracy was the feature extracted by VGG-16 transfer learning model while the classifier was SVM model.

2) To develop a machine learning model for detecting ADHD among children.

The model is successfully developed with the optimization of hyper parameters have tuned and the best set of hyper parameters will be selected to formulate as the classification algorithms of the model with the extracted feature obtain from the best transfer learning model.

3) To evaluate the performance of the developed model

The performance of the developed and selected transfer learning model was well investigated as the classification accuracy (CA) and confusion matrix is the main performance observed.

5.3 Recommendation and Future Works

The application of image classification techniques, particularly using fMRI images, holds great promise for the detection of ADHD among children. Through the research, we utilized a dataset of 566 fMRI images from the ADHD-200 competition, consisting of two classes: ADHD and Normal. Employing a comprehensive methodology encompassing data collection, pre-processing, feature extraction through transfer learning using the Inception v3, VGG-16 and VGG-19 architecture, and classification with the k-NN, SVM, Random Forest, Naïve Bayes and Logistic Regression model.

The use of machine learning algorithms allows for the exploration of complex patterns in brain activity, aiding in the identification and differentiation of ADHD-related neural markers. The high accuracy achieved by our model highlights its potential as a valuable tool for early ADHD detection, providing opportunities for timely intervention and improved outcomes for affected children.

However, there are still areas for future work and improvement. It is recommended to expand the dataset to include a more diverse population and to explore additional imaging features or multimodal approaches that incorporate various data sources. Further optimization and validation of the models are also essential to enhance their generalizability and clinical applicability. Additionally, ensuring interpretability and explainability of the models can foster trust and facilitate their integration into clinical practice.

The research contributes to the growing body of knowledge in fMRI image classification for ADHD detection. The VGG-16+SVM model demonstrates promising accuracy, underscoring the potential of image classification techniques in aiding the diagnosis of ADHD. Continued research efforts and collaborations between researchers and clinicians will advance this field, ultimately leading to more accurate, accessible, and effective diagnostic tools for ADHD.

REFERENCES

Anas Brital. (2021). *Inception-V1 Architecture Explained* . https://anasbrital98.github.io/blog/2021/Inception-V1-CNN-Architecture-Explained/

Chen, Y., Tang, Y., Wang, C., Liu, X., Zhao, L., & Wang, Z. (2020a). ADHD classification by dual subspace learning using resting-state functional connectivity. *Artificial Intelligence in Medicine*, *103*. https://doi.org/10.1016/j.artmed.2019.101786

Chen, Y., Tang, Y., Wang, C., Liu, X., Zhao, L., & Wang, Z. (2020b). ADHD classification by dual subspace learning using resting-state functional connectivity. *Artificial Intelligence in Medicine*, *103*. https://doi.org/10.1016/j.artmed.2019.101786

Duda, M., Ma, R., Haber, N., & Wall, D. P. (2016). Use of machine learning for behavioral distinction of autism and ADHD. *Translational Psychiatry*, *6*(2). https://doi.org/10.1038/tp.2015.221

Jafari, P., Mehrabani-Zeinabad, K., Javadi, S., Ghanizadeh, A., & Bagheri, Z. (2022). A Machine Learning Approach to Assess Differential Item Functioning of the KINDL Quality of Life Questionnaire Across Children with and Without ADHD. *Child Psychiatry and Human Development*, *53*(5), 980–991. https://doi.org/10.1007/s10578-021-01179-6

Maniruzzaman, M., Shin, J., & Hasan, M. A. M. (2022). Predicting Children with ADHD Using Behavioral Activity: A Machine Learning Analysis. *Applied Sciences (Switzerland)*, *12*(5). https://doi.org/10.3390/app12052737

Maniruzzaman, M., Shin, J., Hasan, M. A. M., & Yasumura, A. (2022). Efficient Feature Selection and Machine Learning Based ADHD Detection Using EEG Signal. *Computers, Materials and Continua*, *72*(3), 5179–5195. https://doi.org/10.32604/cmc.2022.028339

N. Lytlea Marisa, Hammer Rubi, & R. Booth James. (n.d.). *A neuroimaging dataset on working memory and reward processing in children with and without ADHD*. https://doi.org/10.18112/openneuro.ds002424.v1.1.1

Niklas Donges. (2019). *What Is Random Forest? A Complete Guide | Built In*. https://builtin.com/data-science/random-forest-algorithm

Parashar, A., Kalra, N., Singh, J., & Goyal, R. K. (2021). Machine learning based framework for classification of children with adhd and healthy controls. *Intelligent Automation and Soft Computing*, *28*(3), 669–682. https://doi.org/10.32604/iasc.2021.017478

Riaz, A., Asad, M., Alonso, E., & Slabaugh, G. (2020). DeepFMRI: End-to-end deep learning for functional connectivity and classification of ADHD using fMRI. *Journal of Neuroscience Methods*, *335*. https://doi.org/10.1016/j.jneumeth.2019.108506

APPENDIX A
PERFORMANCE REPORT

1. Inception v3 + k-NN Model

Dataset	Performance Metric				
	CA	F1 Score	Precision	Recall	Sample Size
Train	0.88	0.88	0.89	0.88	452
Test	0.81	0.81	0.81	0.81	114

2. VGG-16 + k-NN Model

Dataset	Performance Metric				
	CA	F1 Score	Precision	Recall	Sample Size
Train	0.87	0.87	0.88	0.87	452
Test	0.79	0.79	0.79	0.79	114

3. VGG-19 + k-NN Model

Dataset	Performance Metric				
	CA	F1 Score	Precision	Recall	Sample Size
Train	0.88	0.88	0.88	0.88	452
Test	0.82	0.82	0.83	0.82	114

4. Inception v3 + SVM Model

Dataset	Performance Metric				
	CA	F1 Score	Precision	Recall	Sample Size
Train	0.93	0.93	0.94	0.93	452
Test	0.82	0.82	0.82	0.82	114

5. VGG-19 + SVM Model

Dataset	Performance Metric				
	CA	F1 Score	Precision	Recall	Sample Size
Train	0.92	0.92	0.93	0.92	452
Test	0.81	0.81	0.81	0.81	114

6. Inception v3 + Random Forest Model

Dataset	Performance Metric				
	CA	F1 Score	Precision	Recall	Sample Size
Train	0.92	0.93	0.93	0.93	452
Test	0.81	0.81	0.81	0.81	114

7. VGG-16 + Random Forest Model

Dataset	Performance Metric				
	CA	F1 Score	Precision	Recall	Sample Size
Train	0.92	0.92	0.92	0.92	452
Test	0.85	0.85	0.85	0.85	114

8. VGG-19 + Random Forest Model

Dataset	Performance Metric				
	CA	F1 Score	Precision	Recall	Sample Size
Train	0.92	0.92	0.92	0.92	452
Test	0.82	0.82	0.82	0.82	114

9. Inception v3 + Naïve Bayes Model

Dataset	Performance Metric				
	CA	F1 Score	Precision	Recall	Sample Size
Train	0.86	0.86	0.86	0.86	452
Test	0.82	0.82	0.83	0.82	114

10. VGG-16 + Naïve Bayes Model

Dataset	Performance Metric				
	CA	F1 Score	Precision	Recall	Sample Size
Train	0.89	0.89	0.89	0.89	452
Test	0.80	0.80	0.80	0.80	114

11. VGG-19 + Naïve Bayes Model

Dataset	Performance Metric				
	CA	F1 Score	Precision	Recall	Sample Size
Train	0.90	0.90	0.90	0.90	452
Test	0.76	0.76	0.76	0.76	114

12. Inception v3 + Logistic Regression Model

Dataset	Performance Metric				
	CA	F1 Score	Precision	Recall	Sample Size
Train	0.93	0.93	0.93	0.93	452
Test	0.82	0.82	0.82	0.82	114

13. VGG-16 + Logistic Regression Model

Dataset	Performance Metric				
	CA	F1 Score	Precision	Recall	Sample Size
Train	0.93	0.93	0.94	0.93	452
Test	0.83	0.83	0.83	0.83	114

14. VGG-19 + Logistic Regression Model

Dataset	Performance Metric				
	CA	F1 Score	Precision	Recall	Sample Size
Train	0.92	0.92	0.92	0.92	452
Test	0.82	0.82	0.82	0.82	114

APPENDIX B
GANTT CHART

		SEM 1														SEM2													
No	Activity	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	W15	W16	W17	W18	W19	W20	W21	W22	W23	W24	W25	W26	W27	W28
1	Research objective and scope definition																												
2	Literature Review																												
3	Research design planning																												
4	Selecting model																												
5	Dataset collection																												
6	Development of classification algorithm																												
7	Evaluate classification algorithm																												
8	Thesis writing																												

APPENDIX C

```
import numpy as np
import pandas as pd
from keras.utils import image_utils
from keras.preprocessing import image
from PIL import Image
import tensorflow.keras as keras
import tensorflow as tf
import matplotlib.image as img
from keras.utils import to_categorical
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
import random
import os

filenames = os.listdir('C:\\Atiqah\\MRI_Scan\\')
data_path = "C:/Atiqah/MRI_Scan/"

from keras.preprocessing import image
from PIL import Image

size = 224

categories = []
image = []
for filename in filenames:
    category = filename.split(' (')[0]
    if category == 'adhd':
        img_path = os.path.join(data_path, filename);
        img = image_utils.load_img(img_path)
        img = img.resize((size,size),Image.ANTIALIAS)
        image_array = image_utils.img_to_array(img)
        image.append(image_array)
        #image.append(np.array(image_array))
        categories.append(0)  #The categories are set as labels
    elif category == 'normal':
        img_path = os.path.join(data_path, filename);
        img = image_utils.load_img(img_path)
        img = img.resize((size,size),Image.ANTIALIAS)
        image_array = image_utils.img_to_array(img)
        image.append(image_array)
        #image.append(np.array(image_array))
        categories.append(1)

df = pd.DataFrame({
    'filename': filenames,
    'category': categories,
})

df

from sklearn.model_selection import train_test_split
xtrain, xtest, ytrain, ytest = train_test_split(image, categories,
test_size = 0.2, stratify = categories)
```

```
#xtrain, xtest, ytrain, ytest = train_test_split(X, y, test_size =
0.2, stratify = y)

xtrain = np.array(xtrain)
ytrain = np.array(ytrain)

xtest = np.array(xtest)
ytest = np.array(ytest)
from keras.applications.vgg16 import VGG16
VGG_16 = VGG16(weights = 'imagenet', include_top = False, input_shape
= (224,224,3))
print(VGG_16.summary())
xtrain_feature = VGG_16.predict(xtrain)
xtest_feature = VGG_16.predict(xtest)

xtrain_flatted = xtrain_feature.reshape((xtrain_feature.shape[0],
7*7*512))
xtest_flatted = xtest_feature.reshape((xtest_feature.shape[0],
7*7*512))

print(xtrain_feature.shape, xtrain_flatted.shape)
print(xtest_feature.shape,xtest_flatted.shape)


import joblib

joblib.dump(xtrain_flatted, "xtrain_VGG16SVM_5C.dat")
joblib.dump(xtest_flatted, "xtest_VGG16SVM_5C.dat")

joblib.dump(ytrain, "ytrain_VGG16SVM_5C.dat")
joblib.dump(ytest, "ytest_VGG16SVM_5C.dat")

from sklearn.svm import SVC
from sklearn.model_selection import GridSearchCV

SVM = SVC()
parameters = {
        'kernel': ['linear', 'rbf', 'sigmoid'],
        'C': [0.1, 1, 10],
        'gamma': [0.1, 1, 10],
        'degree' : [2,3]
        }
print(parameters)

SVM_para = GridSearchCV(SVM, parameters, cv = 5)

#Define funtion to print report with format
def parameter_result (para_result):
    print('Best parameters set found on SVM:
{}'.format(para_result.best_params_))

    means = para_result.cv_results_['mean_test_score']
    stds = para_result.cv_results_['std_test_score']

    for mean, std, params in zip(means, stds,
para_result.cv_results_['params']):
```

```
        print('\n {} (+/-{}) for {}'.format(round(mean, 3),
round(std*2, 3), params))

#Print and find best parammeters
SVM_para.fit(xtrain_flatted, ytrain)

parameter_result(SVM_para)

SVM_best_para = SVM_para.best_estimator_

joblib.dump(SVM_best_para,'VGG16+SVM_Model_5C.pkl')


from sklearn.model_selection import cross_val_score

#Load model
SVM_model = joblib.load('VGG16+SVM_Model_5C.pkl')

#Print the CV result
scores = cross_val_score(SVM_model, xtrain_flatted, ytrain, cv = 5)
acc = round(scores.mean(), 2)
stds = round(scores.std(), 2)
print(acc, stds)


#Train the model
start = time.time()
SVM_model.fit(xtrain_flatted, ytrain)
end = time.time()

print ("Running time = {end - start}")

#Defined the function for the classification report, acc score,
confusion matrix
def model_report(ytrue, yprey):
    #Import the classification report
    from sklearn.metrics import classification_report

    #Import the accuracy score
    from sklearn.metrics import accuracy_score

    #Import the confusion matrix
    from sklearn.metrics import confusion_matrix
    #Defined and print the classification report
    cp = classification_report(ytrue, yprey, target_names = ['ADHD',
'Normal'])
    print('Classification Report')
    print(cp)

    #Defined and print the acc score
    acs = accuracy_score(ytrue, yprey)
    #print('\n Accuracy Score = {}'.format(round(acs), 2))


    #Defined and print the matrix
    cm = confusion_matrix(ytrue, yprey)
    print('\n Confusion Matrix')
```

```
    print(cm)

    #Set the format of the printed matrix
    xlabels = ['ADHD', 'Normal']
    ylabels = ['ADHD', 'Normal']

    sns.set(font_scale = 1.4)
    sns.heatmap(cm, square = True, annot = True, xticklabels =
xlabels,
                yticklabels = ylabels, annot_kws = {"size": 14},
cmap='Blues', fmt='g')

    plt.title('Confusion Matrix')
    plt.xlabel('Predict')
    plt.ylabel('Actual')

#Predict the class of train data
prey_train = SVM_model.predict(xtrain_flatted)

#Print the report
print('Train Result')
model_report(ytrain, prey_train)


#Predict the class of test data
prey_test = SVM_model.predict(xtest_flatted)

#Print the report
print('Test Result')
model_report(ytest, prey_test)
```